

Slackを無償で使い倒すツールを作った話

株式会社グローバルネットコア
高津 伸哉

Slack活用していますか？

- 弊社では2014年6月から利用開始。
- 日々の業務で活用され、メインワークスペースのChannel数は20オーバー。
- Slack APPを利用し、社内・社外のシステムと連携
⇒ netatomo、IFTTTと連携し、部屋内の二酸化炭素濃度を通知とか。。



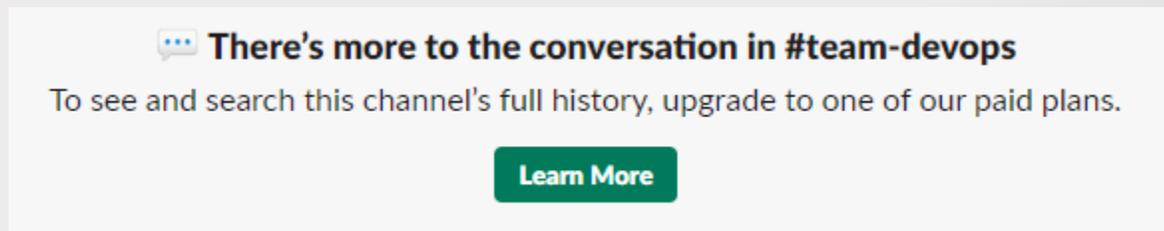
IFTTT APP 12:29 PM

Gnc2F - 昴ビル2F室内 station March 26 2019 at 11:13AM

CO2濃度が2505.00 ppmを超えました。体調に影響を与えるレベルです。早急に換気を行ってください。(March 26, 2019 at 11:13AM)

Slack否定派の意見

- IP Messenger じゃダメなの？
 - Slackの方がオープンで認識共有・意見交換できて良いですよ。
 - 内緒話はIP Messenger（もしくはダイレクトメッセージ）でどうぞ。
- 投稿に抵抗があるんだけど。。
 - 慣れてください。固い空気では無いのでお気軽に。
- 昔の投稿が見れないと、業務で使うには不安
 - **有料版**なら見れますよ。



無償利用の限界

<https://globalnetcore.slack.com/plans>

コレが問題

フリー
¥0

主な機能:

- ✓ 最大10,000件のメッセージから検索
- ✓ 最大10個のアプリとインテグレーション
- ✓ 1対1のビデオ通話

現在ご利用のプラン

スタンダード
¥850

(アクティブユーザー1名あたり、月額)
¥960/ユーザー/月 (月払いの場合)

主な機能:

- ✓ 検索範囲に制限無し
- ✓ 無制限のアプリ
- ✓ 画面共有機能付きのグループ通話
- ✓ ゲストアカウントと共有チャンネル

今すぐアップグレード

もっと詳しく

プラス
¥1,600

(アクティブユーザー1名あたり、月額)
¥1,800/ユーザー/月 (月払いの場合)

主な機能:

- ✓ 99.99%のアップタイムを保証するサービス品質保証契約
- ✓ ユーザーのプロビジョニングとデプロビジョニング
- ✓ SAMLベースのシングルサインオン (SSO)
- ✓ すべてのメッセージの Corporate Export

今すぐアップグレード

もっと詳しく

抜け道を探す

- ~~素直に有料版にする~~
- お金をかけずに何とかならないか。
 - Slack管理画面から、投稿データのExportが出来るらしい。
 - Slack APIからも同じことが可能そう。

