elasticsearch
for developer

Bulk API

book_03.txt

การจัดการข้อมูล

- Retrieve multiple documents

- Bulk API

- Dealing with conflict

Retrieve multiple documents

GET /store/book/**_mget**

```
{
  "docs": [
    {
        "_id": 1,
        "_source": [ "title","views"]
    },
    {
        "_id": 2,
        "_source": "title"
    }]}
```

Retrieve multiple documents

ผลการทำงาน

```json
{
    "docs": [
        {
            "_index": "store",
            "_type": "book",
            "_id": "1",
            "_version": 16,
            "found": true,
            "_source": {
                "title": "partial update",
                "views": 4
            }
        },
        {
            "_index": "store",
            "_type": "book",
            "_id": "2",
            "_version": 5,
            "found": true,
            "_source": {
                "title": "Clean Code: A Handbook of Agile Software Craftsmanship"
            }
        }
    ]
}
```

Retrieve multiple documents
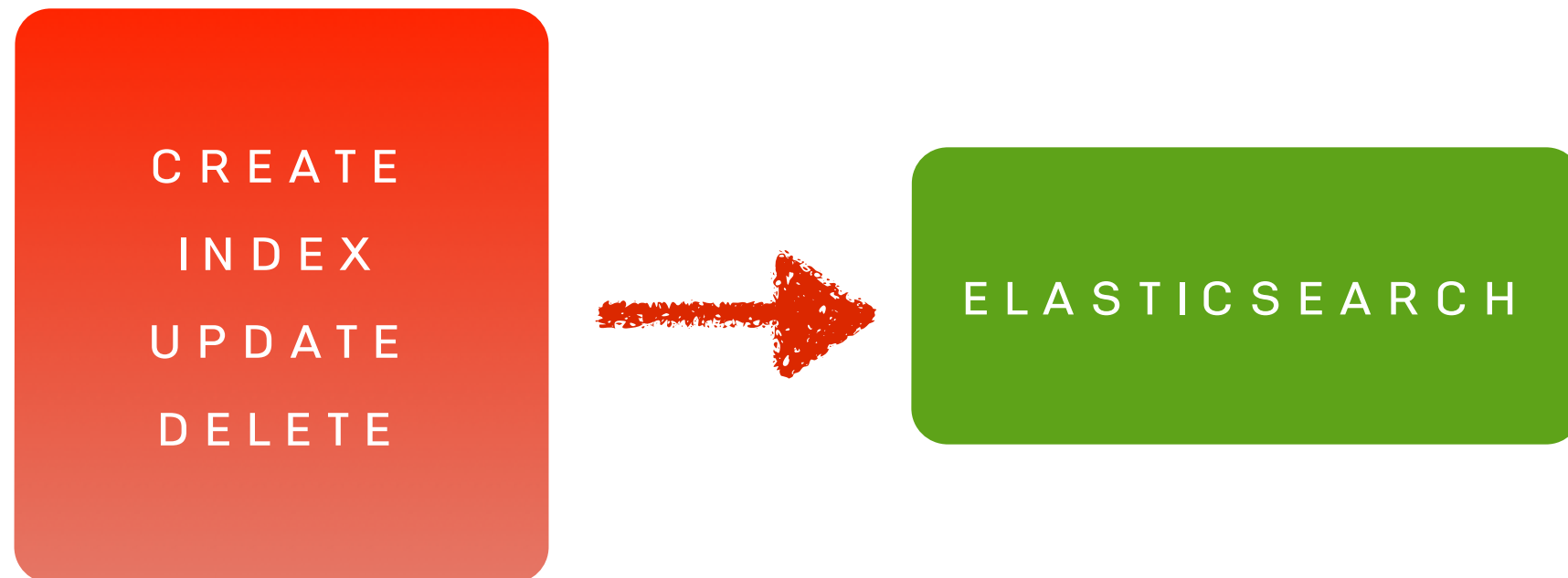
GET /store/book/**_mget**

```
{

  "ids": [1,2]

}
```

Cheaper with Bulk

- create -> สร้าง document ถ้าไม่มี

- index -> สร้าง document หรือ replace

- update -> partial update

- delete -> ลบ

Cheaper with Bulk

All request in one step

CREATE
INDEX
UPDATE
DELETE

→

ELASTICSEARCH

Create document

POST /_bulk

```
{"create":{"_index":"store","_type":"book","_id":"1001"}}

{"title":"new book 1000", "description":"my new book"}
```

Auto ID

POST /_bulk

```
{"index":{"_index":"store","_type":"book"}}
{"title":"new book xxx", "description":"my new book with auto id"}
```

