

# Slack の情報保存について

糠塚元気

# Slack の料金改定とフリープランの制限変更



slack help center

## ワークスペースの管理

設定方法や管理方法について詳しく知りたい時は、ココをチェック！

はじめ方 Slack の使い方 プロフィール ツールと連携 **管理** チュートリアル

Slack ヘルプセンター > ワークスペースの管理 > 料金プランとお支払い設定の管理

### プロプランの料金改定とフリープランの最新情報

Slack が提供するサービスは、これまで何年もの間に拡張されてきました。その中には、さまざまなつながり方を実現する柔軟なツール、強固なセキュリティ機能、アプリのインテグレーション、ワークフローなどが含まれます。引き続きユーザー企業にとってうれしいイノベーションをお届けし、そのビジネスの成功に貢献するため、プロプランの料金を引き上げるとともに、フリープランにいくらかの調整を加えることになりました。この2つの変更は、2022年9月1日から有効になります。

## フリープランの変更

2022年9月1日から、[フリープランの使用上限](#)がシンプルになります。メッセージ数を1万件、ストレージ容量を5GBに制限する代わりに、過去90日間のメッセージ履歴とファイルストレージの利用を無制限にすることで、制限を気にせず皆さまのチームにSlackをご利用いただけます。[クリップ](#)および[メッセージとファイルの保存設定](#)などの機能もご利用いただけるようになります。今後の変更予定についての[よくある質問](#)をご覧ください。

- フリープランの制限
  - これまで
    - メッセージ数：1万件
    - ファイルストレージ：5GB
    - メッセージ履歴：無制限**
  - これから（2022/09/01～）
    - メッセージ数：無制限
    - ファイルストレージ：無制限
    - メッセージ履歴：90日間**
- 90日より古い情報は削除されるようになる
- 情報を取り出して保存しておく必要がある

# Slack のメッセージ保存

- 様々な方法で情報を保存できるらしい
  - 公式の機能

**ワークスペースのデータをエクスポートする**

誰がこの機能を利用できますか？

- ワークスペースのオーナーおよび管理者と OrG のオーナーおよび管理者
- すべてのプランで利用できます

**データのエクスポート**

フリープランとプロプラン | ビジネスプラスプラン

Enterprise Grid プラン

ワークスペースのオーナーと管理者は、次の方法でワークスペース内のパブリックチャンネルからデータをエクスポートできます。

- 1 デスクトップの画面左上にあるワークスペース名をクリックします。
- 2 メニューから「設定とその他管理項目」を選択し、「ワークスペースの設定」をクリックします。
- 3 「データのインポート/エクスポート」をクリックします。
- 4 「エクスポート」タブを選択します。
- 5 「日付範囲をエクスポートする」のドロップダウンメニューを開き、オプションを選択します。
- 6 「エクスポート開始」をクリックします。エクスポートファイルの準備ができ次第、メールをお送りします。
- 7 届いたメールに記載されている「ワークスペースのエクスポートページにアクセスする」をクリックします。
- 8 「ダウンロードを開始する」をクリックして zip ファイルにアクセスします。

zip ファイルには、すべてのパブリックチャンネルからエクスポートされた、ワークスペースのメッセージ履歴（JSON 形式）とファイルへのリンクが含まれます。

→ 糠塚はできなかった  
管理者権限が必要？

- サードパーティ製のツールを使う

**Slack を見た目そのまま HTML ファイルにバックアップする Chrome 拡張機能「SlatickHTML」**

■ SlatickHTML の機能

- ★ 閲覧したチャンネルの内容を自動でログに記録
- ★ ワンクリックでログから HTML ファイルを作成
- ★ Slack のスタイルをそのままバックアップ



