

快適な制作環境の作り方

機材・ワークフローを整えて、強固な制作基盤をつくる

cumuloworks

自己紹介

Tomoya Eguchi (Cumuloworks)

Freelance Director

Motion Designer, CG Generalist

最近はちょっと Developer ?

- 2014 年 映像制作を始める
- 2017 年 大学に通いながらフリーランスに
- 2018 年 株式会社ナナメ 入社
- 2020 年 退社、独立してフリーランスに
- 2021 年 法人化



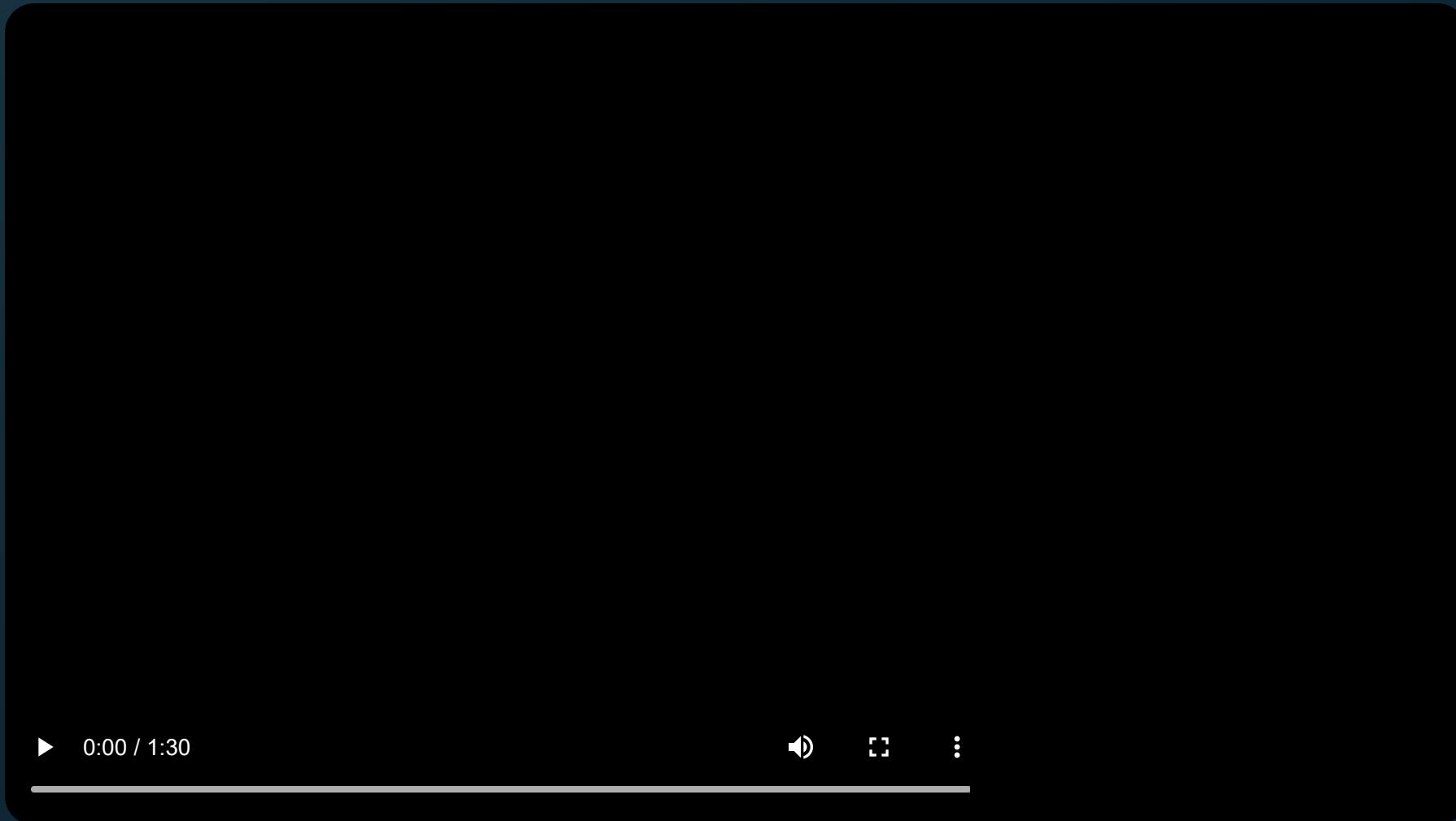
合同会社キュムワークス

