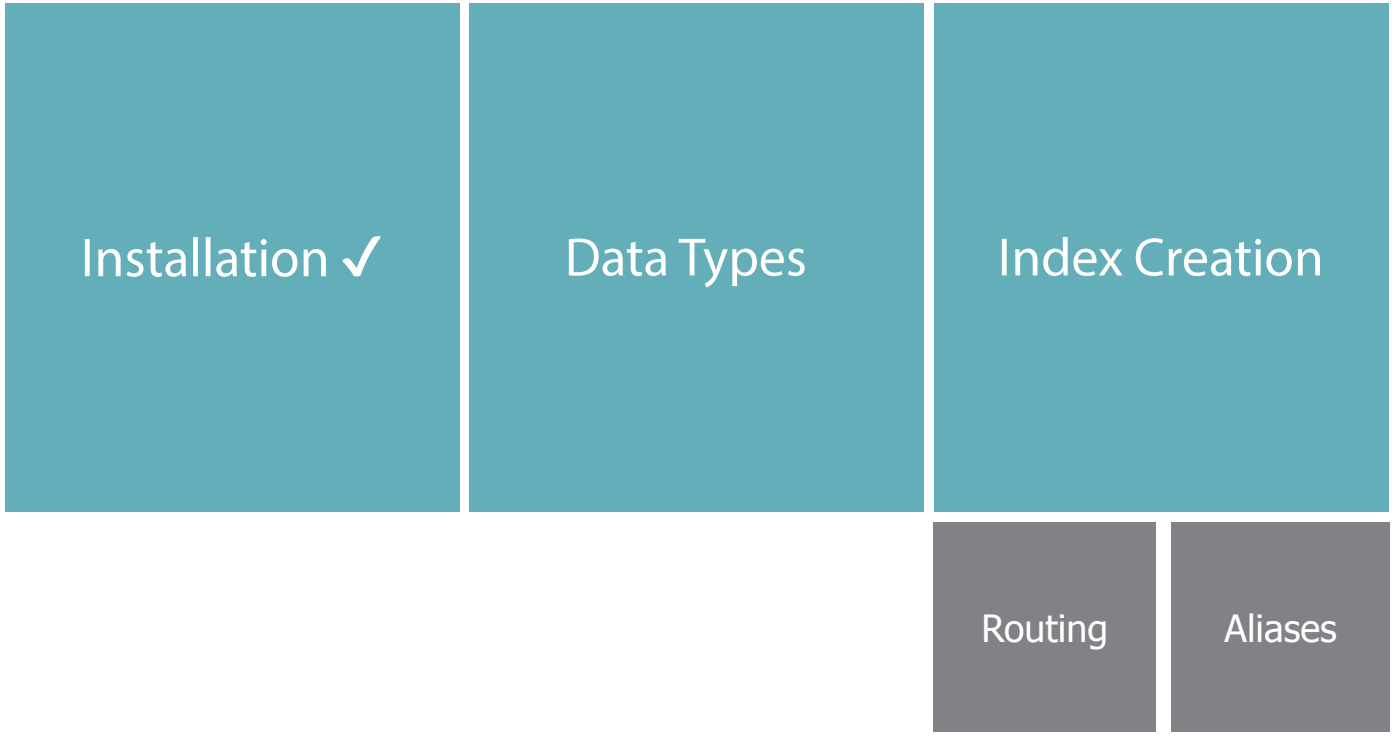
Data Types & Index Creation

JP Toto
http://jptoto.jp
@jptoto

pluralsight
hardcore dev and IT training

Let's talk about Indexes



Elasticsearch Schemas (mappings)

```
(POST) http://localhost:9200/my_blog

{
    "mappings": {
        "post": {
            "properties": {
                "user_id": {
                    "type": "integer"
                },
                "post_text": {
                    "type": "string"
                },
                "post_date": {
                    "type": "date"
                }
            }
        }
    }
}
```

Elasticsearch Data Types

- **String**
- **Numbers**
- **Boolean**
- **Date**
- **Binary**

Elasticsearch Data Types

String

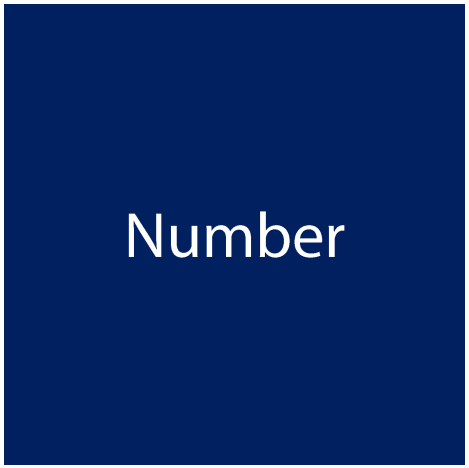
- Most common type. Simlar to varchar in sql.
- Use it to store any collection of characters.
- Many options and settings for flexibility.