→ やって見た

- ツールを自作する → 簡単なツールはできたが、完璧を目指すのは案外手間がかかる



# Slack のメッセージ保存：自作ツール

- ボットをユーザーとして追加し、ボットに対してコマンドを送信することで情報の取得ができる
- Python で簡単なプログラムを書いてみた

```
1 import os
2 import datetime
3 from slack_sdk.web import WebClient
4
5 client = WebClient(token=os.environ["SLACK_API_TOKEN"])
6
7 def GetUserName( message ) :
8     user = client.users_profile_get( user=message['user'] )
9     return user['profile']['real_name']
10
11 response = client.conversations_list()
12 conversations = response[ "channels" ]
13
14 print( "Channels:" )
15
16 for chan in response.get( "channels" ) :
17     print( " -", chan['id'], chan[ 'name' ], '\'' + chan[ 'purpose' ][ 'value' ] + '\'' )
18     if chan[ 'name' ] != 'bnl' :
19         continue
20
21     print( "history" )
22     retval = client.conversations_history( channel=chan['id'] )
23
24     counter = 0
25     for message in reversed( retval['messages'] ) :
26         timing = datetime.datetime.fromtimestamp( float(message['ts']) )
27         print( " *", timing, GetUserName( message ) )
28         print( "   ", message['text'] )
29         counter += 1
30
31     break
```