お試し

channels.historyを実行

```
[root@slack-archive general]# ls -al
合計 2688
drwxr-xr-x  2 slack slack 20480  4月 10 06:30 .
drwxr-xr-x 42 slack slack  4096  4月  5 06:30 ..
-rw-r--r--  1 slack slack  3754  2月 28 20:54 2014-06-01.json
-rw-r--r--  1 slack slack 10312  2月 28 20:54 2014-06-02.json
-rw-r--r--  1 slack slack 13607  2月 28 20:54 2014-06-03.json
-rw-r--r--  1 slack slack 12835  2月 28 20:54 2014-06-04.json
-rw-r--r--  1 slack slack  4865  2月 28 20:54 2014-06-05.json
-rw-r--r--  1 slack slack   723  2月 28 20:54 2014-06-06.json
-rw-r--r--  1 slack slack  1361  2月 28 20:54 2014-06-09.json
-rw-r--r--  1 slack slack  6132  2月 28 20:54 2015-01-28.json
-rw-r--r--  1 slack slack  4004  2月 28 20:54 2015-01-29.json
-rw-r--r--  1 slack slack  2285  2月 28 20:54 2015-02-01.json
-rw-r--r--  1 slack slack   388  2月 28 20:54 2015-02-09.json
-rw-r--r--  1 slack slack  3090  2月 28 20:54 2015-02-10.json
-rw-r--r--  1 slack slack   164  2月 28 20:54 2015-02-11.json
-rw-r--r--  1 slack slack   508  2月 28 20:54 2015-02-12.json
-rw-r--r--  1 slack slack  3251  2月 28 20:54 2015-02-17.json
-rw-r--r--  1 slack slack  1739  2月 28 20:54 2015-02-19.json
```

Slack上で見れなくなった
データも取得出来る！！

※考えてみれば、フリープランから有料版に切り替えた際に過去投稿を参照可能にするならデータは保持してるよね。。

調査

過去ログのViewerがあれば解決しそう！

車輪の再発明は避けたい。（**楽**したい）

絶対に誰かViewerを作ってるはず！

と調べるが・・・

調査例1 馴染みのない言語

README.md

Welcome to the Slack Archive Viewer

Why? This was an experiment to quickly show archived Slack data. It was also a pretty compelling reason to mess around with Node, Bootstrap 4, and Express

food-and-kitchen

- boards-n-dice
- back-beatin-and-edu
- cars
- cycling
- dadjokes
- darksouls
- dev-subst
- div
- documentation
- food-and-kitchen
- fun-in-a-year-for-5
- golang
- general
- gullies
- gsg
- halos
- homeowner-stuff
- idoit-tux
- investing
- libations
- mental-health
- minecraft
- music
- nantesche-lounge
- noble
- philosophy
- politics
- random
- running-plus-health
- sgjk
- sports-ball
- travels
- vent
- vp-peripherals

-  **kint** Mon, 28 Dec 2015 14:58:57 GMT
@kint has joined the channel
-  **kint** Mon, 28 Dec 2015 14:58:57 GMT
@kint set the channel purpose: Share recipes and kitchen/cooking hacks. Also, proud kitchen achievements.
-  **keiner** Mon, 28 Dec 2015 15:00:48 GMT
@keiner has joined the channel
-  **julemp** Mon, 28 Dec 2015 15:00:48 GMT
@julemp has joined the channel
-  **lona007** Mon, 28 Dec 2015 15:00:48 GMT
@lona007 has joined the channel
-  **runnerlyfe** Mon, 28 Dec 2015 15:00:48 GMT
@runnerlyfe has joined the channel
-  **thecreepone** Mon, 28 Dec 2015 15:00:48 GMT
@thecreepone has joined the channel
-  **thehoagie** Mon, 28 Dec 2015 15:00:48 GMT
@thehoagie has joined the channel
-  **robocap** Mon, 28 Dec 2015 15:00:48 GMT
@robocap has joined the channel
-  **smacombie** Mon, 28 Dec 2015 15:00:48 GMT
@smacombie has joined the channel
-  **kint** Mon, 28 Dec 2015 15:05:03 GMT
I'll start this board with stating how proud I am of how my turkey turned out. It was even not dry the next day for sandwiches!
-  **runnerlyfe** Mon, 28 Dec 2015 15:06:30 GMT
Great job @kint! I know for sure that @smacombie's meat about how proud they are not being so proud!

Setup

You will need to export Slack data. Download the zipped JSON file from: <https://<your-team-name>.slack.com/services/export>

- 見た目シンプル過ぎない？ 検索機能なし。
- Node.jsのカスタマイズ難しそう。。
(個人的に)

