HDFS Erasure Coding in Action

2016/9/1

Takuya Fukudome, Yahoo Japan Corporation

Yahoo! JAPAN

- Yahoo! JAPAN 検索 Search
- Yahoo! JAPAN ショッピング Shopping
- ヤフオク! Auction
- Yahoo! JAPAN ニュース News
- Yahoo! JAPAN メール Mail

・
・
・

Providing over 100 services
on PC and mobile
64.9billion PV/month

Hadoop cluster
Total 6000Nodes
120PB of variety data

Agenda

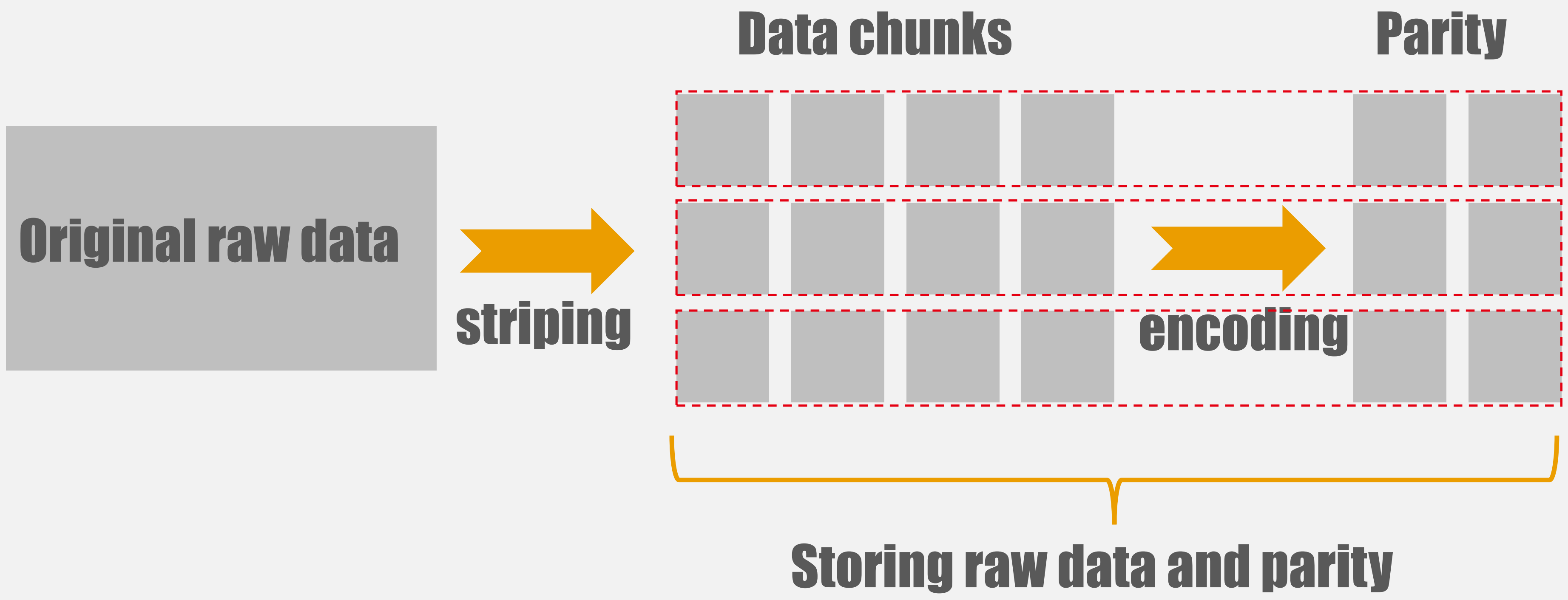
1. **About HDFS Erasure Coding**
   - Key points
   - Implementation
   - Compare to replication
2. **HDFS Erasure Coding Tests**
   - System Tests
     - Basic Operations
     - Reconstruct Erasure Coding Blocks
     - Other features
   - Performance Tests
3. **Usage in Yahoo! JAPAN**
   - Principal workloads in our production
   - Future plan

Agenda

1. **About HDFS Erasure Coding**
    - Key points
    - Implementation
    - Compare to replication
2. **HDFS Erasure Coding Tests**
    - System Tests
        - Basic Operations
        - Reconstruct Erasure Coding Blocks
        - Other features
    - Performance Tests
3. **Usage in Yahoo! JAPAN**
    - Principal workloads in our production
    - Future plan

What is Erasure Coding?

A technique to earn data availability and durability



Key points

- Missing or corrupted data will be reconstruct with living data and parity
- Parity is typically smaller than original data

HDFS Erasure Coding

- New feature in Hadoop 3.0
- To reduce storage overhead
  - Half of the tripled replication
- Same durability as replication

Agenda

1. **About HDFS Erasure Coding**
   - Key points
   - Implementation
   - Compare to replication
2. **HDFS Erasure Coding Tests**
   - System Tests
     - Basic Operations
     - Reconstruct Erasure Coding Blocks
     - Other features
   - Performance Tests
3. **Usage in Yahoo! JAPAN**
   - Principal workloads in our production
   - Future plan

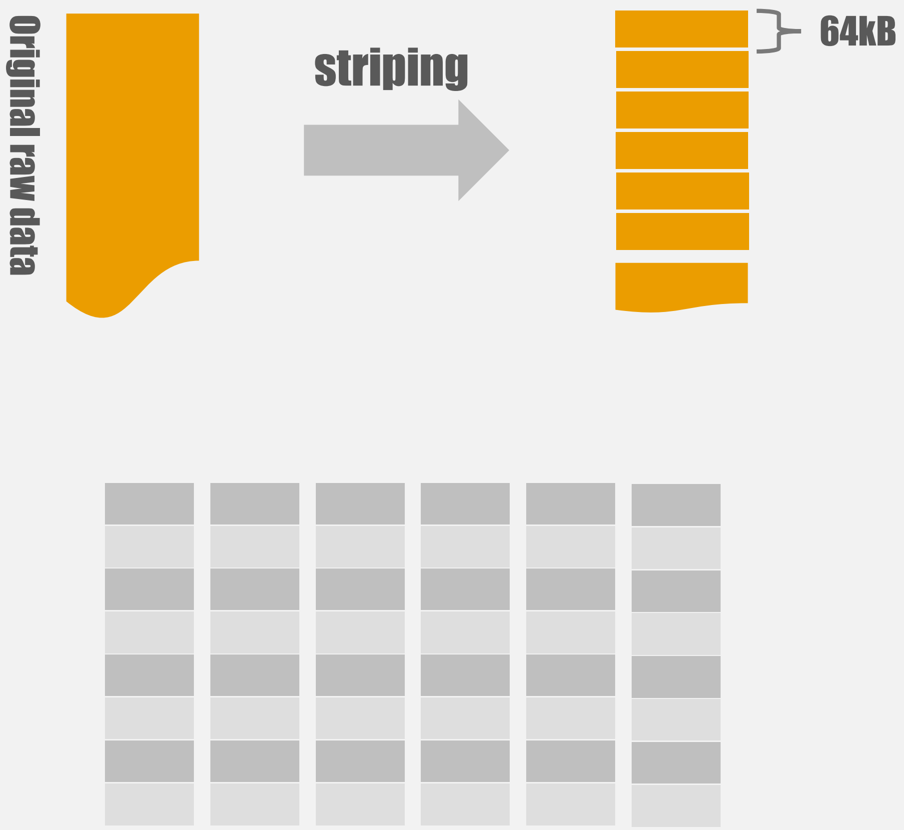
Implementation (Phase1)

- **Striped Block Layout**
  - Striping original raw data
  - Encode striped data to parity data
- **Target on cold data**
  - Not modified and rarely accessed
- **Reed Solomon(6,3) Codec**
  - System default codec
  - 6 data and 3 parity blocks

Striped Block Management

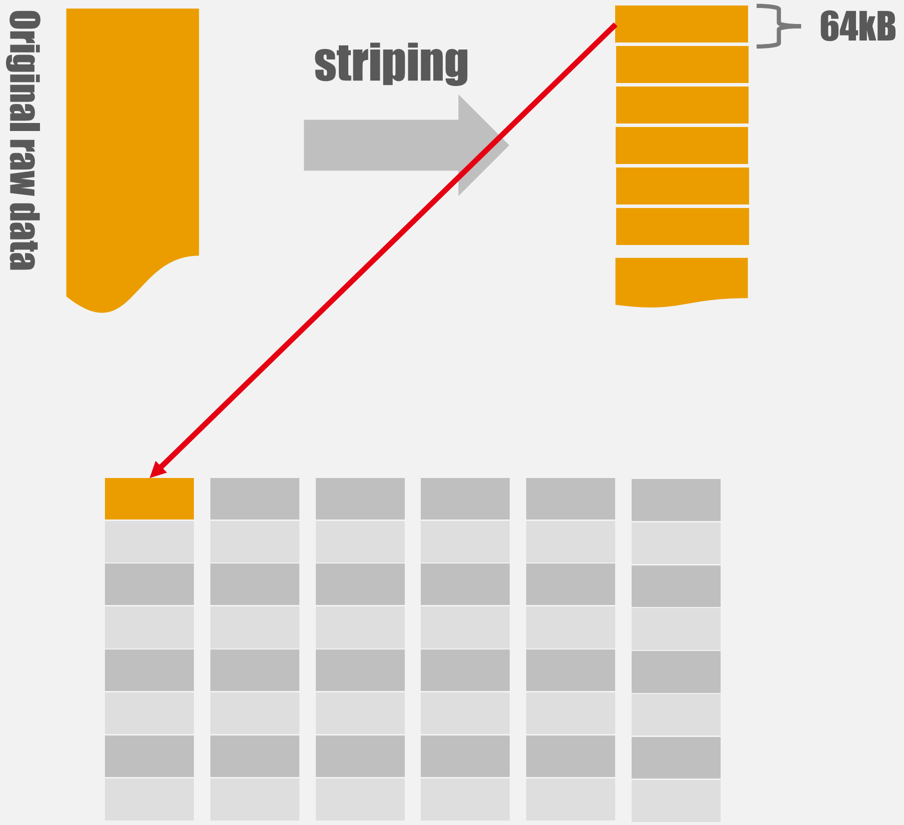
- Raw data is striped
- The basic unit of striped data is called “cell”
- The ”cell” is 64kB