Cumuloworks, Inc.

- 2021 年 9 月設立 (もうすぐ 4 期目)
- 広告・エンタメ双方の CG 映像制作がメイン
- 6 月からは[@shgumo](#)との 2 名体制

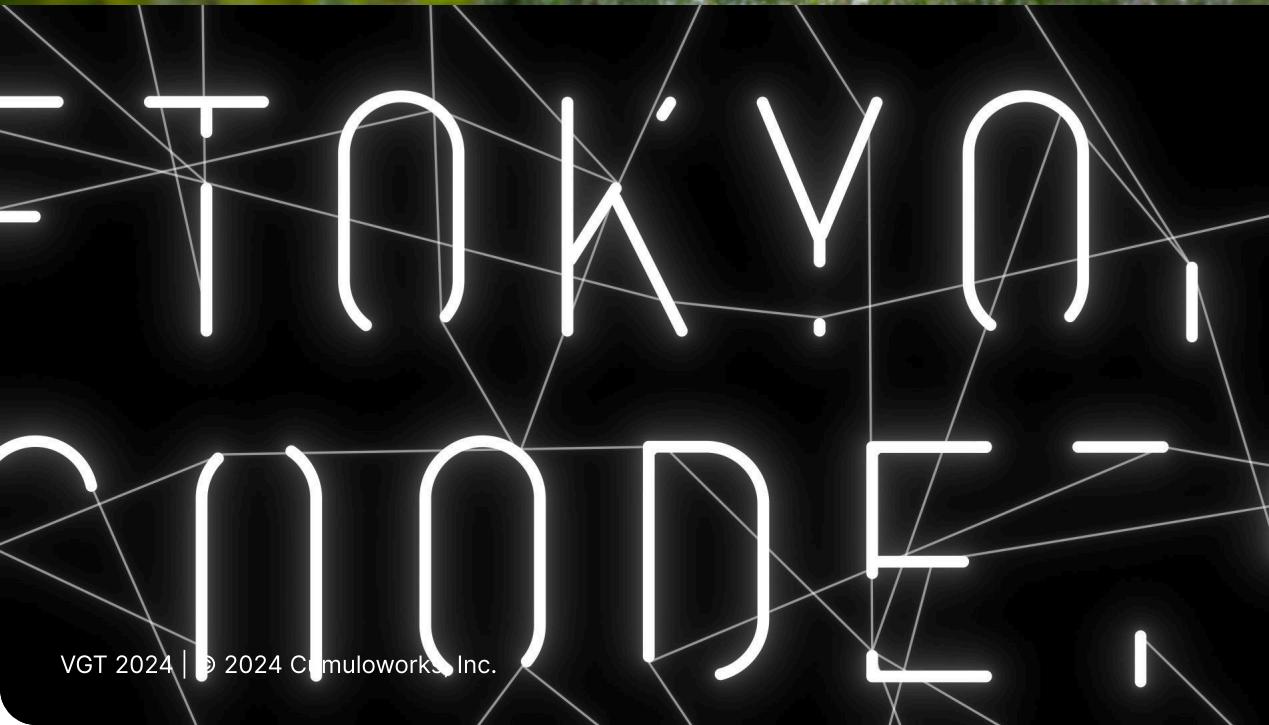


SHOWREEL 2024



▶ 0:00 / 1:30





本日の内容

1. 機材紹介

Cumuloworks, Inc. で導入している機材の紹介

2. ワークフロー紹介

機材をどのように制作に活かしているか

3. 今後の展望

最近興味があること、今後の拡張の計画（妄想）

質問の方法

質問は隨時受け付けています!