⇒ やっぱり使い慣れた言語
が良いなあ。
メンテ出来る人いないし。。

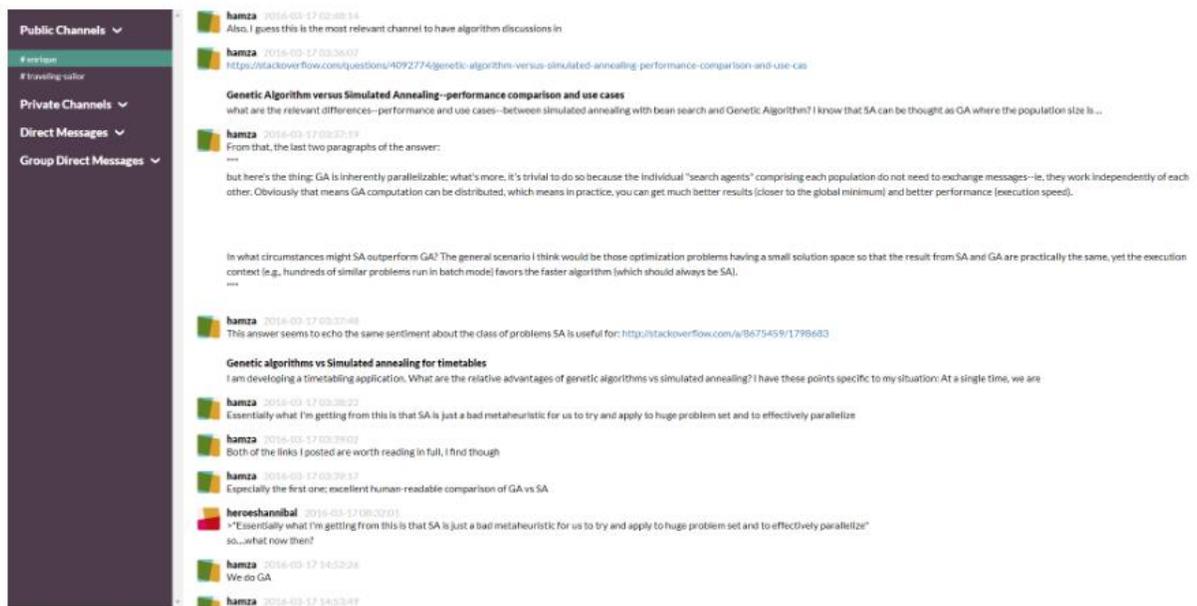
調査例2 表示量考えてない

📖 README.md

Slack Export Viewer

build passing pypi package 0.8.2

A Slack Export archive viewer that allows you to easily view and share your Slack team's export (instead of having to dive into hundreds of JSON files).



The screenshot shows a web interface for viewing Slack export data. On the left, there is a sidebar with navigation options: Public Channels, Private Channels, Direct Messages, and Group Direct Messages. The main content area displays a list of messages from a channel. The messages are displayed in a clean, readable format with user avatars, names, and timestamps. The text of the messages is clearly visible, including a discussion about Genetic Algorithms versus Simulated Annealing. The interface is well-organized and easy to navigate.

- 見た目かなり良い感じ！
- Pythonかあ。。まあ、なんとかなるかな。
インストールしてみた。



- コレ、HTML一枚でChannelの投稿全部表示してる！？重っ！！
- 検索の機能もないのね。。

調査例3 まともに動かない

```
159 // <generate html from channel logs>
160 $htmlDir=$baseDir.'/.html';
161 if(!is_dir($htmlDir)) mkdir($htmlDir);
162 $channels=scandir($jsonDir);
163 $channelNames=array();
164 $mostRecentChannelTimestamps=array();
165 foreach($channels as $channel) {
166     if($channel=='.' || $channel=='..') continue;
167     if(is_dir($channel)) continue;
168
169     $mostRecentChannelTimestamp=0;
170     if($message['ts']>$mostRecentChannelTimestamp) {
171         $mostRecentChannelTimestamp=$message['ts'];
172     }
173     $array=explode('.json',$channel);
174     $channelName=$array[0];
175
176     $channelHtmlFile=$htmlDir.'/$channelName.html';
177     if(file_exists($channelHtmlFile)) {
178         echo "HTML already exists ".$channelJsonFile."\n";
179         continue;
180     }
181
182     array_push($channelNames,$channelName);
183     echo '===='. "\n";
184     echo "Generating HTML for #".$channelName. "\n";
185     echo '===='. "\n";
186     $messages=json_decode(file_get_contents($jsonDir.'/$channel'),true);
187     if(empty($messages)) continue;
188     $htmlMessages='<html><body><style>'.$stylesheet.'</style><div class="messages">';
189     foreach($messages as $message) {
190         if(empty($message)) continue;
191         if(empty($message['text'])) continue;
192         echo '.';
193
194         // change <@U38A3DE9> into levelsio
195         if(stripos($message['text'],'<@')!=false) {
196             $usersInMessage=explode('<@',$message['text']);
197             foreach($usersInMessage as $userInMessage) {
```