Striped Block Management



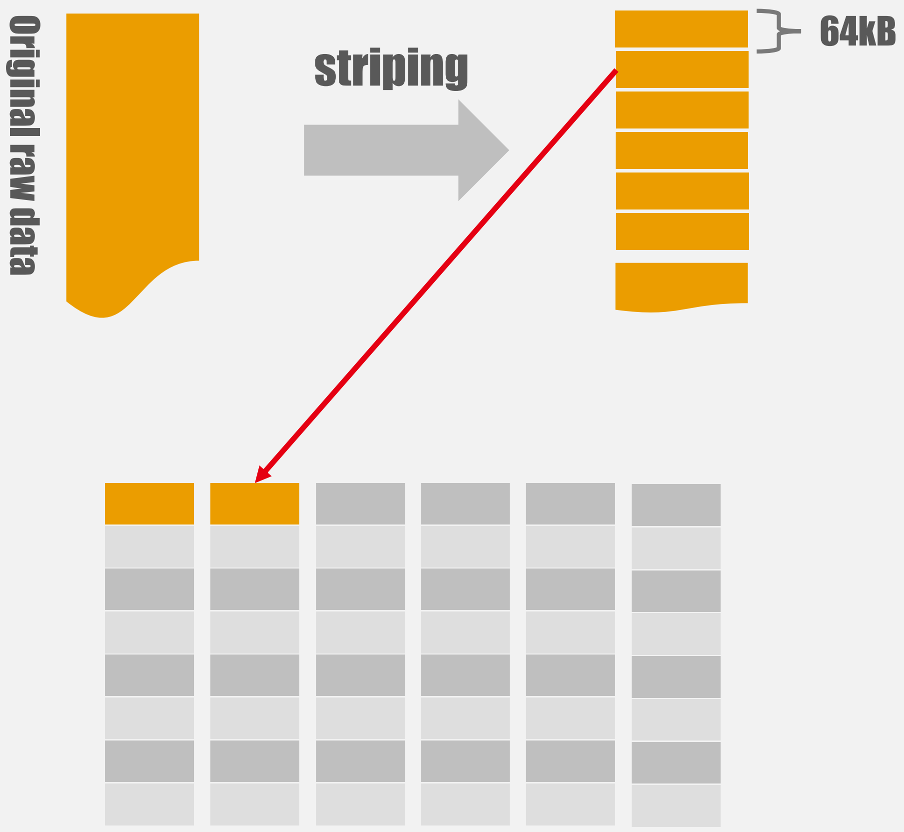
- The cells are written in blocks in order
- With striped layout

Striped Block Management



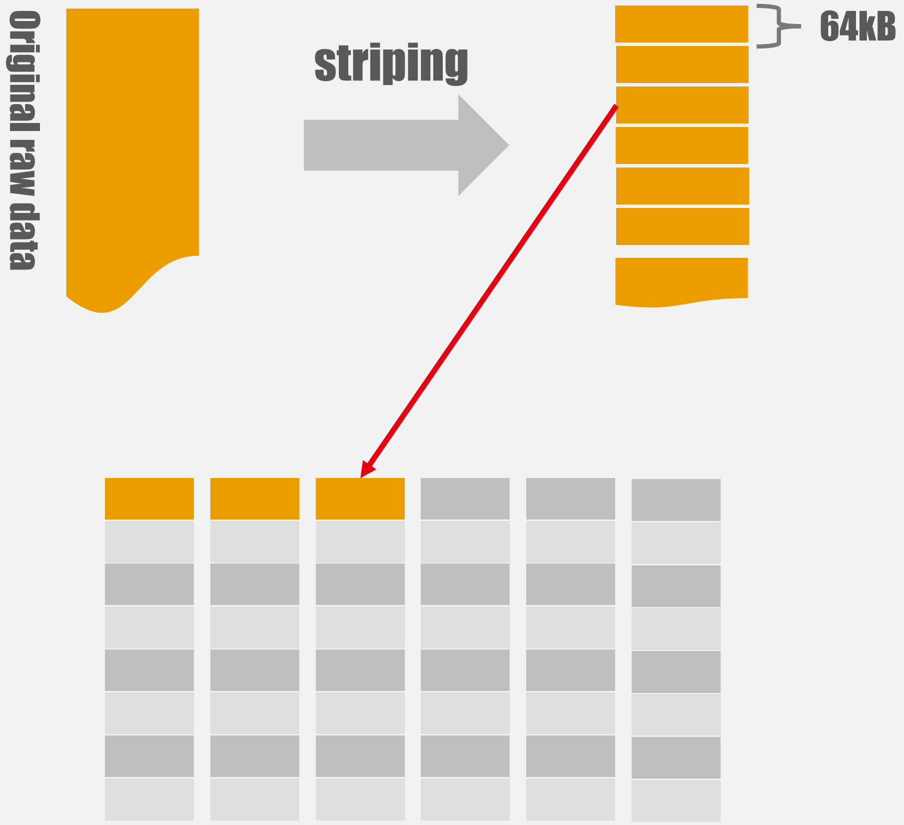
- The cells are written in blocks in order
- With striped layout

Striped Block Management



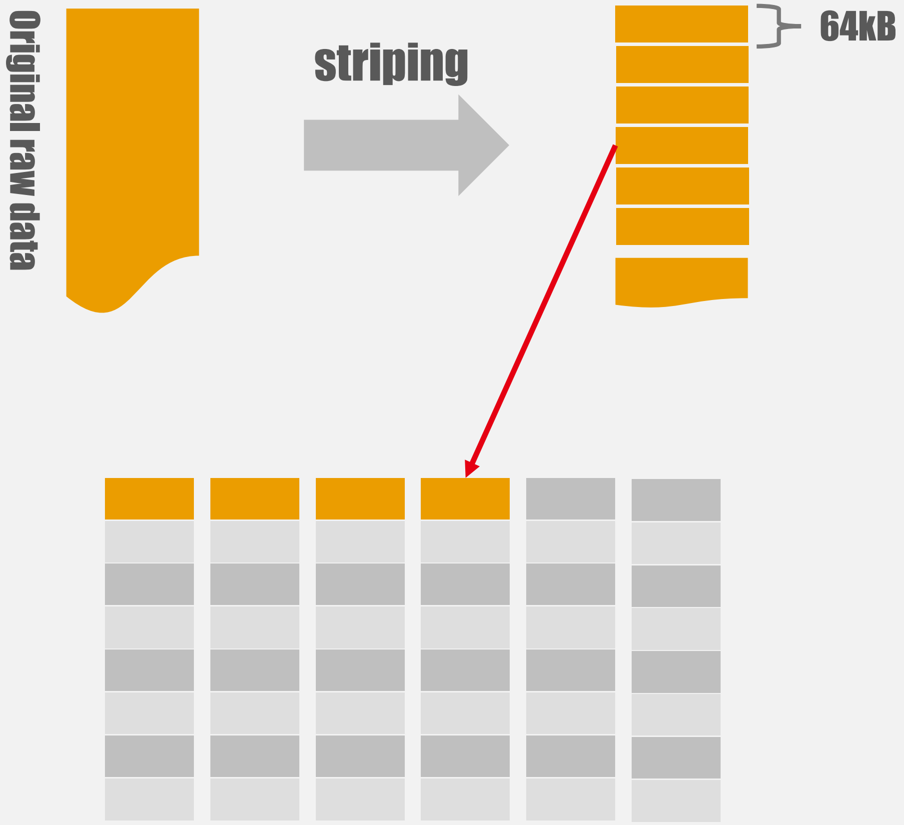
- The cells are written in blocks in order
- With striped layout