- 方法 1: Twitter の DM にて!
- 方法 2: マシュマロにて → 匿名が可能!



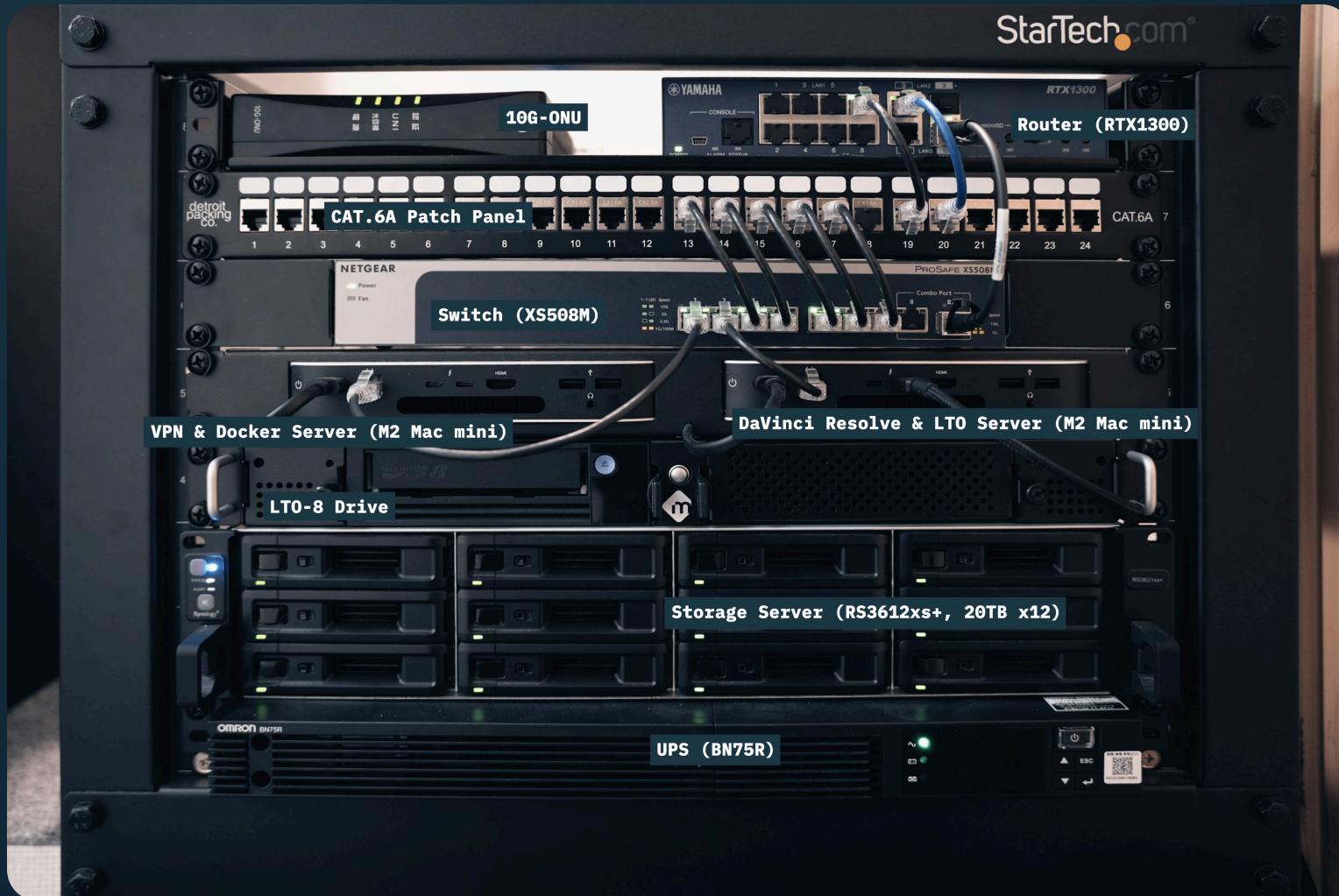
機材

ラック (StarTech 4POSTRACK8U)