Put it all together

POST /_bulk

```
{ "delete": { "_index": "store", "_type": "book", "_id": "1001" }}

{ "create": { "_index": "store", "_type": "book", "_id": "1001" }}

{ "title":    "My new book" }

{ "index":  { "_index": "store", "_type": "book" }}

{ "title":    "My second book" }

{ "update": { "_index": "store", "_type": "book", "_id": "1001",

"_retry_on_conflict" : 3} }

{ "doc" : {"title" : "My updated book"} }
```

Put it all together

POST /store/book/_bulk

```
{ "delete": { "_id": "1001" }}

{ "create": { "_id": "1001" }}

{ "title":    "My new book" }

{ "index":  {  }}

{ "title":    "My second book" }

{ "update": { "_id": "1001", "_retry_on_conflict" : 3} }

{ "doc" : {"title" : "My updated book"} }
```

Put it all together

Result success :: errors=false

```
{
    "took": 3,
    "errors": false,
    "items": [
        {
            "delete": {
                "_index": "store",
                "_type": "book",
                "_id": "1001",
                "_version": 26,
                "status": 200,
                "found": true
            }
        },
```
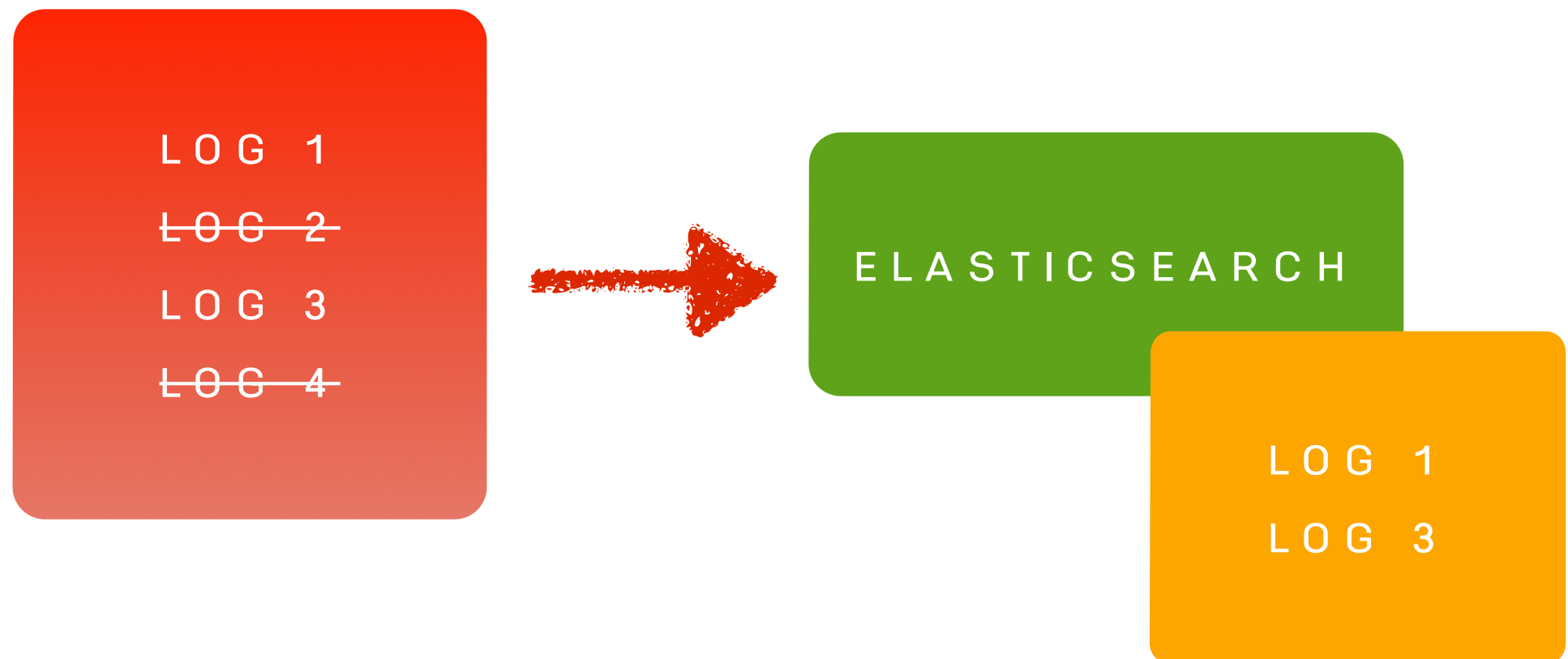
Put it all together

Result failure :: errors=true

```json
{
    "took": 3,
    "errors": true,
    "items": [
        {
            "create": {
                "_index": "store",
                "_type": "book",
                "_id": "1001",
                "status": 409,
                "error": "DocumentAlreadyExistsException[[store][3]
[book][1001]: document already exists]"
            }
        },
```