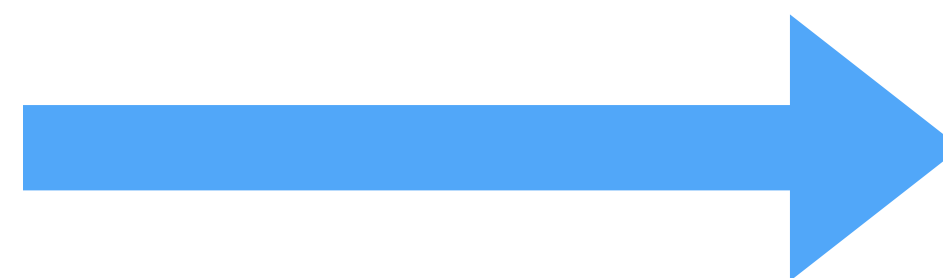


# Slack のメッセージ保存：自作ツール

- ボットをユーザーとして追加し、ボットに対してコマンドを送信することで情報の取得ができる
- Python で簡単なプログラムを書いてみた



テキストファイルに出力



```
[genki 11:35:38 SlackThreadStore] $ more temp.txt
Is OK? True
Channels:
- C02N8BS7BKR beamtest "この *channel* はプロジェクトの作業に使います。ミーティングの開催、資料の共有、チーム一体での意思決定ができます。"
- C02N89UEQ94 general "このチャンネルには、常にすべてのメンバーが含まれます。社内通知やチーム全体の会話にぴったりです。"
- C02NH8QS86Q random "このチャンネルでは、どんなことでも話せます。チーム内のジョークや、とっさに思い付いたアイデア、おもしろい GIF 画像を投稿できます。思いのままに活用しましょう！"
- C02Q3NW169F trackingの課題 ""
- C02Q9M3F5UY zoom_url ""
- C02SCQF19CH programming ""
- C03M5C9KVMM bnl "BNL での INTT に関連する話をここにまとめるといいと思います"
history
* 2022-06-30 09:41:03.864149 HIKARU IMAI
  FVTXのExpertGUIのソースコードってどこに保存されていますか？
* 2022-06-30 13:48:42.495259 藤塚元気
  BNL 生活準備に役立つと思って、写真を色々撮りました。

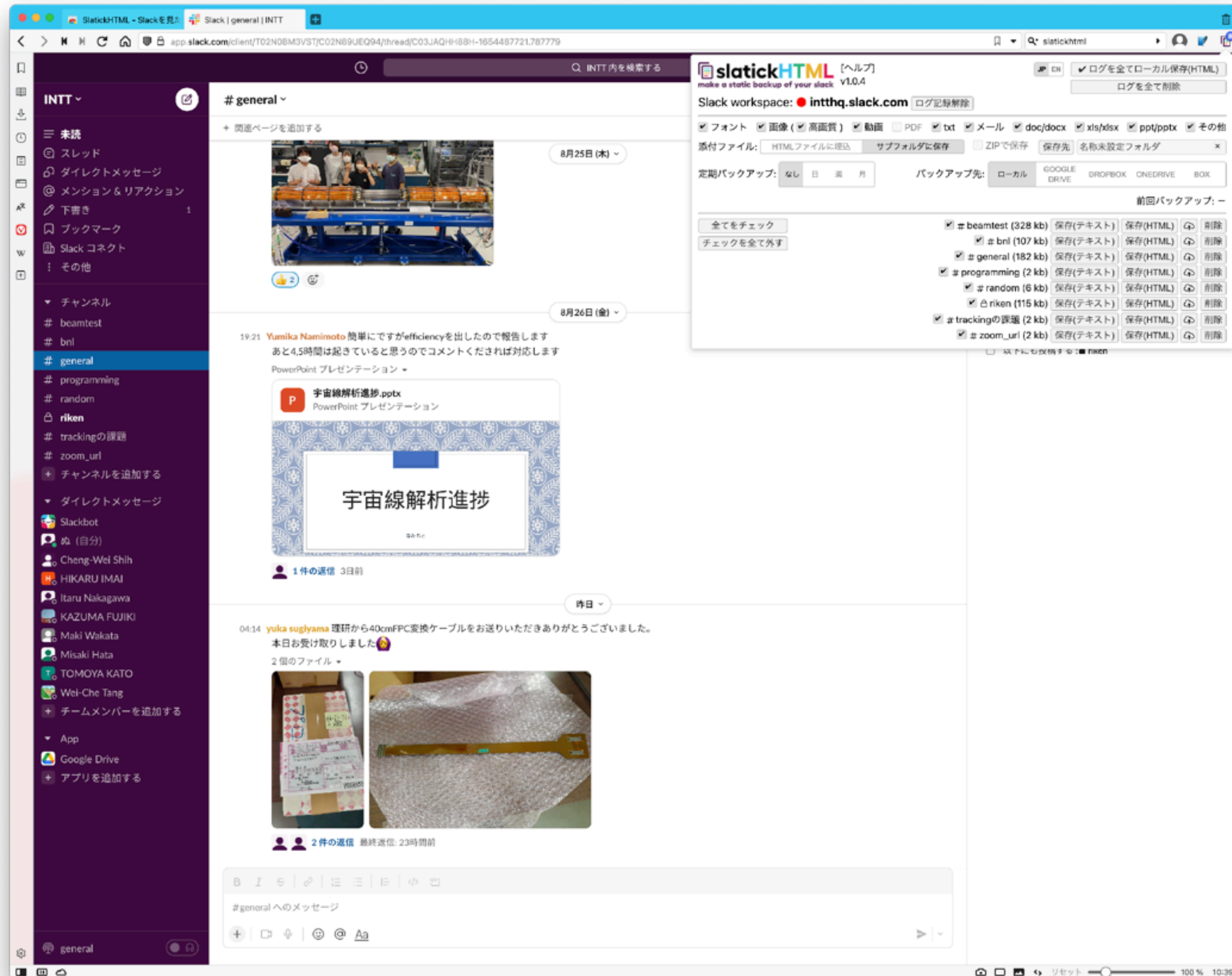
BNL students' room
<https://photos.app.goo.gl/98QHc8uwmUDgobxh7>
* 2022-07-01 09:09:32.354749 Itaru Nakagawa
  BNLのプールが来週オープン予定だそうです。60メートル。飛び込みあり(かなり深い)の日本では珍しいタイプのプールです。<https://www.bnl.gov/bera/pool.php>
* 2022-07-03 23:12:33.345879 藤塚元気
  BNL 内での wifi 接続に eduroam が使えます。設定しておくとかと便利なので、出国前にやっておくといいと思います。PC でもゲータイでもやれるはずですよ。
```