2023 年に導入したラック

- 奥行きを調整できるタイプで、高さは 8U
- デスク周りに機材が散らばらず、コンパクトに収まる
- 見た目がクールだが、あらゆる付属品の価格が高い

ラック構成



メインルーター (YAMAHA RTX1300)

NTT 光クロス(10Gbps) を引き込んでいる

- NTT のレンタルルーター(**XG-100NE**)よりも高い安定性
- **高スループットの回線**を契約することにより、データのやり取りでストレスが減った
- インターネットに直に接する機材 → セキュリティ面で重要な役割

💡 Mac アドレスを基に、主要な LAN 内デバイスの IP アドレスをルーター側で固定



メインスイッチ (NETGEAR XS508M)

8 ポートのシンプルな オール 10GbE スイッチ

- 排熱も良好で、本格的に 10GbE 化を始めたい方にはおすすめ
- ここから各作業用マシン・サーバーなどへネットワークが分配される
- 8 ポートで足りず、買い替えを検討中



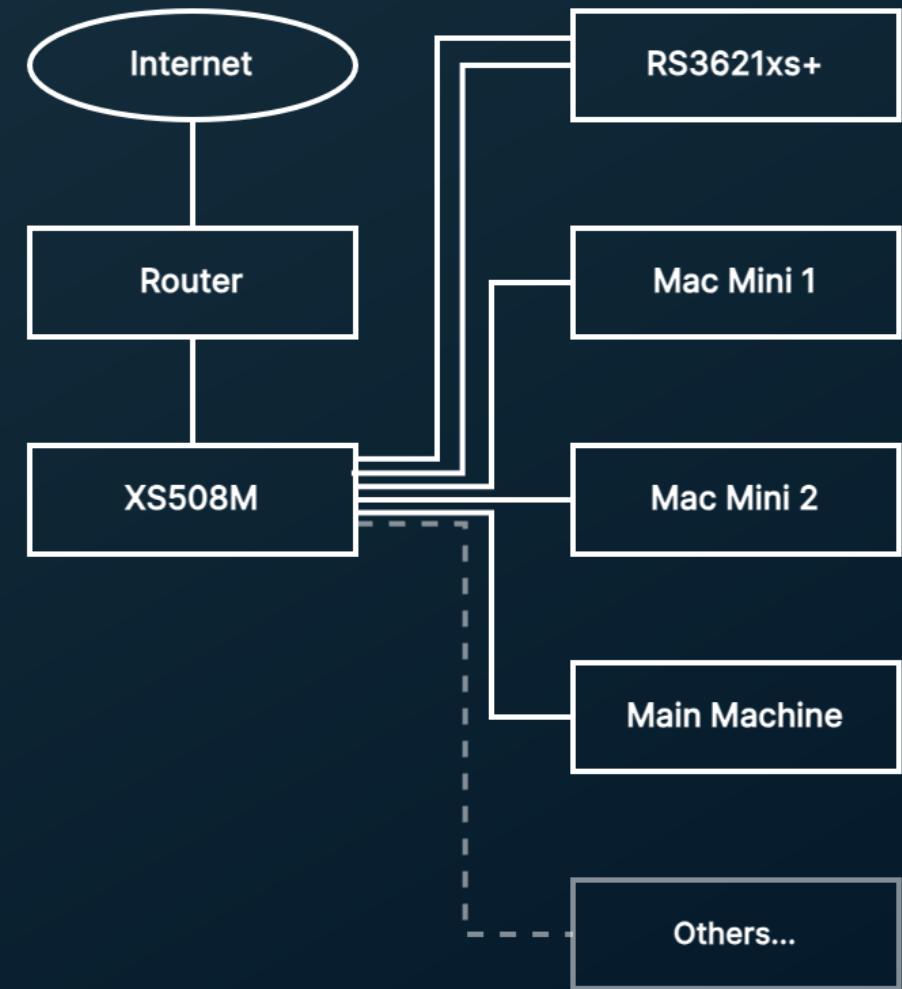
UPS (OMRON BN75R)