Striped Block Management



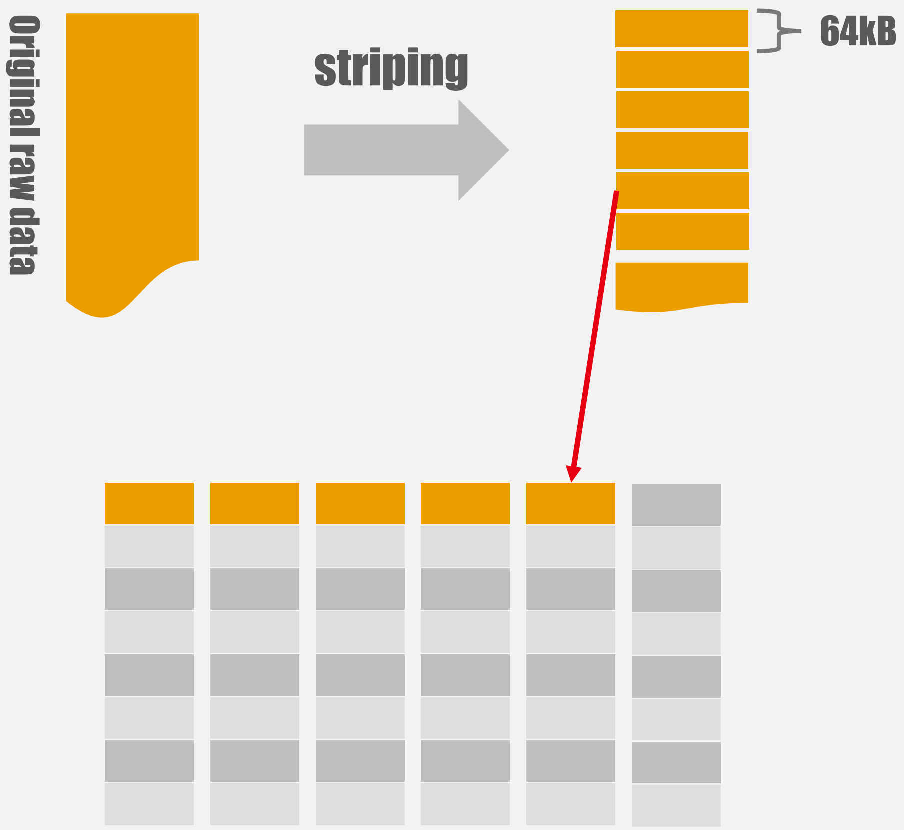
- The cells are written in blocks in order
- With striped layout

Striped Block Management



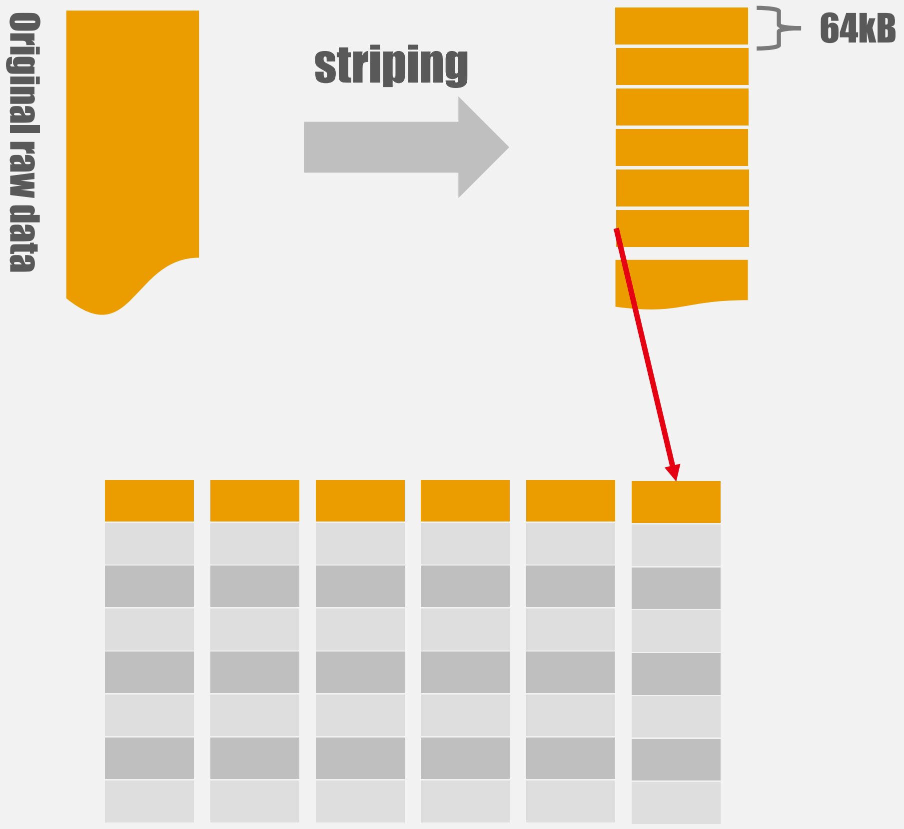
- The cells are written in blocks in order
- With striped layout

Striped Block Management



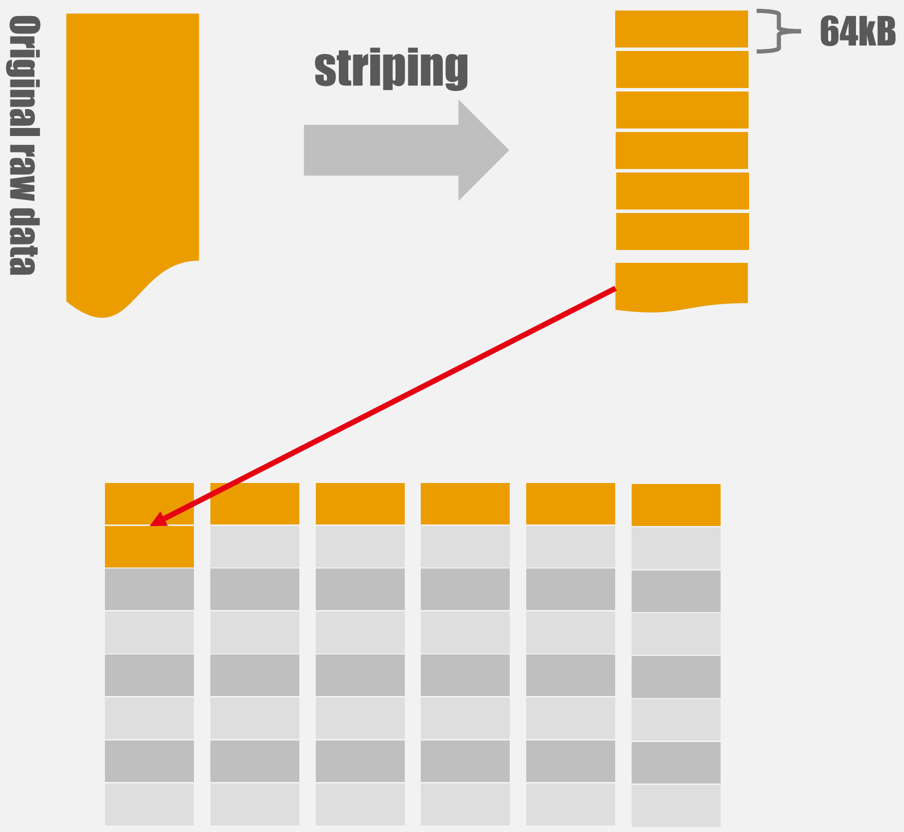
- The cells are written in blocks in order
- With striped layout