Elasticsearch Data Types

Boolean

- **Pretty basic. Either ‘true’ or ‘false’.**

Elasticsearch Data Types



- **Store any type of number for searching.**
- **Float, double, byte, short, integer, long.**

Elasticsearch Data Types

Date

- Dates and times.
- UTC by default.
- Many formatting options, even many at once.

Elasticsearch Data Types

Binary

- Good for image or blob data.
- Stored in base64 strings.
- Not index by default.

Data Types Options

```
{
    "mappings": {
        "post": {
            "properties": {
                "user_id": {
                    "type": "integer",
                },
                "post_text": {
                    "type": "string"
                },
                "post_date": {
                    "type": "date",
                    "format":"YYYY-MM-DD"
                }
            }
        }
    }
}
```

Creating An Index

```
{
    "settings": {
        "index": {
            "number_of_shards": 5
        }
    },
    "mappings": {
        "post": {
           "properties": {
                "user_id": {
                    "type": "integer"
                },
                "post_text": {
                    "type": "string"
                },
                "post_date": {
                    "type": "date",
                    "format":"YYYY-MM-DD"
                }
            }
        }
    }
}
```

Mappings _source

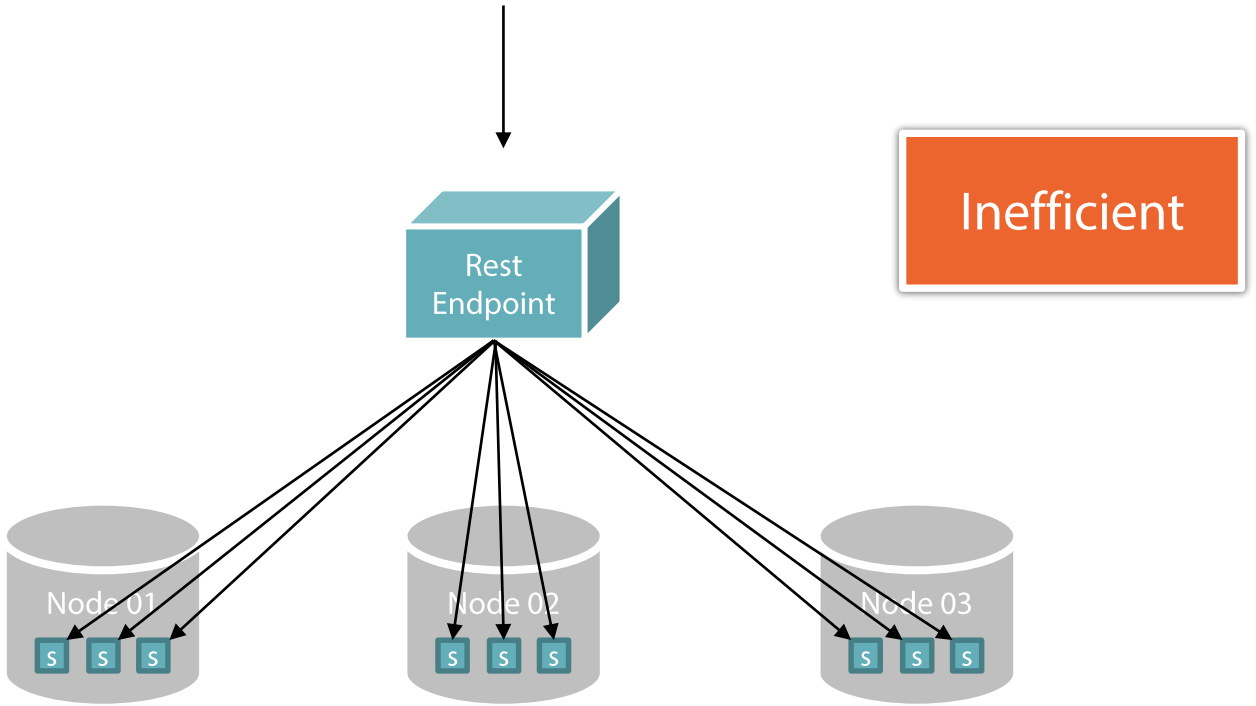

```
{
    "mappings": {
        "post": {
            "_source": {
                "enabled": false
            },
            "properties": {
                "user_id": {
                    "type": "integer",
                    "store": true
                },
                "post_text": {
                    "type": "string"
                },
                "post_date": {
                    "type": "date",
                    "format":"YYYY-MM-DD"
                }
            }
        }
    }
}
```