ラック全体の電源をバックアップする、最大 680W 対応の UPS

- 現環境の高負荷時で、10 分程度は電力を維持可能
- 前面の液晶ディスプレイで、電力消費量やバッテリー残量などが確認できて便利
- NAS と USB 接続することで、電力喪失時に安全にシャットダウンされる
- 商用電源が不安定になっても、安全に電力供給できる（見落とされがちなメリット）
 - 電力逼迫で、電圧低下や周波数不安定化が起こることは今後もありそう

ネットワーク構成

- すべて 10GbE (Cat.6A) 接続
- 10GbE スイッチを介して相互接続
- NAS は 20GbE 帯域を確保
 - 理論上、Internet ↔ NAS と、各マシン ↔ NAS それぞれで 10Gbps の帯域を確保
- RJ45 にしている理由
 - コスト面 + ケーブルの取り回しに気を使う
 - 結局 RJ45 に変換する必要がある



※簡略化したイメージ

メインストレージサーバー (Synology RS3621xs+)

2023 年導入の **200TB** のメインサーバー
ストレージ以外の機能も集約

Intel® Xeon® D-1541 8-core 2.1 GHz

RAM: 32GB DDR4 ECC RDIMM (2× 16GB)

12× 20TB Ultrastar DC HC560 (7200rpm)

+

2× 800GB Synology SNV3510 NVMe SSD (Cache)

SHR-2 (RAID 6) ... 2 ディスク障害耐性

 データセンター用の HDD を採用





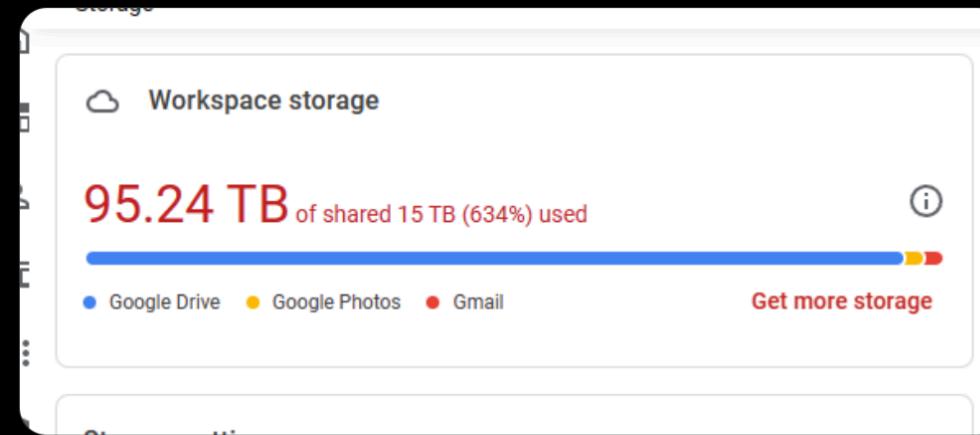
Cumuloworks
@cumuloworks

Promote

...

