

すばる望遠鏡ステータスログの データベース化

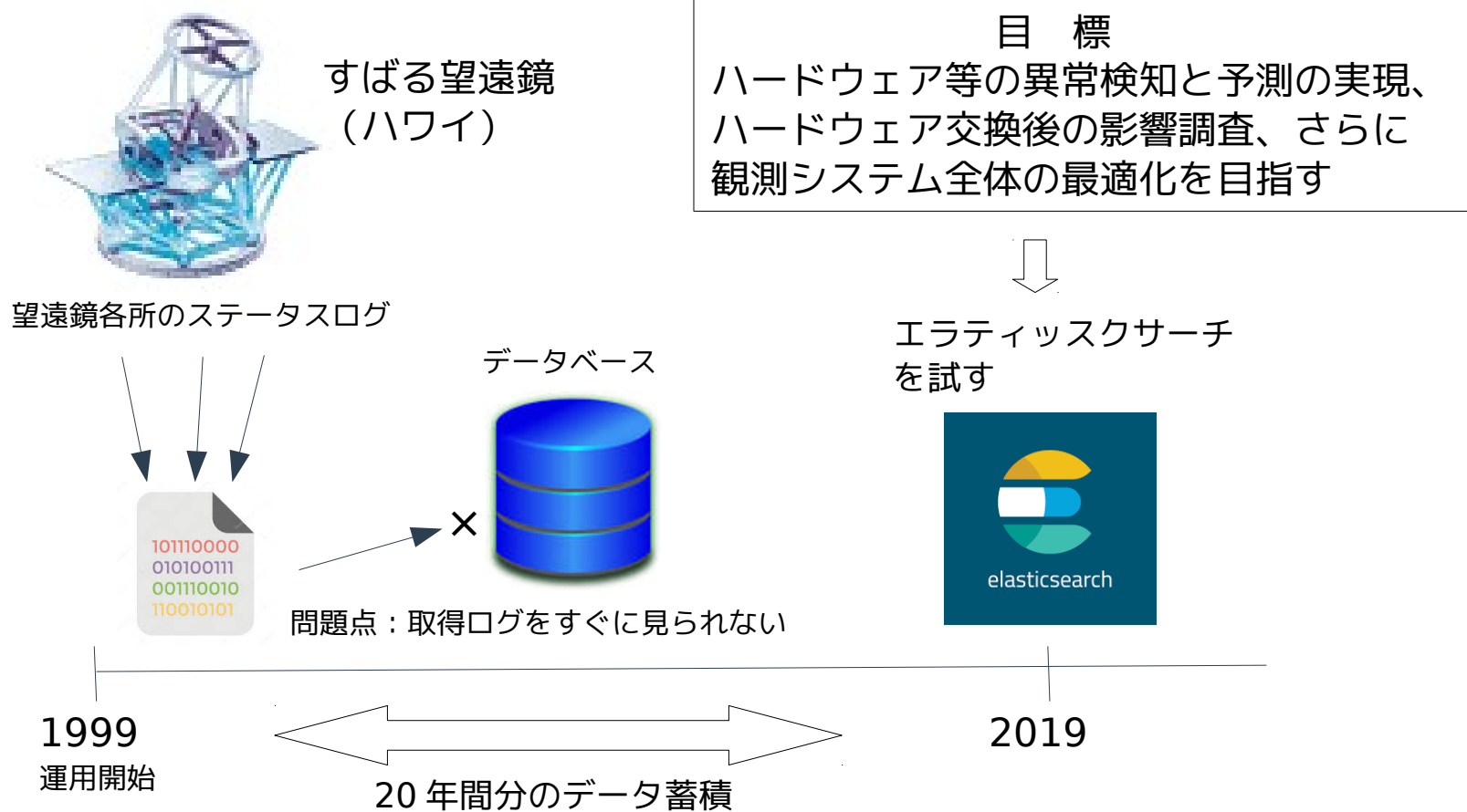
国立天文台

中村京子、小杉城治、佐藤立博、森田英輔、林洋平

発表内容

1. はじめに
2. エラスティックサーチ (**Elasticsearch**) とは
3. ログ投入試験の概要
 3. 1 試験システム
 3. 2 すばる望遠鏡ステータスログ
 3. 3 ログ投入の手順
4. ログ投入試験結果
 4. 1 **Logstash**
 4. 2 **Bulk API**
 4. 3 結果
5. 今後の予定