Mappings _all

```
{
    "mappings": {
        "post": {
            "_all": {
                "enabled": false
            },
            "properties": {
                "user_id": {
                    "type": "integer",
                    "store": true
                },
                "post_text": {
                    "type": "string"
                },
                "post_date": {
                    "type": "date",
                    "format":"YYYY-MM-DD"
                }
            }
        }
    }
}
```

Indexes Routing

```
(GET)    http://localhost:9200/my_blog/post/_search?q=post_text:awesome
```

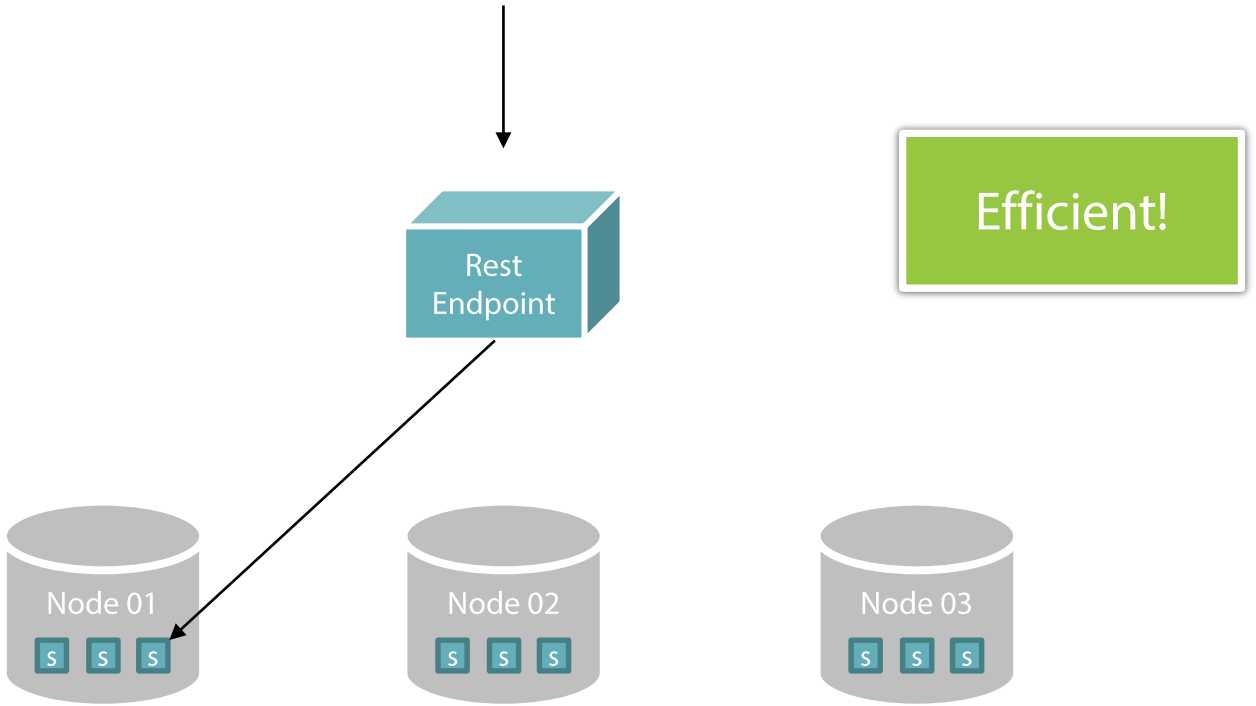


Indexes Routing

```
{
    "mappings": {
        "post": {
            "_routing": {
                "required": true,
                "path": "user_id"
            },
            "properties": {
                "user_id": {
                    "type": "integer"
                },
                "post_text": {
                    "type": "string"
                },
                "post_date": {
                    "type": "date",
                    "format":"YYYY-MM-DD"
                }
            }
        }
    }
}
```