メッセージは保存できたが、添付ファイルの保存、スレッドの取り扱いなどいろいろやるべきことが多く残っている・・・





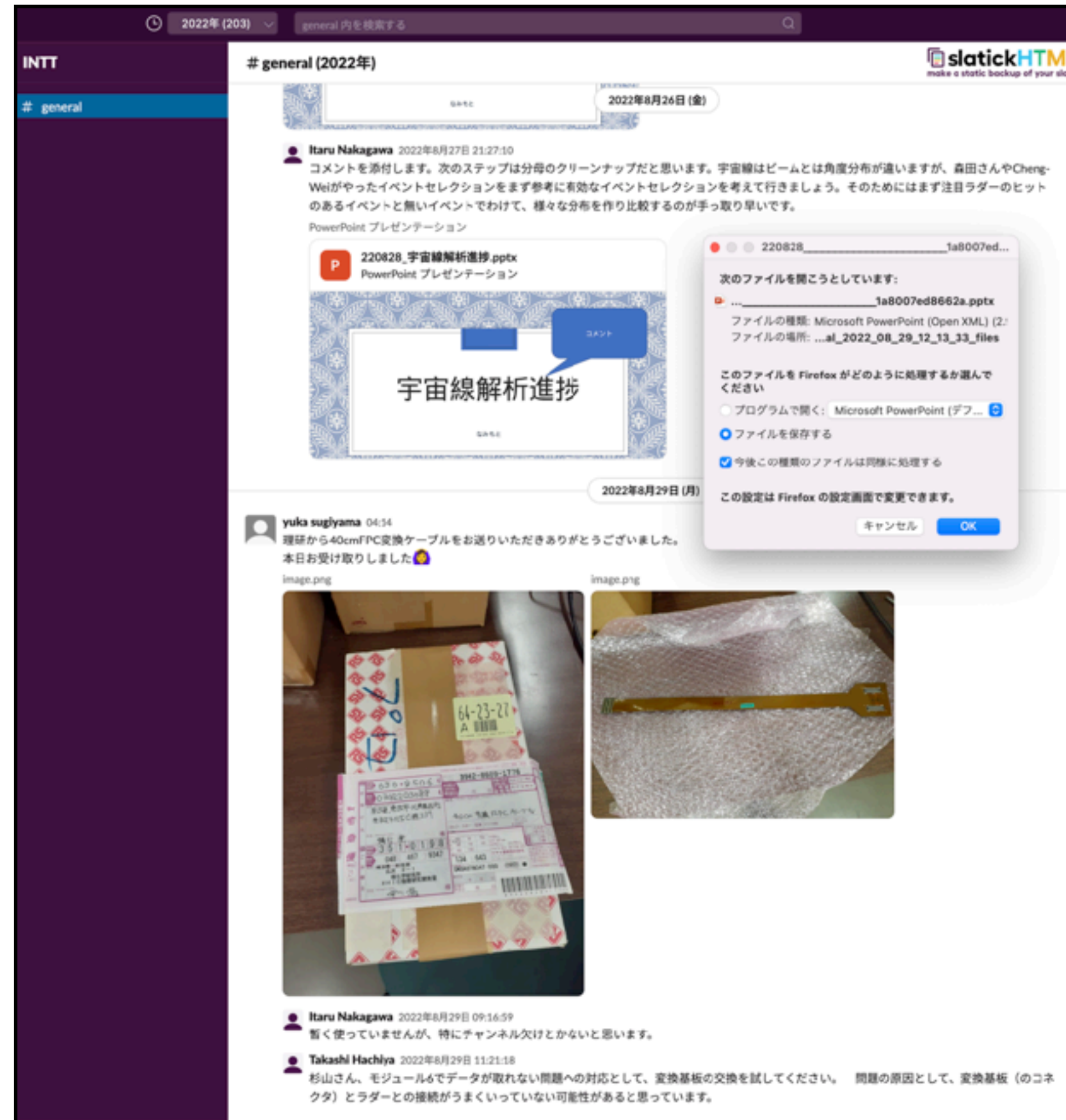
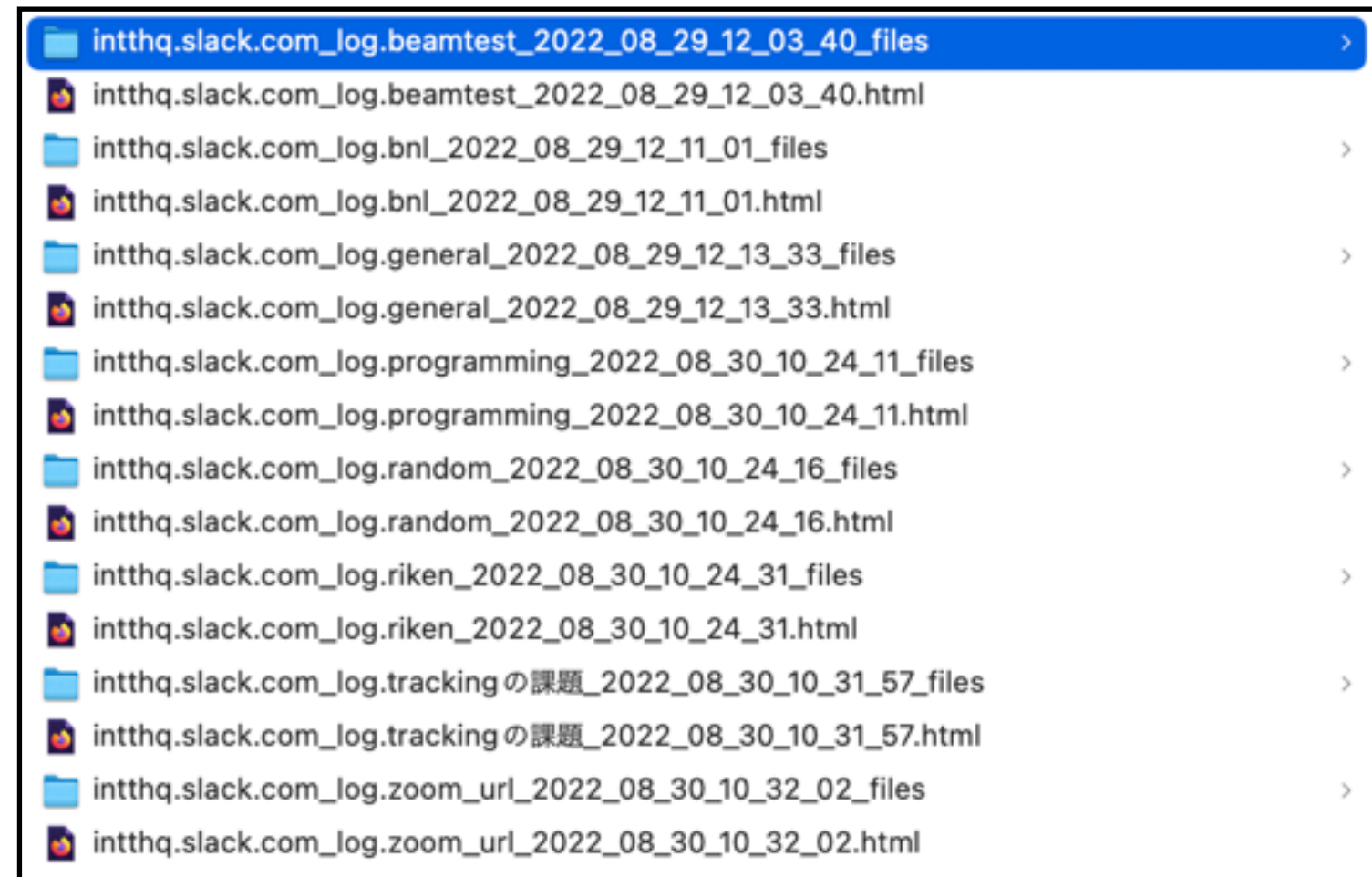
# Slack のメッセージ保存：サードパーティ



- Chrome (系ブラウザ) で Slack を開き、チャンネルの投稿を見る
- ブラウザのキャッシュ(?) に残った情報を使い SlatickHTML が情報を HTML ファイルにまとめて出力する
- 設定はブラウザ上で行う



# Slack のメッセージ保存：サードパーティ



一つのチャンネルにつきメインの HTML ファイルとその他諸々を保存するディレクトリが生成される

例：

<https://u.pcloud.link/publink/show?code=XZLrChVZRIPViTKGc2BpOmhCY3gkhjNiD6dk>



大体同じレイアウトでチャンネルが保存される  
添付ファイルにもアクセスできる



ブラウザで見た箇所を保存する。全情報を保存するには、全投稿を見る必要がある