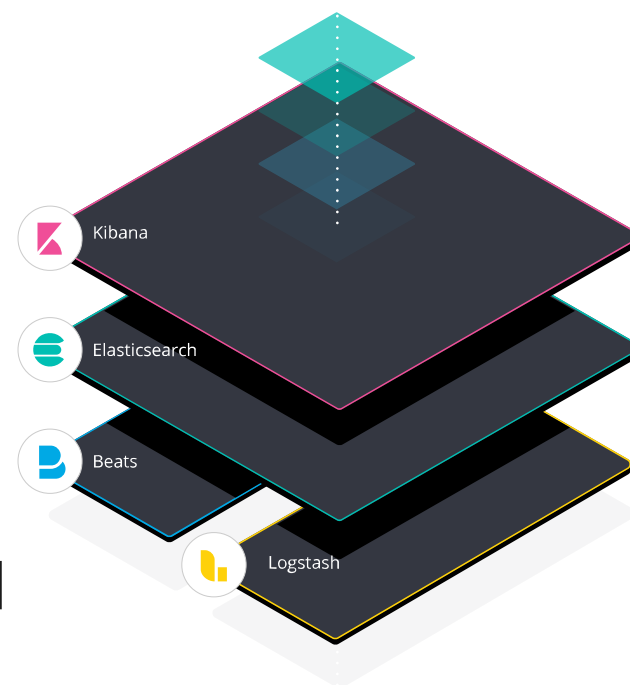
1. はじめに



2. エラスティックサーチ (Elasticsearch) とは

キーワード

- ・ 全文検索ソフトウェア (**2010** 年登場)
- ・ オープンソース
- ・ 分散処理システム (クラスタ構成前提)
- ・ 高速検索、高可用性
- ・ **JSON** に対応したドキュメント指向 **DB**
- ・ **REST API** アクセス
- ・ 多機能プロダクト群「 **Elastic Stack** 」



2. エラスティックサーチ (Elasticsearch) とは

導入企業例

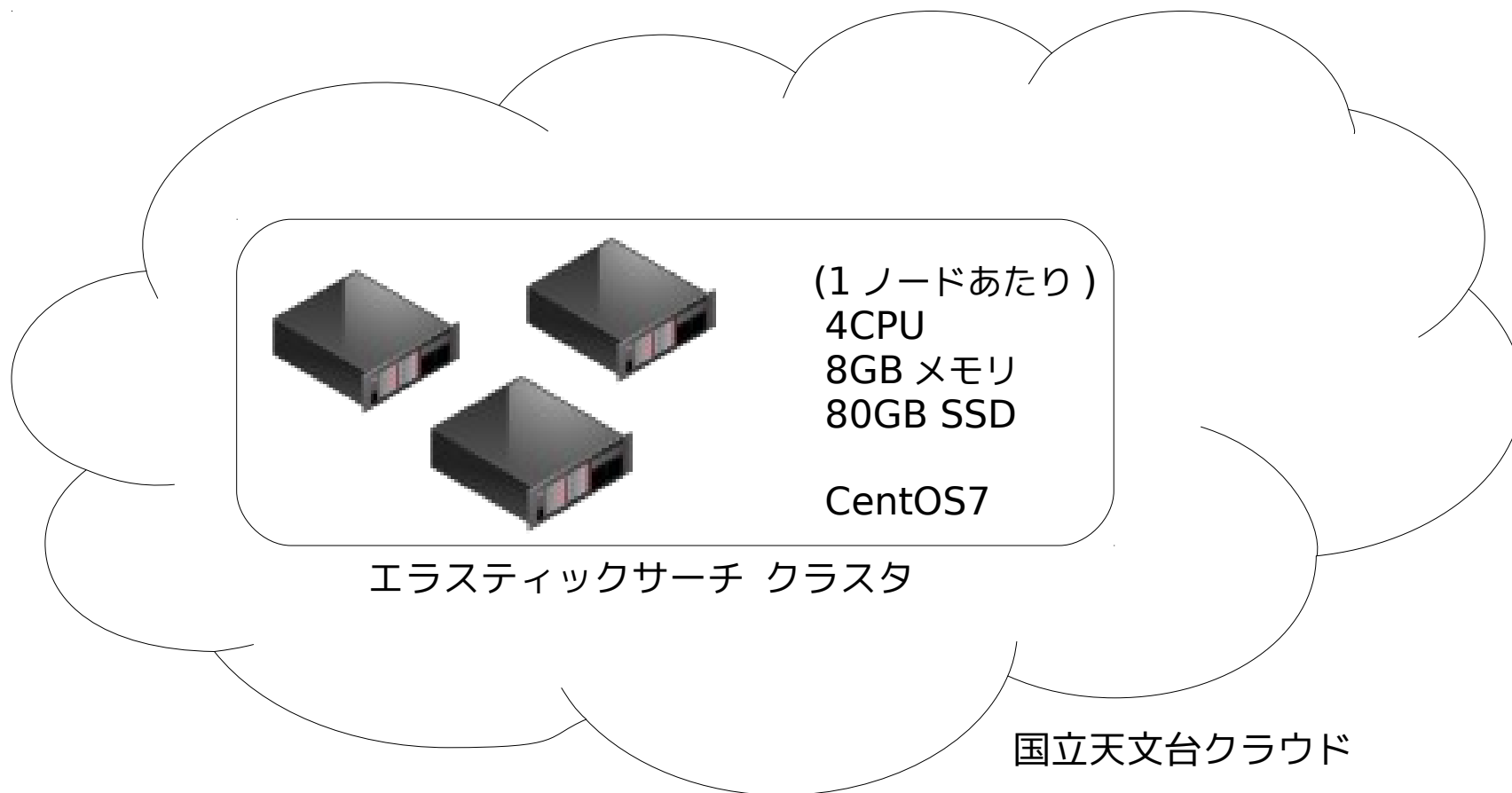
- ・ 「株式会社リコー」
- ・ 「富士通株式会社」
- ・ 「**Docker** 社」
- ・ 「**GitHub** 社」