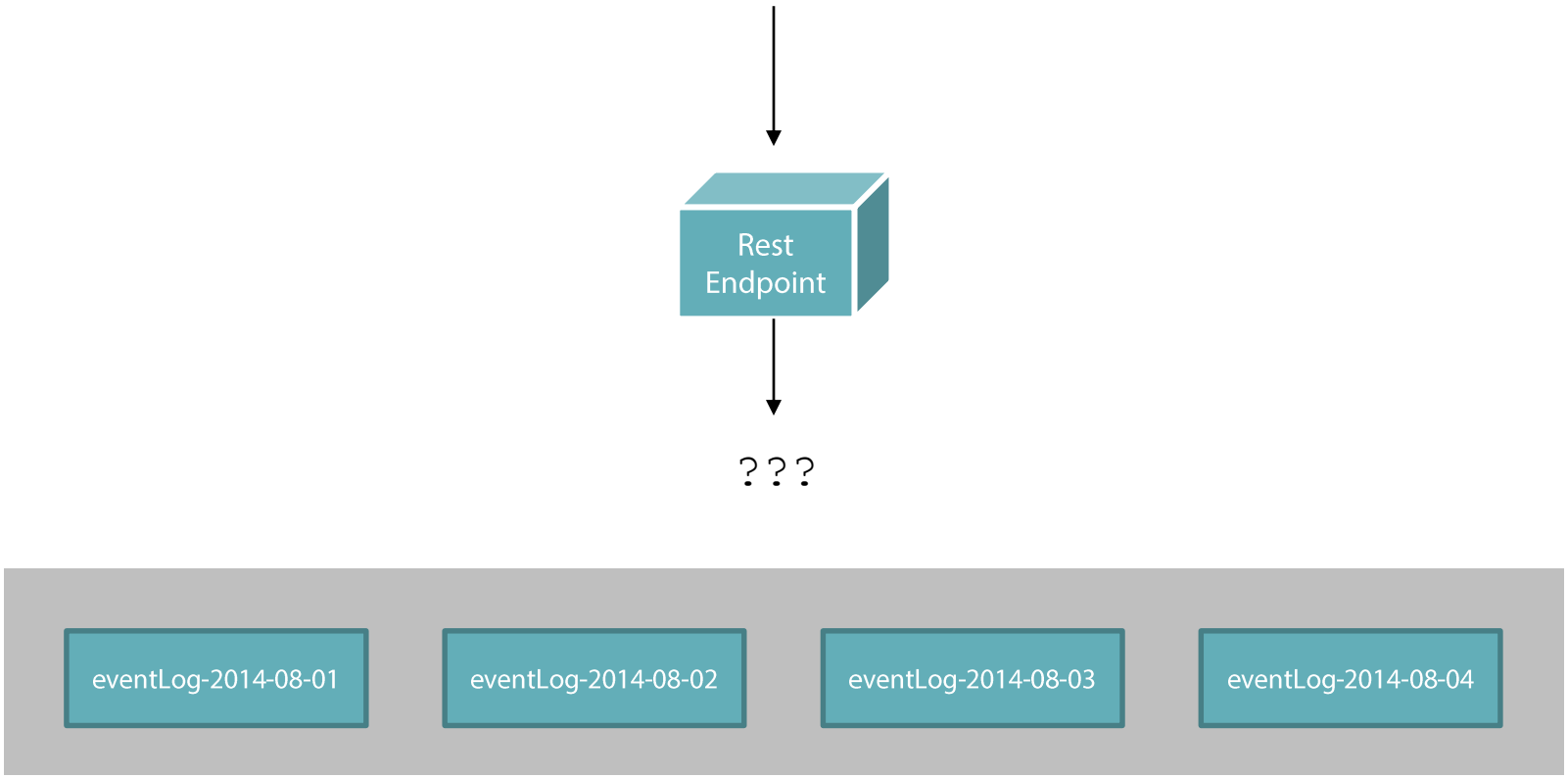
Indexes Routing

```
(GET) http://localhost:9200/my_blog/post/_search?routing=2&post_text:awesome
```



Indexes Aliases

```
(GET) http://localhost:9200/???/Event/_search?q=event:error
```