え、Google WorkspaceのEnterprise版、いつの間に100TB上限追加されたの、．．．．．．

Translate post



11:45 AM · Jan 25, 2023 · 4,716 Views

View post engagements



3



3



23



2



Post your reply

Reply



Cumuloworks @cumuloworks · Jan 25, 2023

追加のストレージは、405,000円/10TB*月らしい。無理過ぎる。。。



2



5



5



1.6K



↑



Add another post

なぜオンプレ化を推し進めるのか (メリット vs リスク)

メリット

- 常にデータが手元にある安心感
 - TB 単位のデータは、クラウドにアップロード・ダウンロードするだけで膨大な時間がかかる
- サードパーティーのサービスへの依存から脱却できる
 - 急なサービス内容の変更や価格改定による影響を受けない
- たのしい・クール

デメリット (リスク = 対策が必要)

- ランサムウェア・物理的な盗難など、セキュリティ面での不安
- 災害などによるデータ損失リスク
- ストレージなどの機材の購入・管理コストがかかる

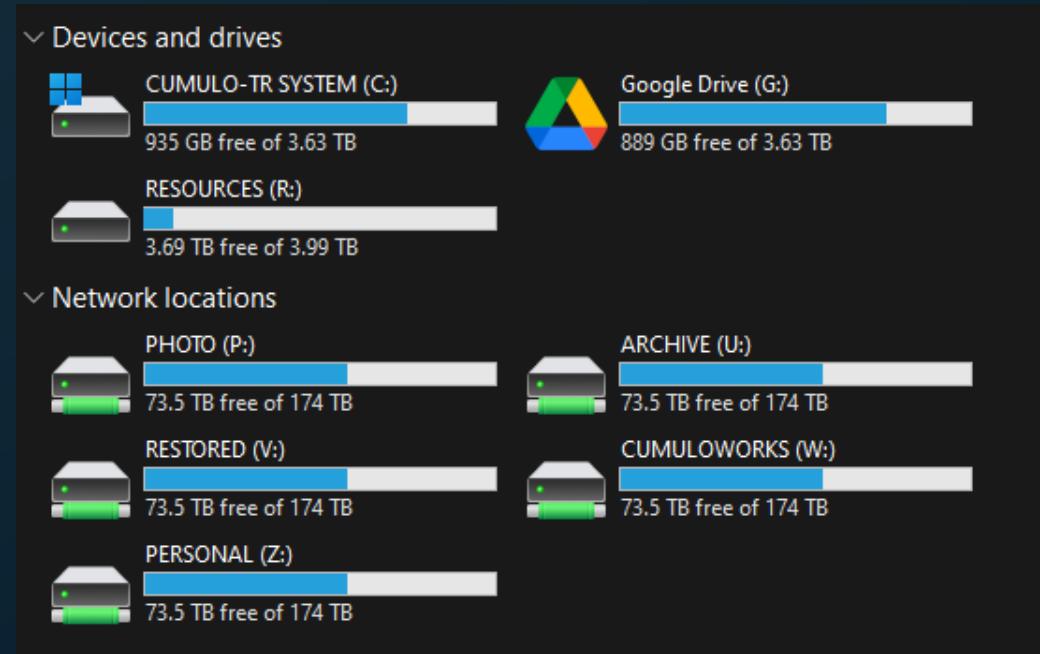
 Synology のようなパッケージで、デメリットを最小化



1. ストレージサーバーとして

プロジェクトファイル・アセットなどを集約

- SMB プロトコル + 10GbE 接続で、高速なファイルアクセス
- 社内でデータをリアルタイム共有(同期のラグなし)
- アクセスログ・バージョン履歴の保存(Synology Drive)