- PHPは有難い。
- GitHubではソースコード1ファイルのみ。大丈夫かな？ 試してみた。



- なんか凄い数のエラーが吐かれるんですけど！！
- （修正して）一応、HTML出力されるけど、それだけ。
- なんかコードが素人臭い。。
正規表現使えば、もっと簡単に出来そう。

調査例 (番外) 使われないツール

Kibanaを用いたSlackの統計情報の可視化と全文検索

<https://qiita.com/shinkbr/items/1d4cbec07996a5d65bee>

概要

Kibanaを用いてSlackの過去ログから統計情報の可視化と全文検索を行う方法を紹介します。

Slackの過去ログをELKのような全文検索エンジンに入れることで、以下のメリットが得られます。

1. Slackの提供していない統計情報の可視化
2. 無料版Slackのメッセージ数制限を超えて全ての過去ログから全文検索

本記事では、簡単のためDockerでELKスタックを立ち上げ、自作したスクリプトでSlackの過去ログを流し込み、Kibanaから可視化と全文検索を行う方法を解説します。



- やりたい事はほぼ一致、おまけに統計情報の可視化も可能！
- ElasticsearchとKibanaを軽く勉強して、Slackデータを取り込み、社内で公開。



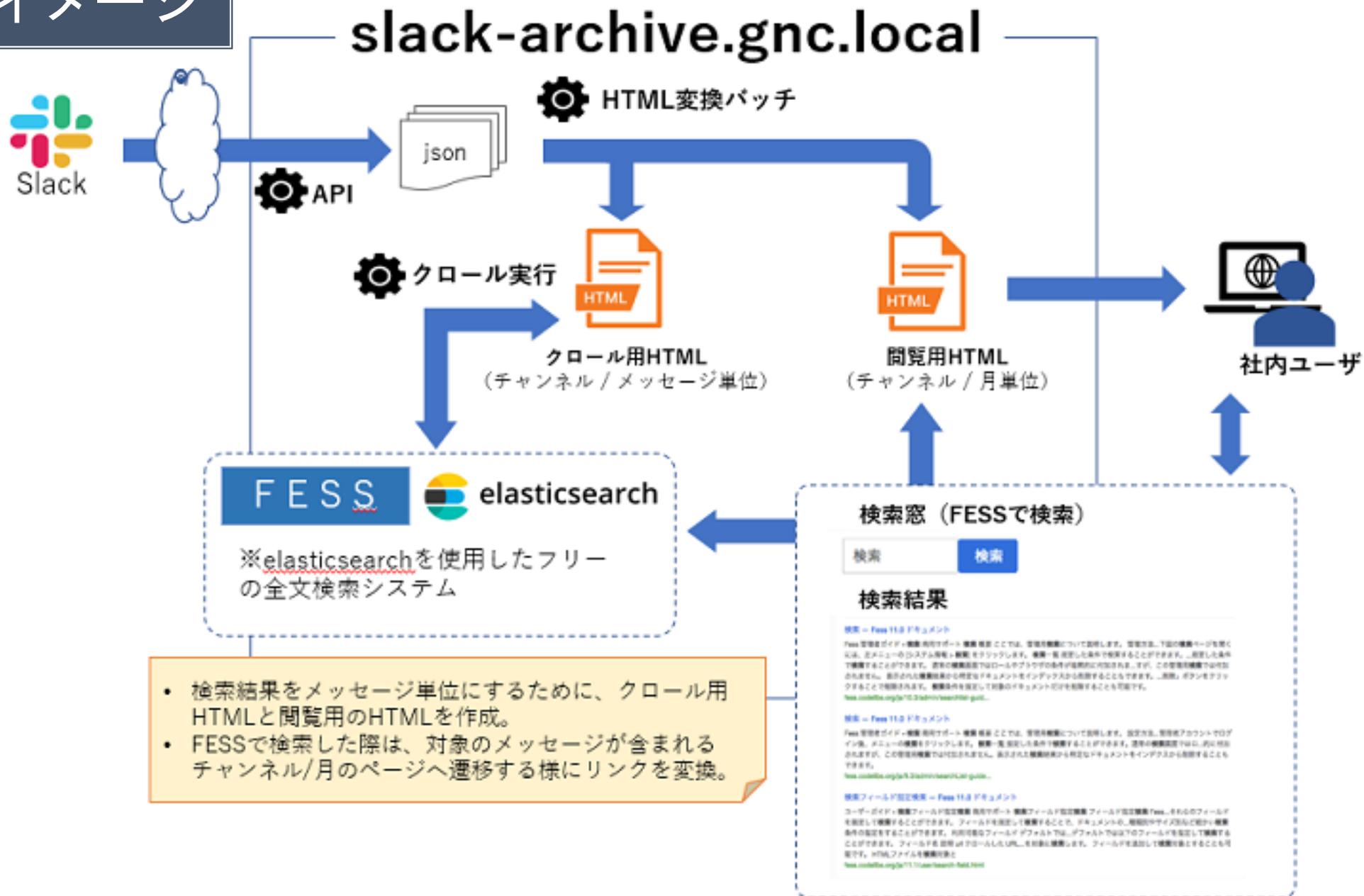
- ユーザの反応皆無！
- Kibanaで検索って敷居高いよね。。

経験値にはなりました。

自前で作ろう！

- 日々の投稿データを蓄積
 - 1日1回で前日投稿分を取得（リアルタイム性は問わない）
- チャンネルの投稿データを年月毎に表示
 - JsonからHTMLへの変換は、調べたコードを参考に
- 検索機能必須
 - FESS（フリーの全文検索システム）を利用（工数削減）
（以前、導入しようと調査した経験アリ）

概要イメージ



- 検索結果をメッセージ単位にするために、クロール用HTMLと閲覧用のHTMLを作成。
- FESSで検索した際は、対象のメッセージが含まれるチャンネル/月のページへ遷移する様にリンクを変換。

Slackデータの取得

公式でデータ取得APIが用意されているので、それを使えばOK！

今回使用したAPIは下記の通り

- ・ユーザ情報取得

(公式マニュアル <https://api.slack.com/methods/users.list>)

- ・チャンネル情報取得

(公式マニュアル <https://api.slack.com/methods/channels.list>)

- ・チャンネル投稿データ取得

(公式マニュアル <https://api.slack.com/methods/channels.history>)