Striped Block Management



- The cells are written in blocks in order
- With striped layout

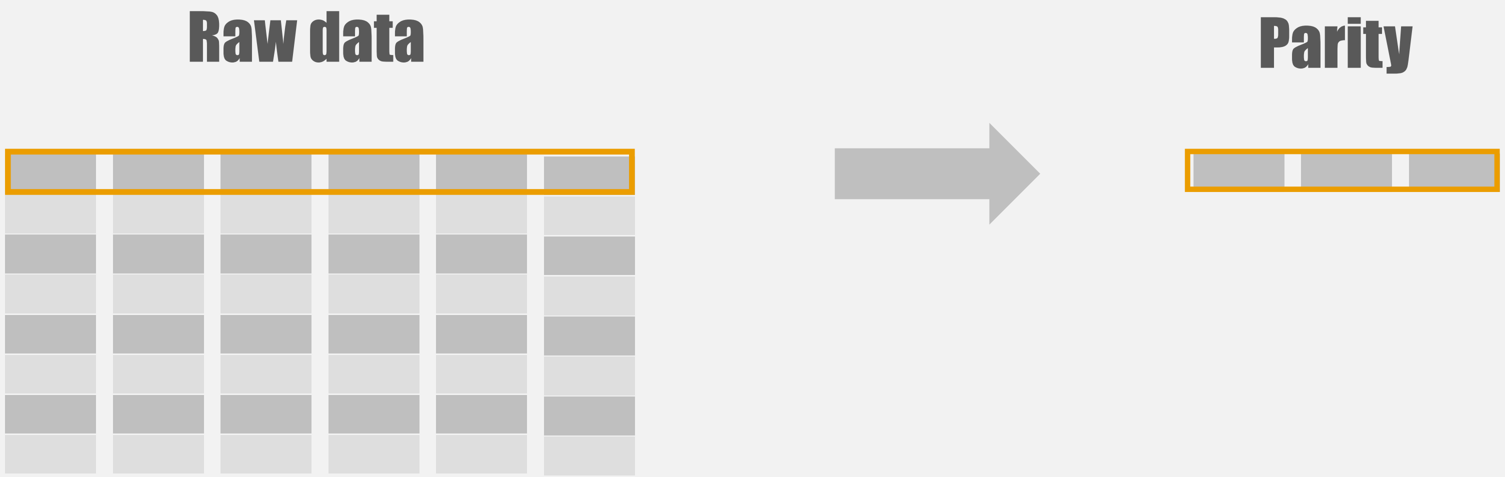
Striped Block Management



- The cells are written in blocks in order
- With striped layout

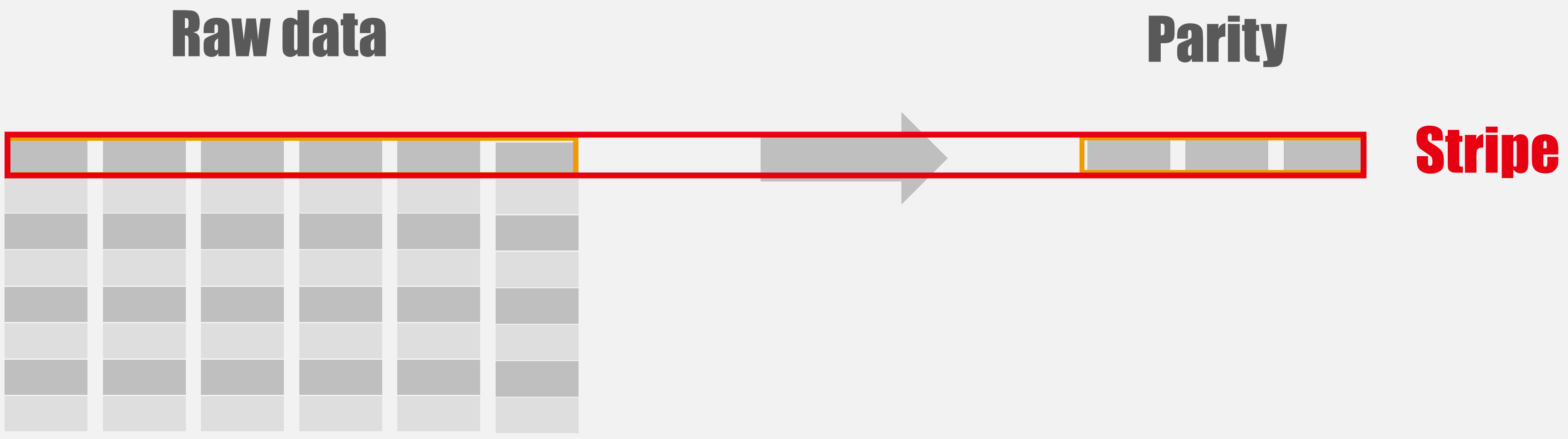
Striped Block Management

The six cells of raw data
will be used to calculate three parities



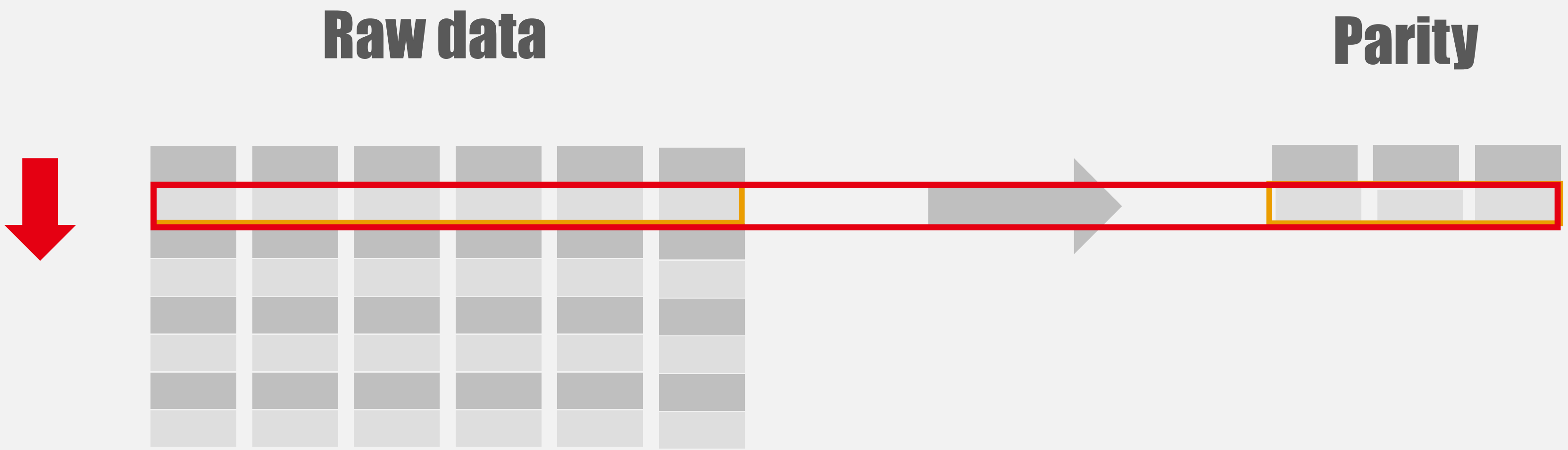
Striped Block Management

The six data cells and three parity cells are named "stripe"