Add logs with bulk api

POST /store/log/_bulk

```
{ "create": { "_id": "1" }}

{ "code":"200", "message":"success" }

{ "update": { "_id": "2", "_retry_on_conflict" : 3}}

{ "doc" : {"code" : "404", "message" : "not found"}}

{ "create": { "_id": "3" }}

{ "code":"200", "message":"success" }

{ "update": { "_id": "4", "_retry_on_conflict" : 3}}

{ "doc" : {"code" : "404", "message" : "not found"}}
```

Result

```json
{
    "took": 5,
    "errors": true,
    "items": [
        {
            "create": {
                "_index": "store",
                "_type": "log",
                "_id": "1",
                "_version": 1,
                "status": 201
            }
        },
        {
            "update": {
                "_index": "store",
                "_type": "log",
                "_id": "2",
                "status": 404,
                "error": "DocumentMissingException[[store][-1] [log][2]: document missing]"
            }
        },
        {
```

Result

```json
{
    "create": {
        "_index": "store",
        "_type": "log",
        "_id": "3",
        "_version": 1,
        "status": 201
    }
},
{
    "update": {
        "_index": "store",
        "_type": "log",
        "_id": "4",
        "status": 404,
        "error": "DocumentMissingException[[store][-1] [log][4]: document missing]"
    }
}
]
}
```
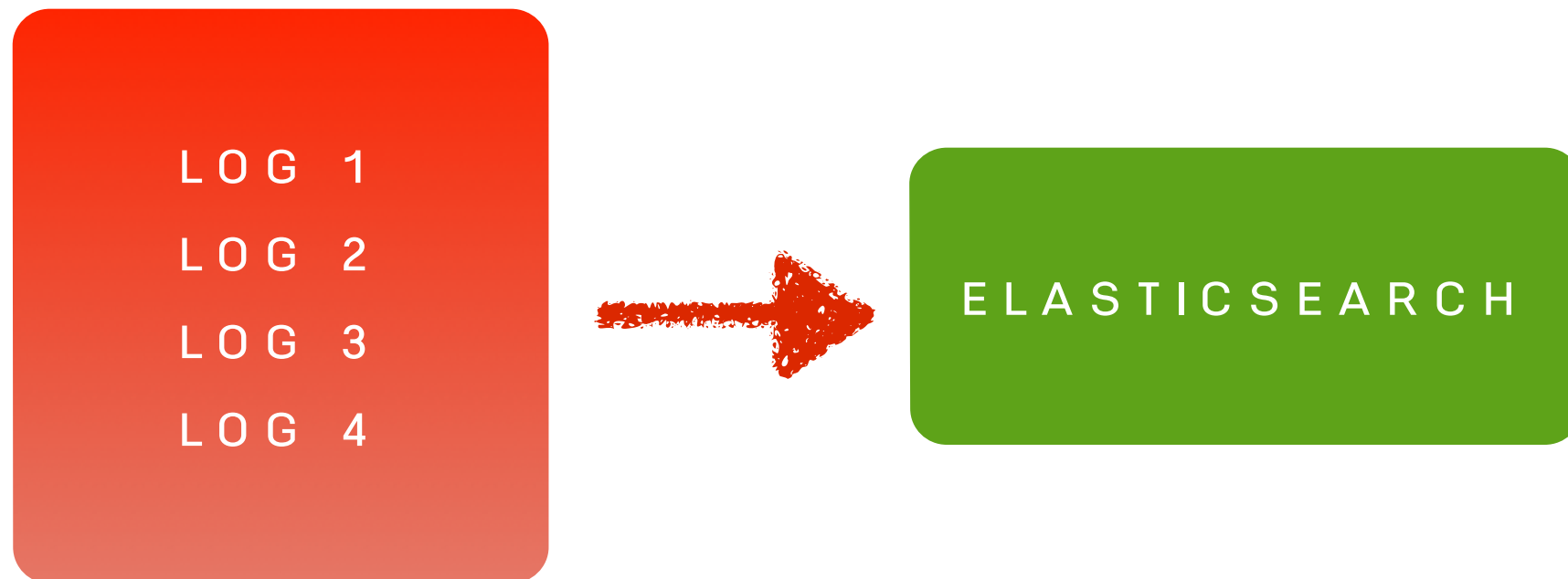
How big of data in bulk ?

????

LOG 1
LOG 2
LOG 3
LOG 4

ELASTICSEARCH

How big of data in bulk ?

ตรงปรับให้เหมาะสมแต่ละ use case

How big of data in bulk ?

5-15 MB คือ คำแนะนำ

LOG 1
LOG 2
LOG 3
LOG 4

ALL DATA IN MEMORY

ELASTICSEARCH