JSON ⇒ HTML変換バッチ

投稿データJSON（一部抜粋）

```
{
  "client_msg_id": "c909a035-9ace-4707-b760-5089a6b521d7",
  "type": "message",
  "text":
  "Trello¥u306bSprint09¥u306e¥u30dc¥u30fc¥u30c9¥u3092¥u4f5c¥u6210¥u3057¥u307e
  ¥u3057¥u305f¥u3002",
  "user": "U03KRSXJ1",
  "ts": "1554879148.050800"
},
{
  "client_msg_id": "61a3b9af-1cc3-44f1-ad28-3cdec3470911",
  "type": "message",
  "text":
  "¥u6b21¥u56de¥u306f4¥/15(¥u6708)15¥u6642¥u304b¥u3089¥u3067¥u8a2d¥u5b9a¥u30
  57¥u307e¥u3057¥u305f¥u3002¥n¥u3053¥u3061¥u3089¥u306e¥u5185¥u5bb9¥u3092¥u
  4e88¥u5b9a¥u3057¥u3066¥u3044¥u307e¥u3059¥u3002¥n``¥n¥u30fb¥u524d¥u56de¥u
  4f5c¥u6210¥u3057¥u305f¥u696d¥u52d9¥u30d5¥u30ed¥u30fc¥u306b¥u5f93¥u3063¥u3
  066¥u3001¥u4eee¥u60f3¥u7684¥u306b¥u4e00¥u5ea6¥u56de¥u3057¥u3066¥u307f¥u30
  8b¥n
  ¥u5c0e¥u5165¥u30fb¥u969c¥u5bb3¥u5bfe¥u5fdc¥u30fb¥u89e3¥u7d04¥n¥n¥u30fb¥u3
  05d¥u306e¥u4e0a¥u3067¥u6539¥u5584¥u70b9¥u3092¥u6d17¥u3044¥u51fa¥u3057¥n``
  `¥n¥u305d¥u308c¥u307e¥u3067¥u306b¥u3001MFW¥u306b¥u3064¥u3044¥u3066¥u540
  4¥u81ea¥u3067¥u696d¥u52d9¥u30d5¥u30ed¥u30fc¥u6539¥u5584¥u70b9¥u306e¥u4e8
  b¥u524d¥u6d17¥u3044¥u51fa¥u3057¥u306e¥u691c¥u8a0e¥u3092¥u304a¥u9858¥u304
  4¥u3057¥u307e¥u3059¥u3002",
  "user": "U03KRSXJ1",
  "ts": "1554874301.050200"
},
```