その他いろいろ <https://www.elastic.co/jp/customers/>

- ・ 「アルマ望遠鏡」 (チリ) **2017年10月**以降の観測ログをオンライン保管、活用

3. 試験環境の概要

3. 1 試験システム



3. 2 すばる望遠鏡ステータスログ

使用したログデータ

期間： **2018年7月23日08:00** から **31日07:59:59(HST)**

ファイル：バイナリをタブ区切りのテキスト形式に変換したもの

ファイルの種類： **76**

ステータスの個数： **9,448** 個

一日分のデータ： **08:00 ~ 16:59:59** と **17:00 ~翌07:59:59** の二
ファイルに分かれる

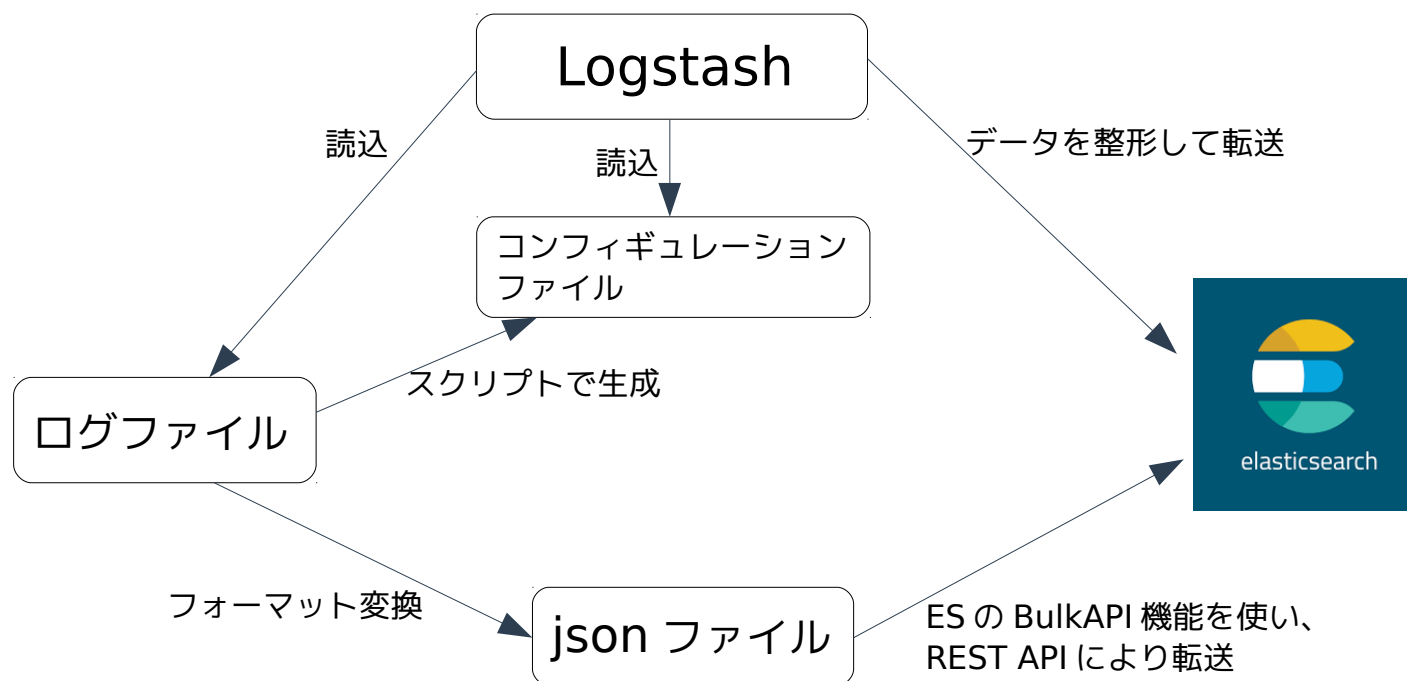
一日あたりログファイルの大きさ（平均）： **1.77GB**

※ 一日あたりログ個数（平均）： **243,056,934** 個

八日間分のファイルの大きさ： **14GB**

3. 3 ログ投入の手順

当初は Logstash 使用
その後 BulkAPI 機能を使用



4 ログ投入試験結果

4.1 Logstash (ファイル例)

ログファイル (抜粋)

#rxdate	rxtime(HST)	sec-from-00:00:00UTC	Dome Rate REF	Dome Rate FDBK-1	Dome Rate FDBK-2	Dome Rate FDBK-3	Dome Rate FDBK-4	Dome Current REF-1	Dome Current REF-2	Dome Current REF-3	Dome Current REF-4	Dome Data SMPLG Time ANS	Dome CMD VEL ANS	Dome Data SMPLG INTVL
2018-07-27	16:59:59.9	10799.9	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000
2018-07-27	17:00:00.9	10800.9	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000

コンフィギュレーションファイル (部分)

```
input {
  file { path => "/<絶対パス>/test.log"
         start_position => "beginning" } }
filter {
  csv { separator => "      "
        skip_header => true
        columns => ["rxdate","rxtime","sec-from-00:00:00UTC","Dome_Rate_REF","Dome_Rate_FDBK-1","Dome_Rate_FDBK-2",...]
        convert => {
          "sec-from-00:00:00UTC" => "float"
          "Dome_Rate_REF" => "float"
          "Dome_Rate_FDBK-1" => "float"
          ... } }
  mutate { add_field => { "time" => "%{rxdate} %{rxtime}" } }
  date { match => [ "time", "yyyy-MM-dd HH:mm:ss.S" ]
        timezone => "HST"
        target => "time" }
  prune { interpolate => true
         whitelist_names => ["time","sec-from-00:00:00UTC","Dome_Rate_REF","Dome_Rate_FDBK-1","Dome_Rate_FDBK-2", ... ] }
}
output {
  elasticsearch { hosts => [ "http://12.34.56.78:9200" ]
                 index => "test" }
}
```