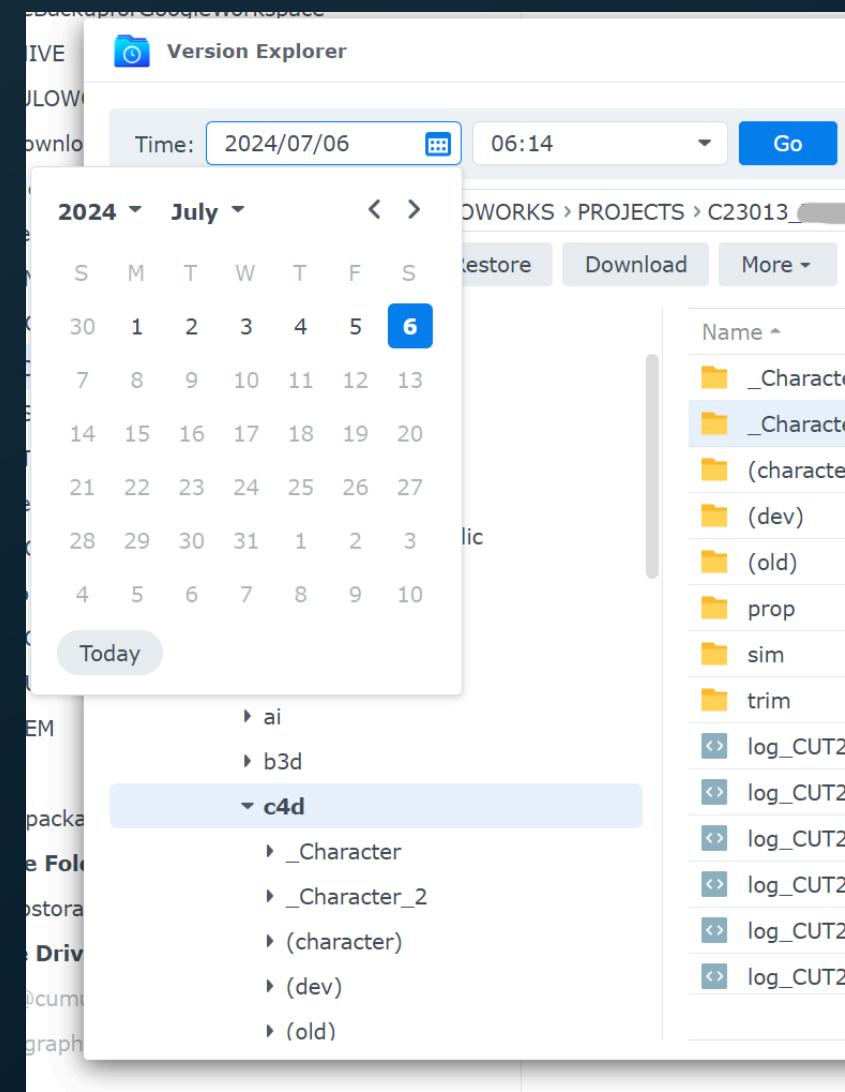


2. バージョン管理 (Version Explorer)

すべてのフォルダ・ファイルの変更を保存し、削除したデータも含めて任意のタイミングに遡ることができる
(Synology Drive)