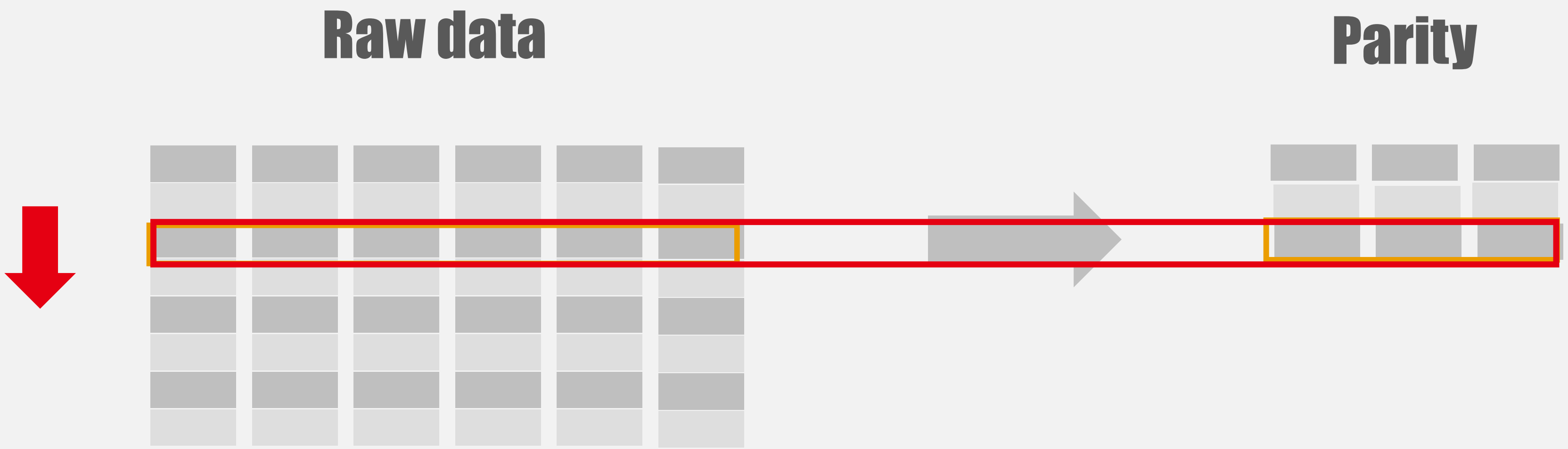
Striped Block Management

Every stripes are written in blocks in order



Striped Block Management

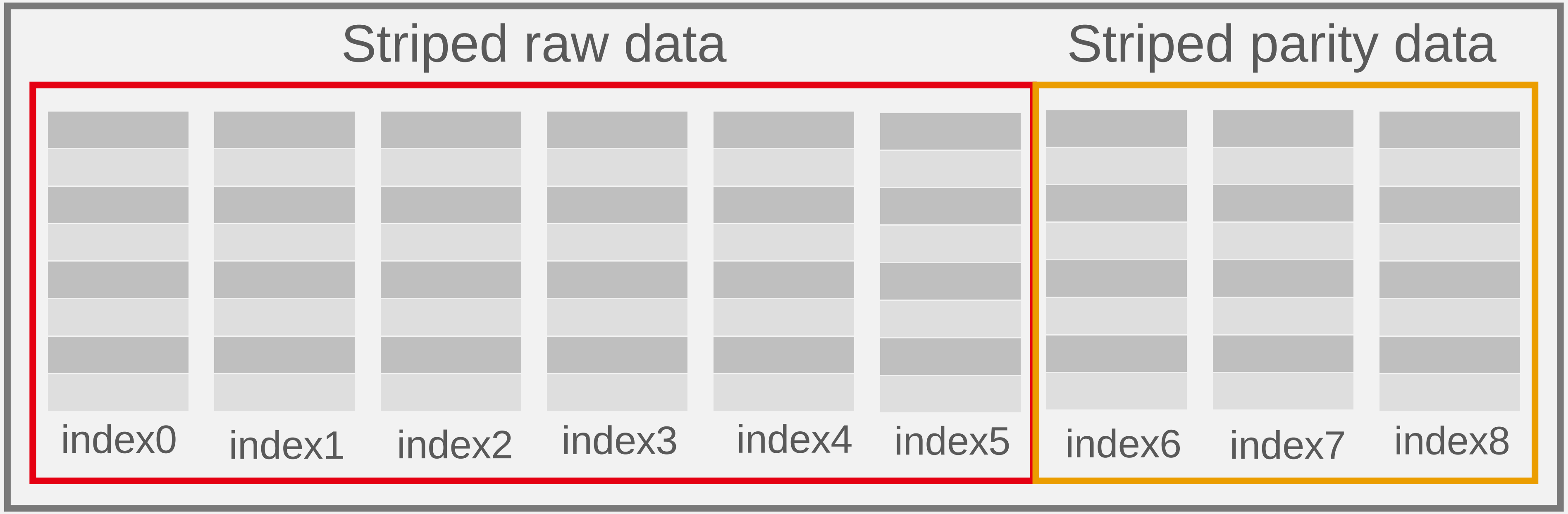
Every stripes are written in blocks in order



Striped Block Management

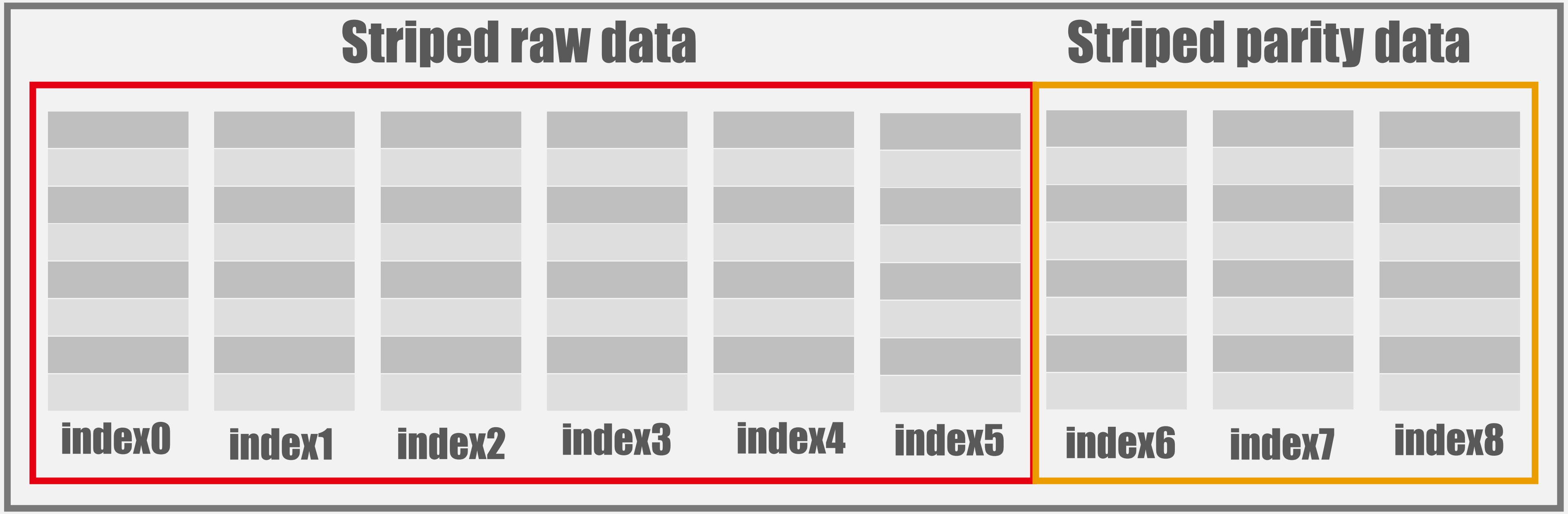
Block Group

- Combining data and parity striped blocks



Internal Blocks

- The striped blocks in Block Group are called "internal block"
- Every internal block has index



Agenda

1. **About HDFS Erasure Coding**
    - Key points
    - Implementation
    - Compare to replication
2. **HDFS Erasure Coding Tests**
    - System Tests
        - Basic Operations
        - Reconstruct Erasure Coding Blocks
        - Other features
    - Performance Tests
3. **Usage in Yahoo! JAPAN**
    - Principal workloads in our production
    - Future plan

Replication vs EC

	Replication	Erasure Coding(RS-6-3)
Target	HOT	COLD
Storage overhead	200%	50%
Data durability	✓	✓
Data Locality	✓	✕
Write performance	✓	✕
Read performance	✓	△
Recovery cost	Low	High

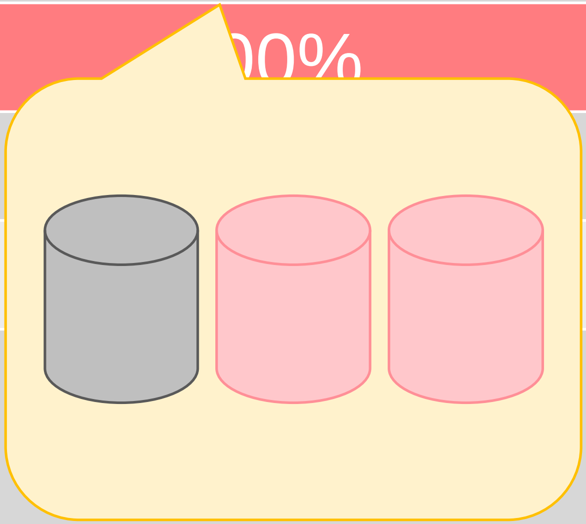
Replication vs EC

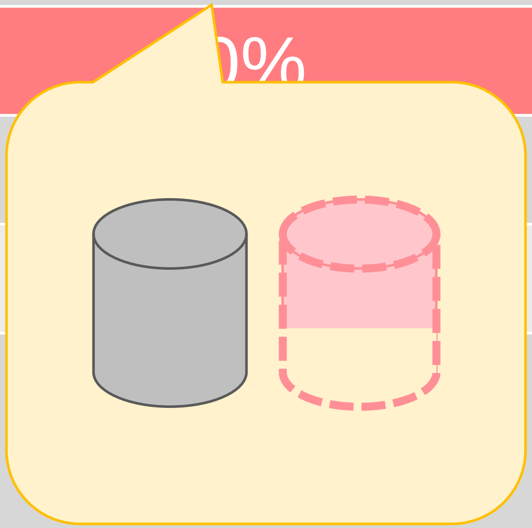
	Replication	Erasure Coding(RS-6-3)
Target	HOT	COLD
Storage overhead	200%	
Data durability	✓	
Data Locality	✓	
Write performance	✓	✕
Read performance	✓	△
Recovery cost	Low	High

It will not be modified and rarely accessed.

Replication vs EC

	Replication	Erasure Coding(RS-6-3)
Target	HOT	COLD
Storage overhead	00%	0%
Data durability		
Data Locality		
Write performance		
Read performance	✓	△
Recovery cost	Low	High





Replication vs EC

	Replication	Erasure Coding(RS-6-3)
Target	HOT	COLD
Storage overhead	200%	50%
Data durability	✓	✓
Data l		
Write perfor		
Read performance	✓	△
Recovery cost	Low	High

The tripled replication mechanism could tolerant missing 2/3 replica.
In the case of the Erasure Coding, if 3/9 of storages were failed, missing data could be reconstructed

Replication vs EC

	Replication	Erasure Coding(RS-6-3)
Target	HOT	COLD
Storage overhead	200%	50%
Data durability	✓	✓
Data Locality	✓	✗
W p		
Read performance	✓	△
Recovery cost	Low	High

The data locality would be lost by using the Erasure Coding.
However, cold data in the Erasure Coding would not be accessed frequently.

Replication vs EC

In the Erasure Coding, the calculation of the parity data will decrease the write throughput.
In the reading situation, the performance will not decrease so much.
But if some internal blocks were missing, the reading throughput would be drop down.

		6-
Targe		
Stora		
Data		
Data Locality	✓	✗
Write performance	✓	✗
Read performance	✓	△
Recovery cost	Low	High

Replication vs EC

	Replication	Erasure Coding(RS-6-3)
Target	HOT	COLD
Stora		
Data		
Data		
Write performance	✓	✗
Read performance	✓	△
Recovery cost	Low	High

In the Erasure Coding, in order to recovery the missing data, a node need to read other living raw data and parity data from remote.
And then the node reconstruct missing data.
These process will use network traffics and CPU resources.