4 ログ投入試験結果

4.1 Logstash (ログ投入所要時間の違い)

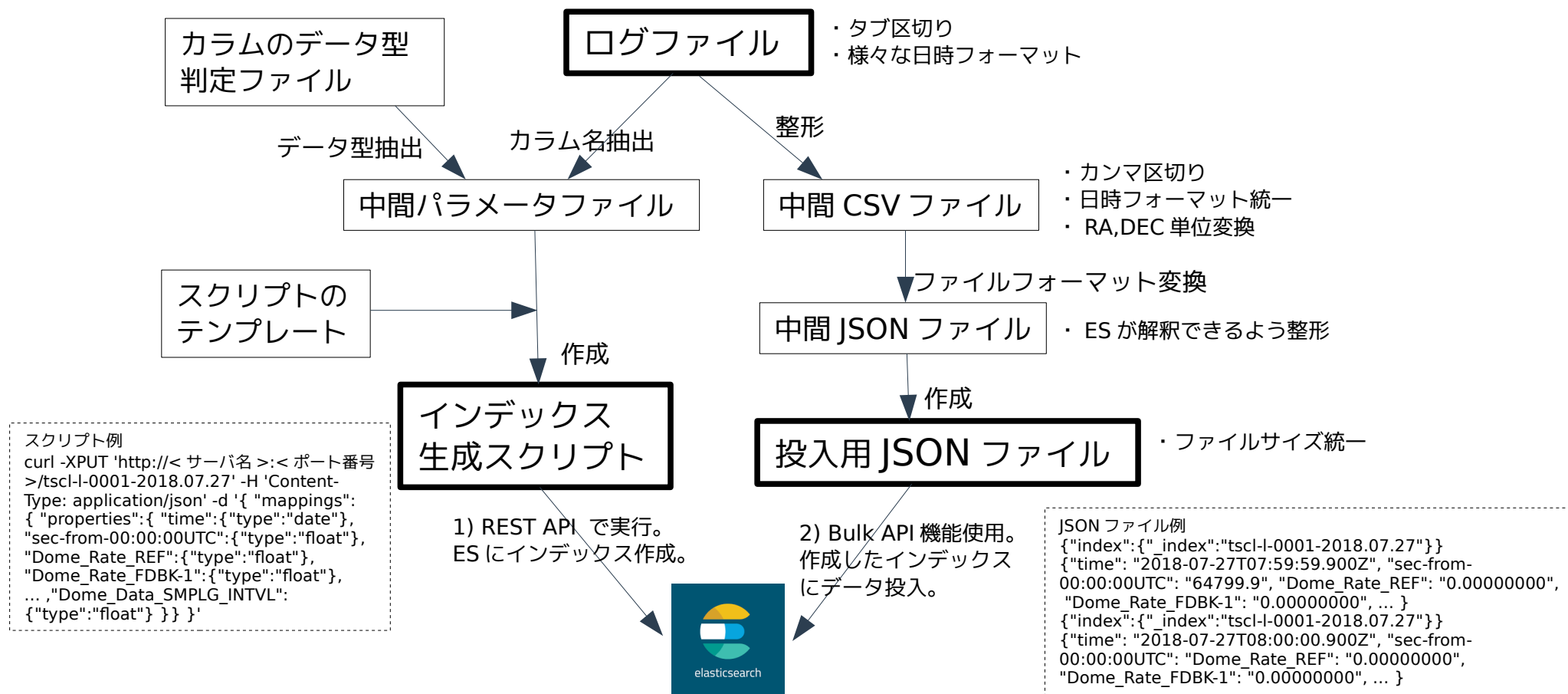
様々な条件で試した結果、わかったこと

- ・ ログファイルのフィールド数が同じ ⇒ 総データ量が少ない方が所要時間は短い
- ・ ログファイルの総データ量が同じ ⇒ フィールド数が少ない方が所要時間は短い
- ・ **Logstash** 実行マシンの処理性能：高い方が所要時間は短い
- ・ **Logstash** の稼働場所：クラスタノード内・外で、所要時間に大きな変化なし
- ・ **Logstash** の稼働台数：二台（同程度の性能）同時の方が一台単独より所要時間は短い

けれども、どう頑張っても **20** 年間分のログ投入に一年以上かかることがわかったため、**Logstash** の使用は諦めた。

4. 2 Bulk API (ログ投入の流れ)

ログ投入の流れ (シェルスクリプトで自動実行)



4. 2 Bulk API (ログ投入所要時間)

試した結果、わかったこと

- ・ 同一規模のデータ量における投入所要時間は、**100MB** で **Logstash** の **1/4** 以下
- ・ 一度に送るデータ量が多いほど所要時間は短い (**100MB** は **50MB** の **7/10** 程度)