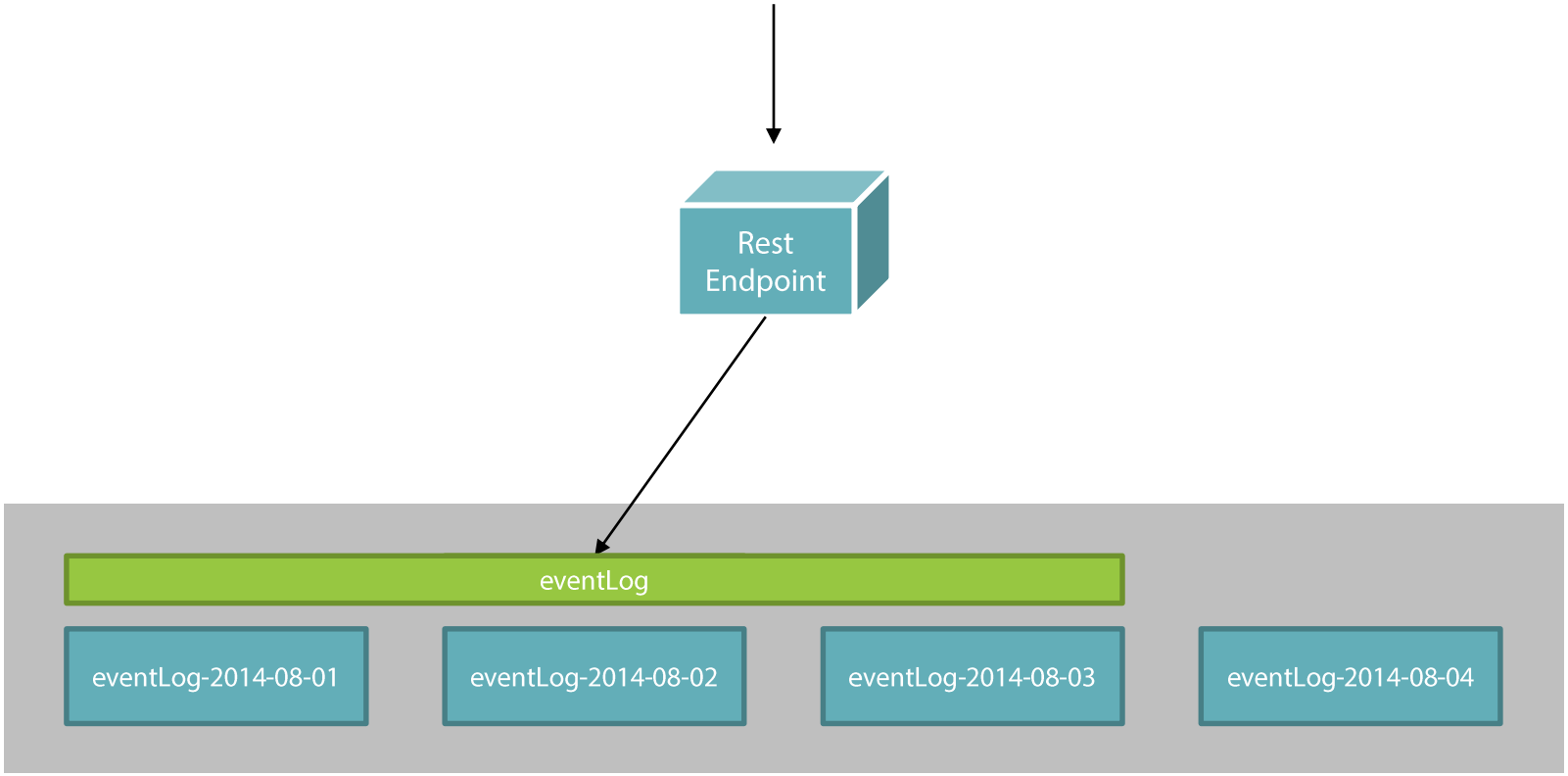
Indexes Aliases

```
(POST) http://localhost:9200/_aliases

{
    "actions" : [
        { "add" : { "index" : "eventLog-2014-08-02", "alias" : "eventLog" } }
    ]
}
```

Indexes Aliases

```
(GET) http://localhost:9200/eventLog/event/_search?q=event:error
```



Elasticsearch Summary

What did we do?

- Learned what Elasticsearch data types there are and how to use them
- Created our first index and then improved it
- Talked about routing, storage, and aliasing

What's next?

- Analysis and choosing analyzers.
- Querying Elasticsearch for data