Agenda

1. About HDFS Erasure Coding
   - Key points
   - Implementation
   - Compare to replication
2. The results of the erasure coding testing
   - System tests
   - Performance tests
3. Usage in our production
   - Principal workloads in our production
   - Future plan

HDFS EC Tests

- **System Tests**
  - Basic operations
  - Reconstruct EC blocks
  - Decommission DataNodes
  - Other Features

HDFS EC Tests

- System Tests
  - Basic operations
  - Reconstruct EC blocks
  - Decommission DataNodes
  - Other Features

Basic Operations

"hdfs erasurecode –setPolicy"

- Target
  - Only directory
  - Must be empty
  - Sub-directory and files inherit policy
- Superuser privilege needed
- Default policy: Reed Solomon(6,3)

```
$ sudo -u hdfs hdfs erasurecode -setPolicy /test/ec
EC policy set successfully at hdfs://hdpsecbha/test/ec
```

Basic Operations

To confirm whether the directory has the Erasure Coding policy

"hdfs erasurecode –getPolicy"

- Show the information about the codec and cell size

```
$ sudo -u hdfs hdfs erasurecode -getPolicy /test/ec
ErasureCodingPolicy=[Name=RS-6-3-64k, Schema=[ECSchema=[Codec=rs, numDataUnits=6, numParityUnits=3]], CellSize=65536 ]
```

File Operations

After setting EC policy,

Basic file operations are conducted against EC files

- File format transparent to HDFS clients
- Write/read datanode failure tolerant

Move operation is a little different.

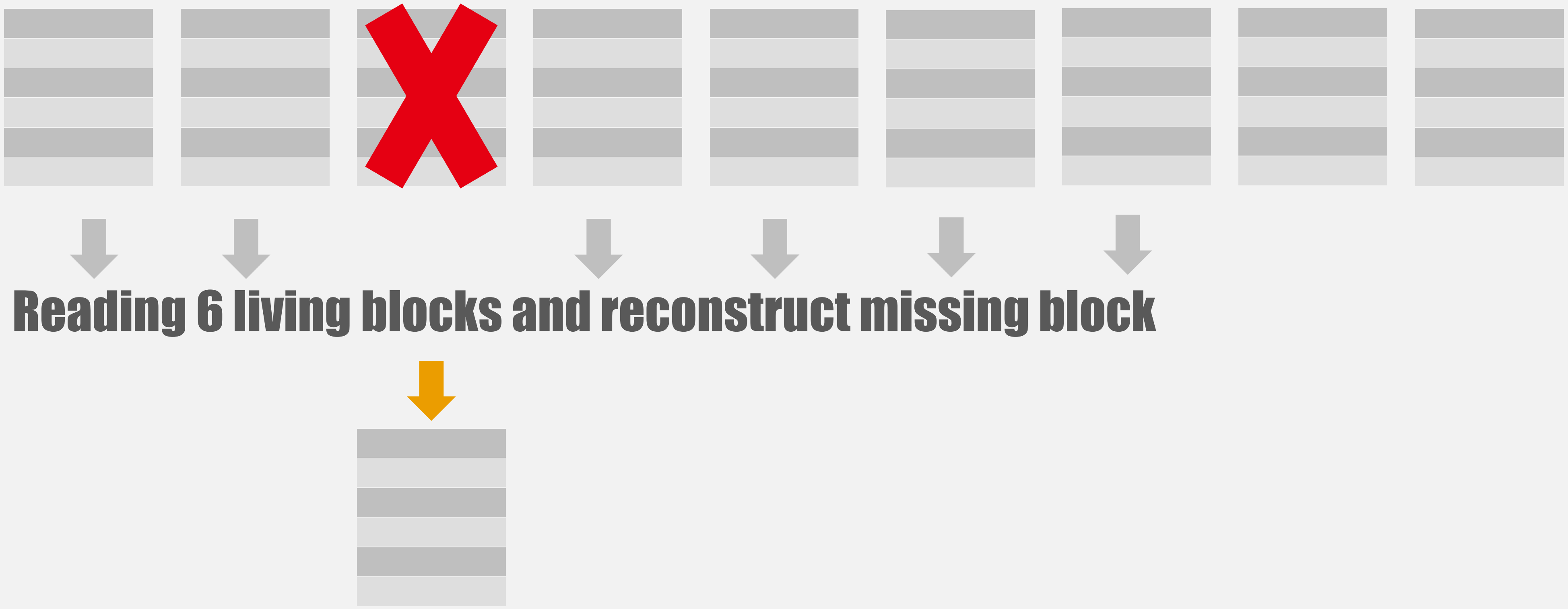
- File format not automatically changed by move operation.

HDFS EC Tests

- **System Tests**
  - Basic operations
  - Reconstruct EC blocks
  - Decommission DataNodes
  - Other Features

Reconstruct EC Blocks

The missing blocks can be reconstructed with at least 6 living internal blocks.



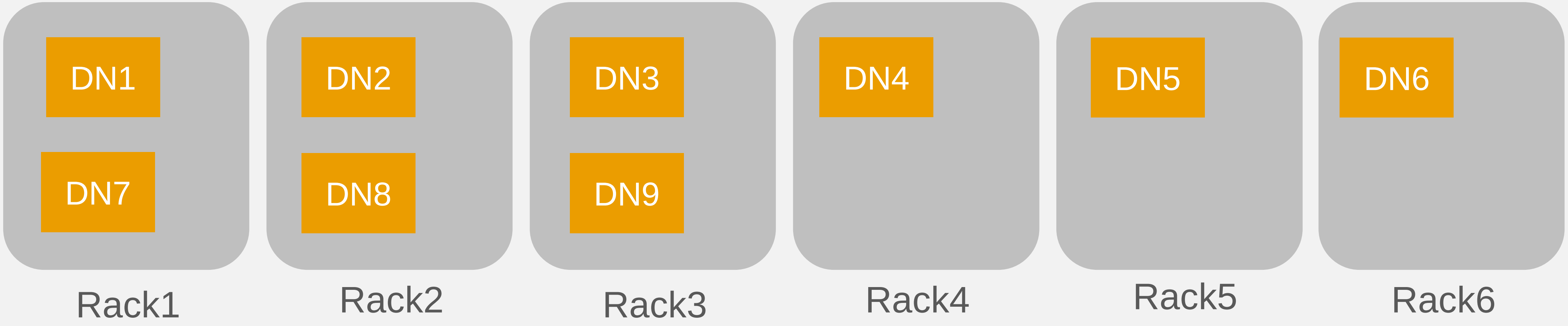
Reconstruct EC Blocks

- **The cost of the reconstruction is irrelevant with missing internal block count**
  - No matter one or three internal blocks are missing, the reconstruction costs are the same
  - block groups with more missing internal blocks has higher priority

Rack Fault Tolerance

BlockPlacementPolicyRackFaultTolerant

- A new block placement policy
- Choses the storages to distributed blocks to racks as many as possible



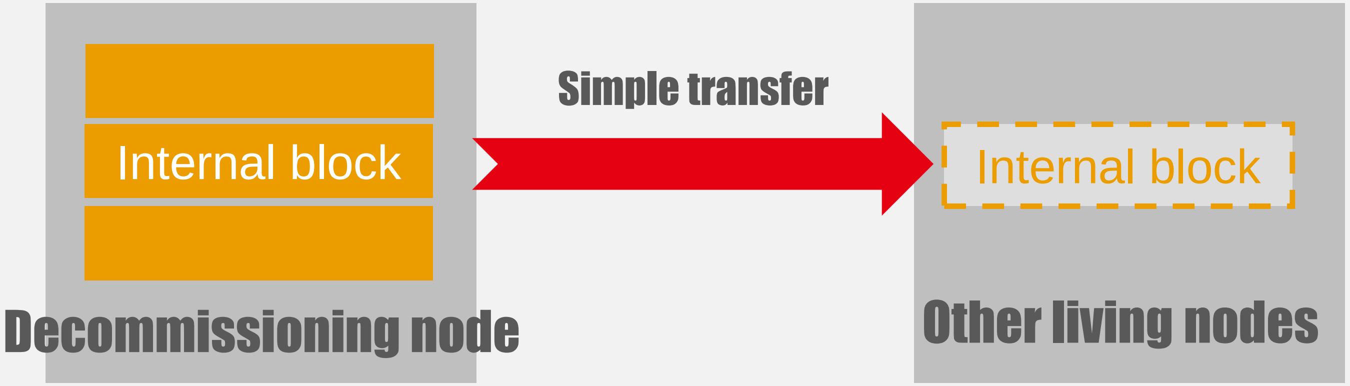
HDFS EC Tests

- **System Tests**
  - Basic operations
  - Reconstruct EC blocks
  - Decommission DataNodes
  - Other Features

Decommission DNs

Decommission is basically same as recovery blocks

But, transfer blocks to another DataNode is enough in Erasure Coding.