QiitaやGitHubにあるソースを参考に、ゴリゴリ変換プログラム（PHP）を書く。

※参考にしたソースが腐ってたので、余計に苦労した。

なるべく本物に近い雰囲気にしたかったので、Web版SlackのHTML、CSSを解析。

検索機能

FESSのインストールと設定

- 当初はDockerで立てたが、性能が出ず。。
- FESS内部でElasticsearchを持っているが、こちらもイマイチだったため、外部のElasticsearchを参照するように設定。
- クローリング設定
 - ⇒ メッセージ単位に検索出来るように、クローリング用、参照用とHTMLを別に用意した。

そして出来上がったものがコチラ

メッセージ閲覧画面

team-devops 2019 02

検索

詳細検索

2019 02

At that time channels

alert-task

general

important

question

share-information

team-devops

team-idccablingteam

team-idcservice

team-kikaku

team-nyo

team-support

technical-information

web-dc-share-info

websystem

websystem-alert

websystem-important

- Top -

Please select the year and month

2019

01 02 03 04

2018

06 07 08 09 10 11 12

- Top -

資料にはないけど、ストリーミングを見ると、いつも出てくる運用チームと設計チームの役割分担の話が出てきま

yasuyuki.kaneko 2019-02-04 10:31
あとは、運用チームの人のモチベーションアップ、技術力アップを、どう実現できるか・

yasuyuki.kaneko 2019-02-04 10:32
名言

- 「運用エンジニアが一番技術をわかっていないといけない」
- 「単純作業しかしないオペレーター」は運用エンジニアではない
 - 良い（安定して楽な）運用を作るのが運用エンジニア
 - 良い運用を作るには技術が必要

takatsu 2019-02-04 19:14
Fessの環境にHTMLをアップしました。リンクも張ったので単体でもそれなりに見れるはず。。
<http://slack-archive.gnc.local/slack-viewer/view/>

hirokuni.sato 2019-02-05 16:29
デイリーで話していたtokenの件、legacy以外で取得できることを確認しました。

月毎、Channel毎に表示 (のリンクも追加)

この状態でFESSのコントロールがわかります。
<http://slack-archive.gnc.local/slack-viewer/view/>

takatsu 2019-02-06 13:06
FESSのマニュアルにFESS利用しているサイトとして新潟県がありました。
<https://fess.codelibs.org/ja/users.html#id1>
<http://search.pref.niigata.lg.jp/search/index/>

takatsu 2019-02-06 18:31
Slackのインラインコード記述に対応しました。大分見た目は良い感じ。
2019-02-06 18:31 takatsu posted.

image.png

FESS検索窓

検索窓

286 件 (0.02秒)

投稿日

検索結果のページ内遷移と、
ハイライトはjQueryで実現

#team-devops 2019-04-04 09:47 hideo.aota

高津 or 宏邦 どちらか行ってみる？ <https://www.slack-archive.gnc.local/slack-viewer/view/team-...>

#team-devops 2019-04-02 14:24 hideo.aota

メンバーは、後藤・清・高津・宏邦・青田で。期間なども含め [slack-archive.gnc.local/slack-viewer/view/team-...](https://www.slack-archive.gnc.local/slack-viewer/view/team-...)

#team-devops 2019-03-26 09:38 yasuyuki.kaneko

時修正する人→宏邦・指摘事項や要修正事項のログを残す人→ [slack-archive.gnc.local/slack-viewer/view/team-...](https://www.slack-archive.gnc.local/slack-viewer/view/team-...)

#team-support 2019-03-25 09:24 kiyoshi.kobayashi

予定 打合:サポートチーム業務MT 参加者 (5名) 小林 清 高津 [slack-archive.gnc.local/slack-viewer/view/team-...](https://www.slack-archive.gnc.local/slack-viewer/view/team-...)