注) **Bulk API** がデフォルトで転送できるデータ量の上限は **100MB** である。この値は変更可能だが、投入実行マシンやクラスタの処理性能が低いと、データ投入時にエラーが発生する。

4. 3 結果

ログ投入方法：**Bulk API**、データ投入単位 **90MB**

一日分のログ投入所要時間：平均 **30** 分間

八日間分のログ投入所要時間：約 **4** 時間

20 年間のログ投入の所要時間（予想）：

$0.5(\text{h}) * 365(\text{日}) * 20(\text{年}) = 5$ カ月 **2** 日 **2** 時間

※ 投入前処理にかかる時間（フォーマット変更等）は一日分のログあたり約 **15** 分間。
投入と並列実行可能として、上の計算には含めていない。

5. 今後の予定

現試験システムでの確認は終了。

目下、より高機能の試験用クラスタシステムを **DB** 担当者が構築中。

→ **VMware** 上、**1** ノード **1** クラスタ、ハイパースレッド無効化により、一日分のログ投入時間が **10** 分間程度になった。

(注) **VMware** はデフォルトがハイパースレッド「有効化」。その際は **30** 分間強。

今後の予定

- ・ 動作やデータの確認のため、半年間程度のログ投入の実験
- ・ **Elastic Stack** の **Kibana** 等による、投入データの分析
- ・ 観測コマンドログも含め、望遠鏡と観測装置を合わせた動作分析

目標：観測システム全体の最適化を目指す！