HDFS EC Tests

- **System Tests**
  - Basic operations
  - Reconstruct EC blocks
  - Decommission DataNodes
  - Other Features

Supported

- NameNode HA
- Quotation Configuration
- HDFS File System Check

```
Erasure Coded Block Groups:
 Total size:    15361641564186 B (Total open files size: 8309833728 B)
 Total files:   8239 (Files currently being written: 108)
 Total block groups (validated):        24819 (avg. block group size 618946837 B)
(Total open file block groups (not validated): 118)
 Minimally erasure-coded block groups:  24819 (100.0 %)
 Over-erasure-coded block groups:       4 (0.016116684 %)
 Under-erasure-coded block groups:      109 (0.43917966 %)
 Unsatisfactory placement block groups: 0 (0.0 %)
 Default ecPolicy:              RS-6-3-64k
 Average block group size:      8.176316
 Missing block groups:          0
 Corrupt block groups:          0
 Missing internal blocks:       109 (0.053686384 %)
FSCK ended at Mon Aug 01 18:07:10 JST 2016 in 215 milliseconds
```

Unsupported

- Flush and Synchronize(hflush/hsync)
- Append to EC files
- Truncate EC files

These features are not supported.

However, those are not so critical, because the target of HDFS erasure coding is storing cold data.

Agenda

1. About HDFS Erasure Coding
   - Key points
   - Implementation
   - Compare to replication
2. The results of the erasure coding testing
   - System tests
   - Performance tests
3. Usage in our production
   - Principal workloads in our production
   - Future plan

HDFS EC Tests

- **Performance Tests**
  - Writing/Reading Throughput
  - TeraGen/TeraSort
  - Distcp

Cluster Information

Alpha

- 37 Nodes
- 28 cores CPU * 2
- 256GB RAM
- SATA 4TB * 12 Disks
- Network 10Gbps
- One Rack

Beta

- 82 Nodes
- 28 cores CPU * 2
- 128GB RAM
- SATA 4TB * 15 Disks
- Network 10Gbps
- Five Racks

HDFS EC Tests

- Performance Tests
  - Writing/Reading Throughput
  - TeraGen/TeraSort
  - Distcp

Read/Write Throughput

ErasureCodingBenchmarkThroughput

- Write and read files on the replication and erasure coding formats with multithreads
- In Erasure Coding, writing throughput was about 65% of replication’s
- Reading throughput decreased slightly in Erasure Coding

	Replication	Erasure Coding
Write	111.3 MB/s	73.94 MB/s
Read(stateful)	111.3 MB/s	111.3 MB/s
Read(positional)	111.3 MB/s	107.79 MB/s

Read With Missing Internal Blocks

The throughput decreased when internal blocks was missing

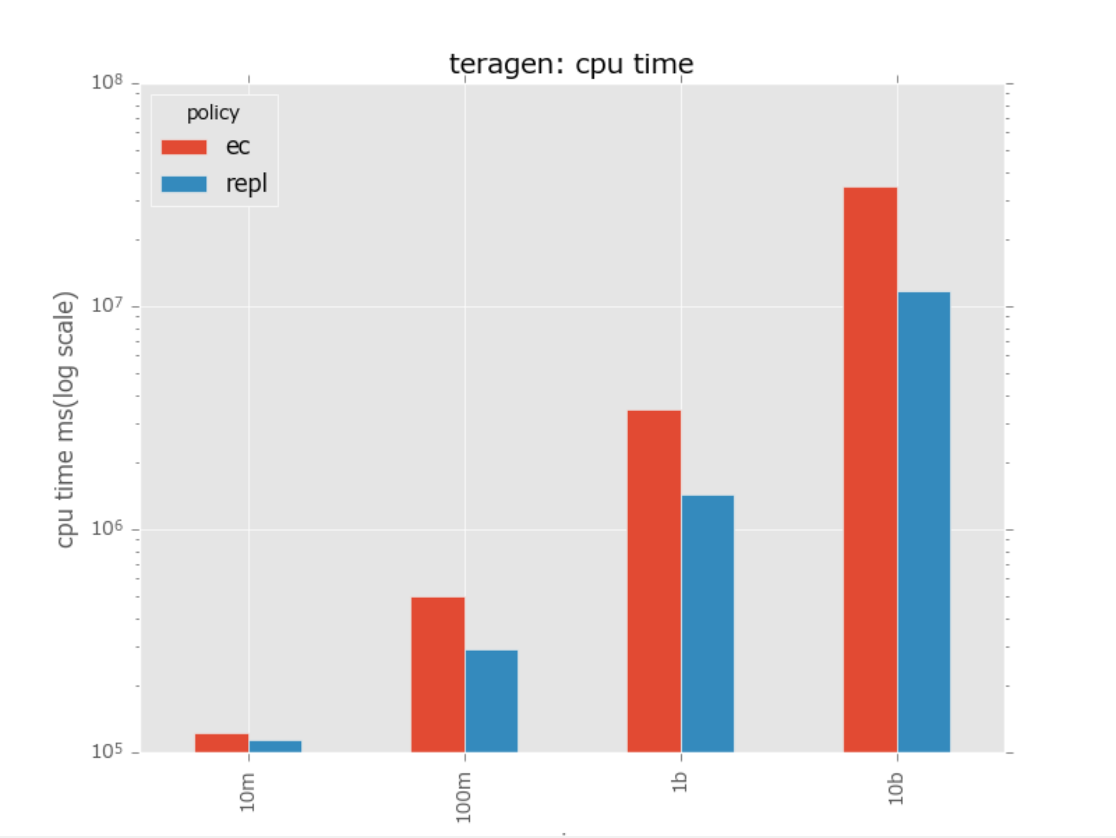
Because the client needed to reconstruct missing blocks

	All blocks living	An internal block missing
Read(stateful)	111.3 MB/s	108.94 MB/s
Read(positional)	107.79 MB/s	92.25 MB/s

HDFS EC Tests

- **Performance Tests**
  - Writing/Reading Throughput
  - TeraGen/TeraSort
  - Distcp

CPU Time of TeraGen



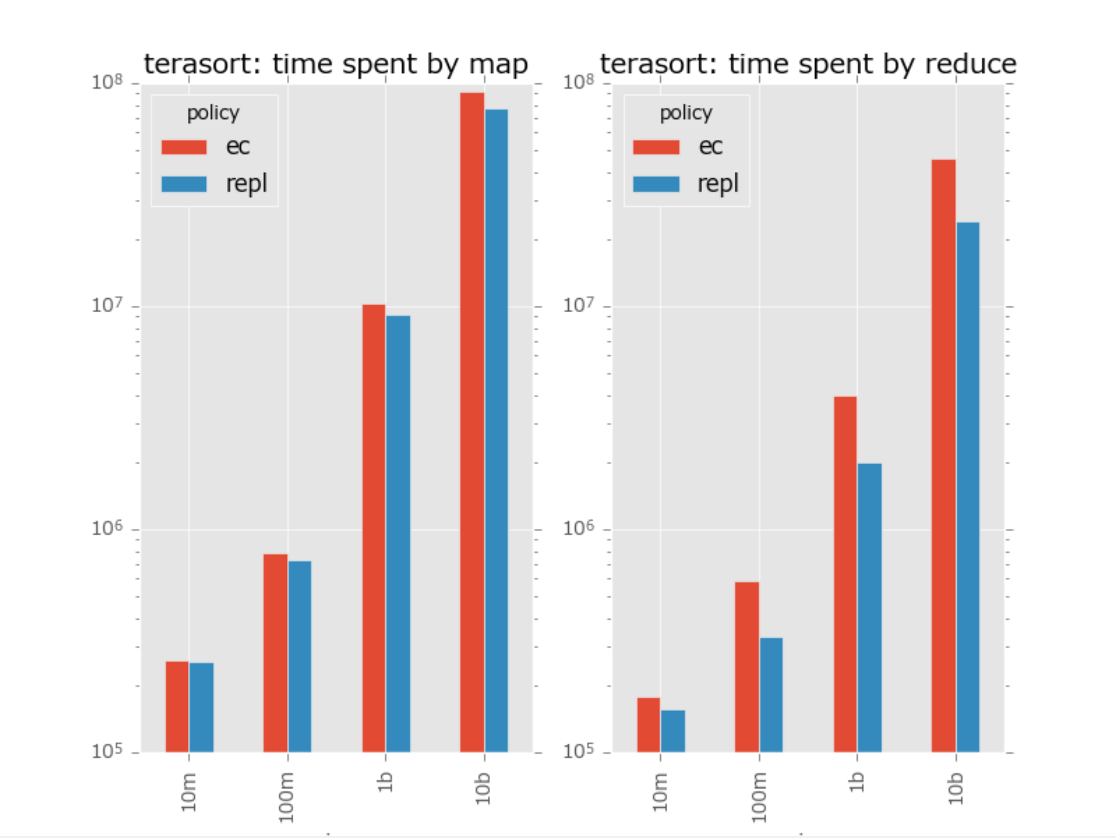
X-axis: number of rows of outputs

Y-axis: log scaled CPU times(ms)

The CPU time increased in the erasure coding format.

The overhead of the calculate parity data affected writing performance.

TeraSort Map/Reduce Time Cost



Total time spent by map(left) and reduce(right)

X–axis: number of rows of outputs

Y–axis: log scaled CPU times(ms)

The times spent by the reduce tasks increased significantly in the erasure coding.

Because the main workload of the reduce was writing.