#team-devops 2019-03-14 11:48 takatsu

—高津 稼働後にpostfixをyumアップデートすることは無いのでは？
—青田、宏邦 ↑羽賀君に確認する—高津... [slack-archive.gnc.local/slack-viewer/view/team-...](https://www.slack-archive.gnc.local/slack-viewer/view/team-...)

#team-devops 2019-03-13 12:12 kiyoshi.kobayashi

—高津 <https://globalnetcore.slack...> 無いので必要ないのでは？
2019-03-01 17:29... [slack-archive.gnc.local/slack-viewer/view/team-...](https://www.slack-archive.gnc.local/slack-viewer/view/team-...)

Slack Archive System

2019 04

At that time channels

- #_emergency
- #_release
- #sales-nw-share-info
- #sales-web-share-info
- #alert-task
- #general
- #important
- #question
- #share-information
- #team-devops**
- #team-idccablingteam
- #team-idcservice
- #team-kikaku
- #team-unyo
- #team-support
- #technical-information
- #web-dc-share-info
- #websystem
- #websystem-alert

team-devops 2019



hideo.aota 2019-04-04 09:47
@takatsu @hirokuni.sato
Kompiraのセミナーの案内が来ていました。
高津 or 宏邦 どちらか行ってみる？

<https://www.kompira.jp/kompirameeting2019tokyo/>

Kompiraミーティング 2019 Tokyo

効率化は当たり前。自動化はいよいよ次のステージへ



takatsu 2019-04-04 09:55
良さそうですね。自分は前にkompiraのハンズオンに参加したことがあるので、(宏邦さんじゃなくとも)他の方もいいですね。
あまり手を挙げる人いなそうですが。。



takashi.goto 2019-04-04 11:51
@takatsu 営業から、営業slackもアーカイブがあると嬉しいという意見があったんだけど、対応するのは大変？



takatsu 2019-04-04 12:13
2、3日まとまった時間があれば対応できると思います。来週中であればなんとか。。ちょうどいまスプリントの隣



takatsu 2019-04-04 17:32
Slack、globalnetcoreで4つチャンネル追加ということで、slack-archiveで参照できるように設定しておきました。
明日から反映されるはずです。

比較

Slack (本物)

The screenshot shows the Slack interface for the #team-devops channel. The left sidebar contains a list of channels, with #team-devops selected. The main area shows a conversation starting on Thursday, April 4th. The messages are as follows:

- hideo.aota** (9:47 AM): @takatsu @hirokuni.sato
Kompiraのセミナーの案内が来ていました。
高津 or 宏邦 どちらか行ってみる？
<https://www.kompira.jp/kompirameeting2019tokyo/>
kompira.jp
Kompiraミーティング 2019 Tokyo
効率化は当たり前。自動化はいよいよ次のステージへ
- takatsu** (9:55 AM):
良さそうですね。自分は前にkompiraのハンズオンに参加したことがあるので、（宏邦さんじゃなくとも）他の方で興味のある人に行ってもらっても良いかなと思います。
あまり手を挙げる人いなそうですが。。
- takashi.goto** (11:51 AM): @takatsu 営業から、営業slackもアーカイブがあるとうれしいという意見があったんだけど、対応するのは大変？
- takatsu** (12:13 PM): 2、3日まとまった時間があれば対応できると思います。来週中であればなんとか。。ちょうどいまスプリントの隙間期間で時間取れますし。(edited)
- takatsu** (5:32 PM): Slack、globalnetcoreで4つチャンネル追加ということで、slack-archiveで参照できるように設定しておきました。明日から反映されるハズです。
👍 1
- takashi.goto** (5:33 PM): ありがとうございます！！
- hideo.aota** (6:14 PM): 自分も何度か行ってるけど、なかなか良い話し聞けるんだよね
誰も手を挙げなければ自分が行きますよ^^

The bottom of the interface shows a message input field with the text "Message #team-devops" and a send button.

Slack-Archive (偽物)

The screenshot shows the Slack-Archive website. The top left has the title "Slack Archive System" and a date filter for "2019 04". Below this is a list of channels, with #team-devops selected. The main content area shows a conversation from the #team-devops channel on 2019-04-04. The messages are as follows:

- hideo.aota** (2019-04-04 09:47): @takatsu @hirokuni.sato
Kompiraのセミナーの案内が来ていました。
高津 or 宏邦 どちらか行ってみる？
<https://www.kompira.jp/kompirameeting2019tokyo/>
Kompiraミーティング 2019 Tokyo
効率化は当たり前。自動化はいよいよ次のステージへ
- takatsu** (2019-04-04 09:55):
良さそうですね。自分は前にkompiraのハンズオンに参加したことがあるので、（宏邦さんじゃなくとも）他の方です。
あまり手を挙げる人いなそうですが。。
- takashi.goto** (2019-04-04 11:51): @takatsu 営業から、営業slackもアーカイブがあるとうれしいという意見があったんだけど、対応するのは大変？
- takatsu** (2019-04-04 12:13): 2、3日まとまった時間があれば対応できると思います。来週中であればなんとか。。ちょうどいまスプリントの隙間期間で時間取れますし。(edited)
- takatsu** (2019-04-04 17:32): Slack、globalnetcoreで4つチャンネル追加ということで、slack-archiveで参照できるように設定しておきました。明日から反映されるハズです。
- takashi.goto** (2019-04-04 17:33): ありがとうございます！！
- hideo.aota** (2019-04-04 18:14): 自分も何度か行ってるけど、なかなか良い話し聞けるんだよね
誰も手を挙げなければ自分が行きますよ^^

The bottom of the page has a date filter for "2018" and a "Please select the year and month" prompt.

補足

プライベートチャンネルや、ダイレクトメッセージは対応していません。

-- Slackヘルプページからの転記 --

プライベートチャンネルやダイレクトメッセージ、グループDMからコンテンツをエクスポートする際には、ワークスペースのオーナーは Slack に連絡し、エクスポートを申請する必要があります。申請の際には、ワークスペースのオーナーは毎回必ず以下のいずれかを Slack へ提出する必要があります。これらの提出がない場合、データのエクスポート申請は却下となりますのでご注意ください。

- (a) 政府機関もしくは規制当局による有効かつ拘束力のある要求 (命令) 書
- (b) メンバーの同意書
- (c) データのエクスポートが適切な法律および規定に基づく請求・権利であることの証明。

苦勞した点

•jsonの解読

⇒ 引用や、画像プレビュー、ファイルDLなどそれなりにパターンがある。
対応するためのコーディングがツライ。

•FESSのクローラ設定

⇒ 当初は月単位で作成した閲覧ページに対してクローラしていた。
検索窓で表示される内容が、月単位になるため使い物にならなかった。

•ページ構成

⇒ 投稿閲覧ページ以外、あまり考えて無かった。。
使い易い構成にするため、TOPページや、サイドメニューは関係者の意見を取り入れました。

評価

- 利用者からはそれなりに高評価！

⇒ 「安心してSlackを使える！」、「検索助かる！」等々。

ただし、たまに「偽Slack」とか呼ばれる（悲しい）。

Slackのデザインを流用しすぎて、区別し辛いとの声も・・・。

- Slack否定派の人にこそ使ってもらいたいんだけど、あんまり利用されてないっぽい？

⇒ 薄々気付いていましたが、Slackを普段使わない人が、こういったツール積極的に使わないよね・・・。

所感

- 以前のスクラム開発で触ったDocker、Elasticsearch、FESS等の経験があったため、スムーズに開発が行えた。（デザインを除いた製造部分は3～4日程度）
- 検索部分をFESSに丸投げすることで、大分楽に構築出来た。全文検索を自前で作成するのは大変。
 - ⇒ Slackの投稿をそのままElasticsearchに投入すれば、分析・可視化が可能。
もう少しNoSQLを勉強して、検索の機能を自前で設けたい。

最後に

- Qiitaの「やってみた」系の記事単体では、参考程度で終わってしまうが、**組合せ + a**のアイデアで実用に耐えるツール、システムをお手軽に構築出来た！！
 - ⇒ 新しい技術や、トレンドへの興味が大事。
作成したツールが利用されればモチベーションアップで、良いサイクルに。
- 最初のアイデアはスクラム開発でのバズ・セッションから。フワツとしたアイデアを、「アレが使えそう」、「こうすればもっと簡単、便利」と議論していくのが楽しい。
 - ⇒ 緩い感じのDevOpsがおススメです。