HDFS EC Tests

- **Performance Tests**
  - Writing/Reading Throughput
  - TeraGen/TeraSort
  - Distcp

Elapsed Time of Distcp

- Our real log data(2TB)
- Copying replication to EC was tripled of copying replication to replication

- Currently using distcp is the best way to convert to Erasure Coding

	Real elapsed time	Cpu time
Replication to EC	17mins, 11sec	64,754,291ms
Replication to Replication	5mins, 5sec	20,187,156ms

Agenda

1. About HDFS Erasure Coding
    - Key points
    - Implementation
    - Compare to replication
2. HDFS Erasure Coding Tests
    - System Tests
        - Basic Operations
        - Reconstruct Erasure Coding Blocks
        - Other features
    - Performance Tests
3. Usage in Yahoo! JAPAN
    - Principal workloads in our production
    - Future plan

Target

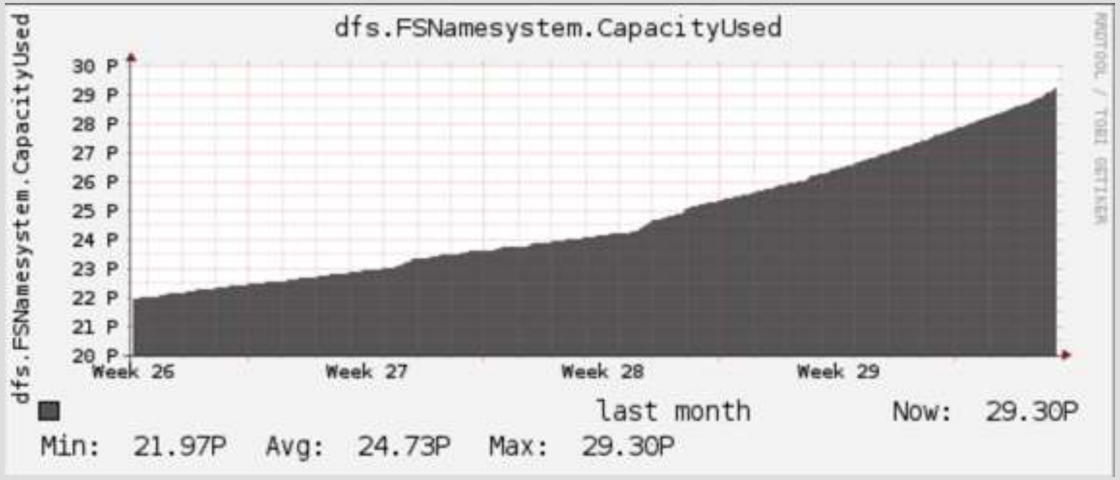
Raw data of weblogs

Daily total 6.2TB

Up to 400 days

The capacity used of our production HDFS.

We need to reduce storage space cost.

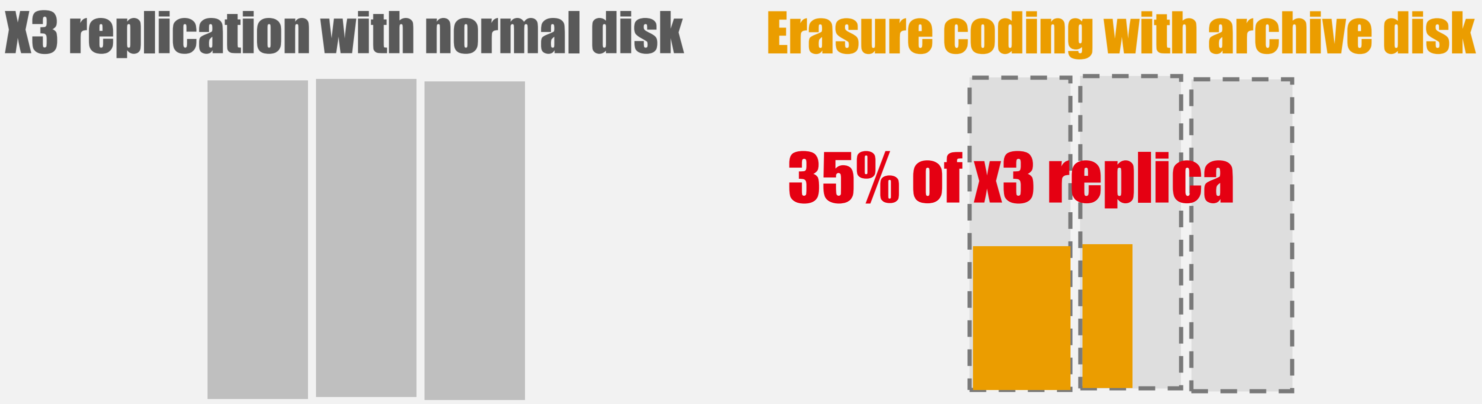


EC with Archive Disk

We are using Erasure Coding with archive storages

The cost of archive disk is 70% of normal disk.

In total, the storage cost of EC with archive disk could be reduced to 35% of x3 replication with normal disk.

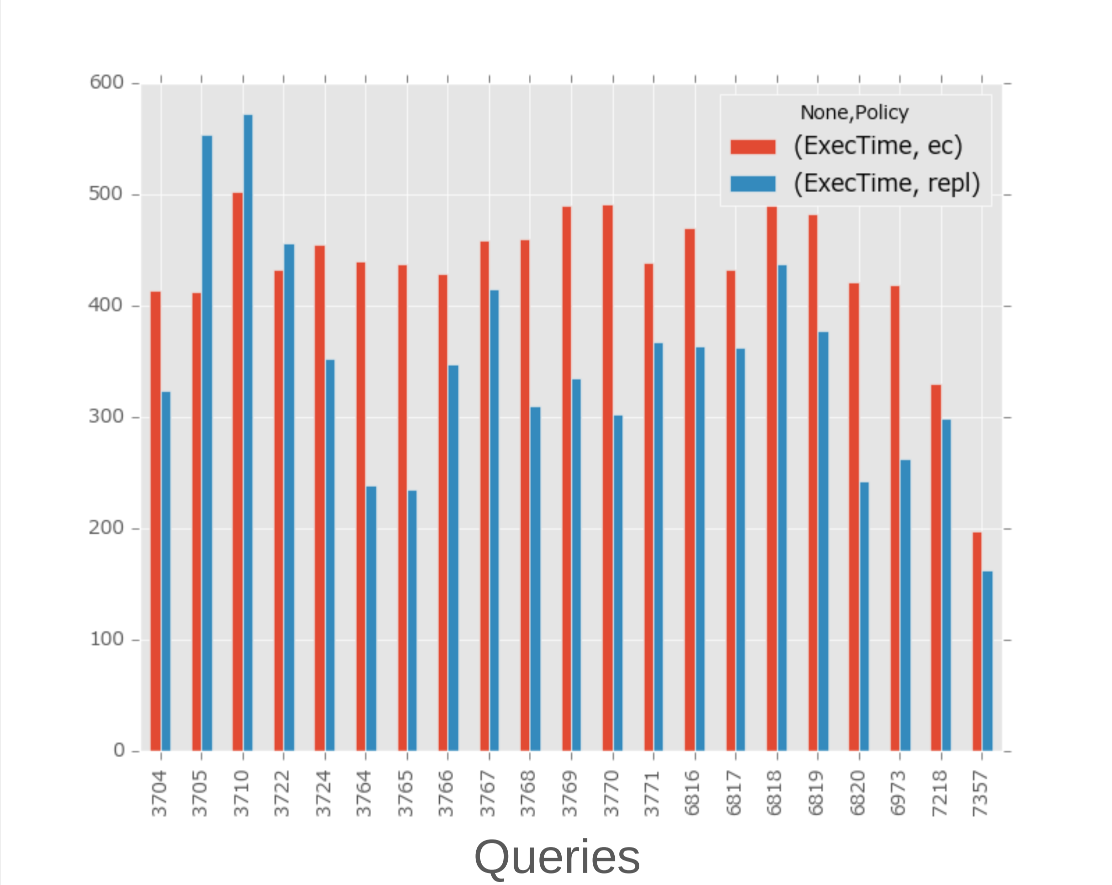


EC Files as Hive Input

Erasure Coding ORC files

2TB, 17billion records

- The query execution time seemed to increase when the input data is erasure coded files
- But it will be acceptable, considering the queries are rarely executed



Agenda

1. About HDFS Erasure Coding
    - Key points
    - Implementation
    - Compare to replication
2. HDFS Erasure Coding Tests
    - System Tests
        - Basic Operations
        - Reconstruct Erasure Coding Blocks
        - Other features
    - Performance Tests
3. Usage in Yahoo! JAPAN
    - Principal workloads in our production
    - Future plan

Future Phase of HDFS EC

- **Codecs**
  - Intel ISA-L
  - Hitchhiker algorithm
- **Contiguous layout**
  - To provide data locality
- **Implement hflush/hsync**

If they were implemented, the erasure coding format would be used in much more scenarios

Conclusion

- **HDFS Erasure Coding**
  - The target is storing cold data
  - It reduces half storage costs without sacrificing data durability
  - It’s ready for production

Thanks for Listening!