- 間違えてファイルを上書きしてしまった
- 途中からプロジェクトファイルが壊れた
- 前のレンダーデータが必要になった

 Intelliversioning 機能によって、最低限の容量で差分バックアップ



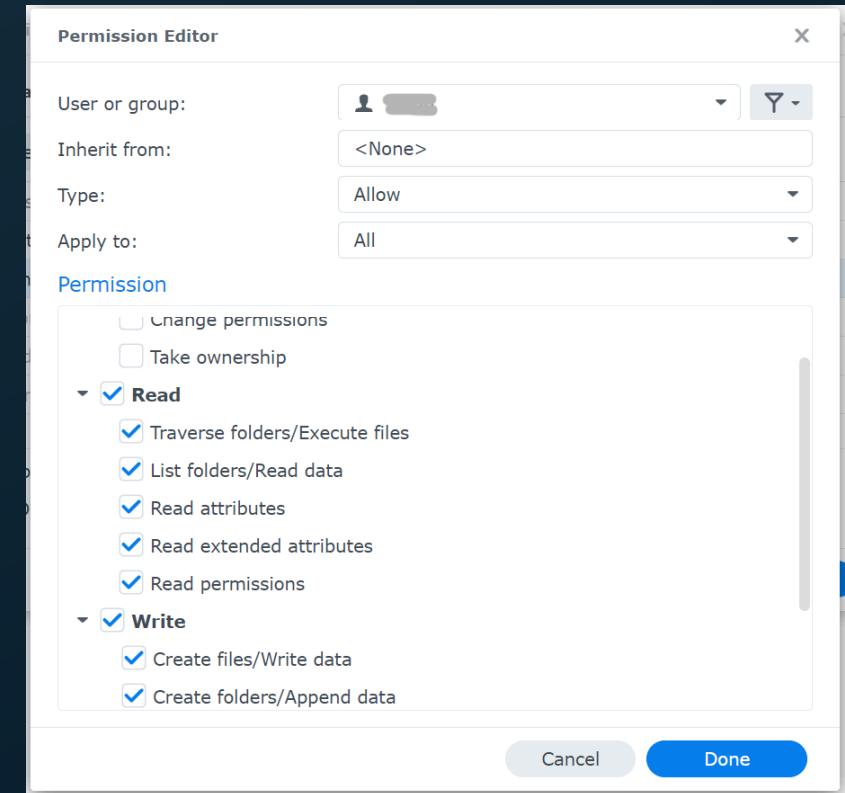
3. 社外とのファイル同期

Synology Drive を使って、NAS 上のフォルダ・ファイルを社外の共同作業者に同期してもらう

- オンプレミスの Dropbox のような感じ
- ユーザーとファイルの組み合わせで、細かく権限を設定可能

使用例

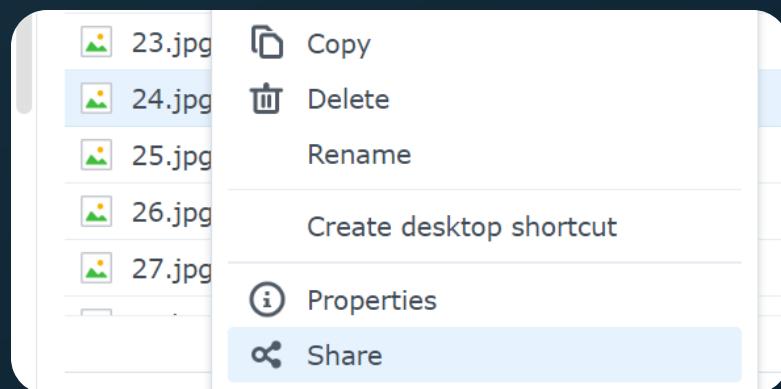
- 社内で制作したモデルデータをリアルタイムで受け渡し
- 社外で作業してもらったデータをアップロードしてもらう



4. 社外とのファイルのリンク共有

ファイルの共有リンクを発行して、外部の人にファイルをダウンロードしてもらう

- オンプレの Gigafolder 便のような感じ
- フォルダごと・ファイルごとにすぐにリンクを発行して共有できるので便利
 - e.g. エディターと共同でオンライン作業をする場面などで、フォルダから最新ファイルを隨時ダウンロードしてもらえる
- 有効期限やパスワードの設定も可能



5. その他機能(仮想化系)

1. Docker コンテナの運用 (**Container Manager**)
2. Windows や Linux のバーチャルマシンの運用 (**Virtual Machine Manager**)

応用例

- Windows で時間がかかる処理を、NAS のバーチャルマシンで行う
- 社内ツールの Web サーバー・SQL サーバーとしての利用
- **PhotoPrism**のサーバーとして運用
 - ミラーレスカメラで撮影 → FTP で直接 NAS にアップロード → PhotoPrism で管理





Mac Mini サーバー (Mac Mini (M2))

10GbE オプションの Mac Mini (M2) を 2 台導入

ラックマウント化し、ディスプレイなしで運用(Parsec
でリモートアクセス)

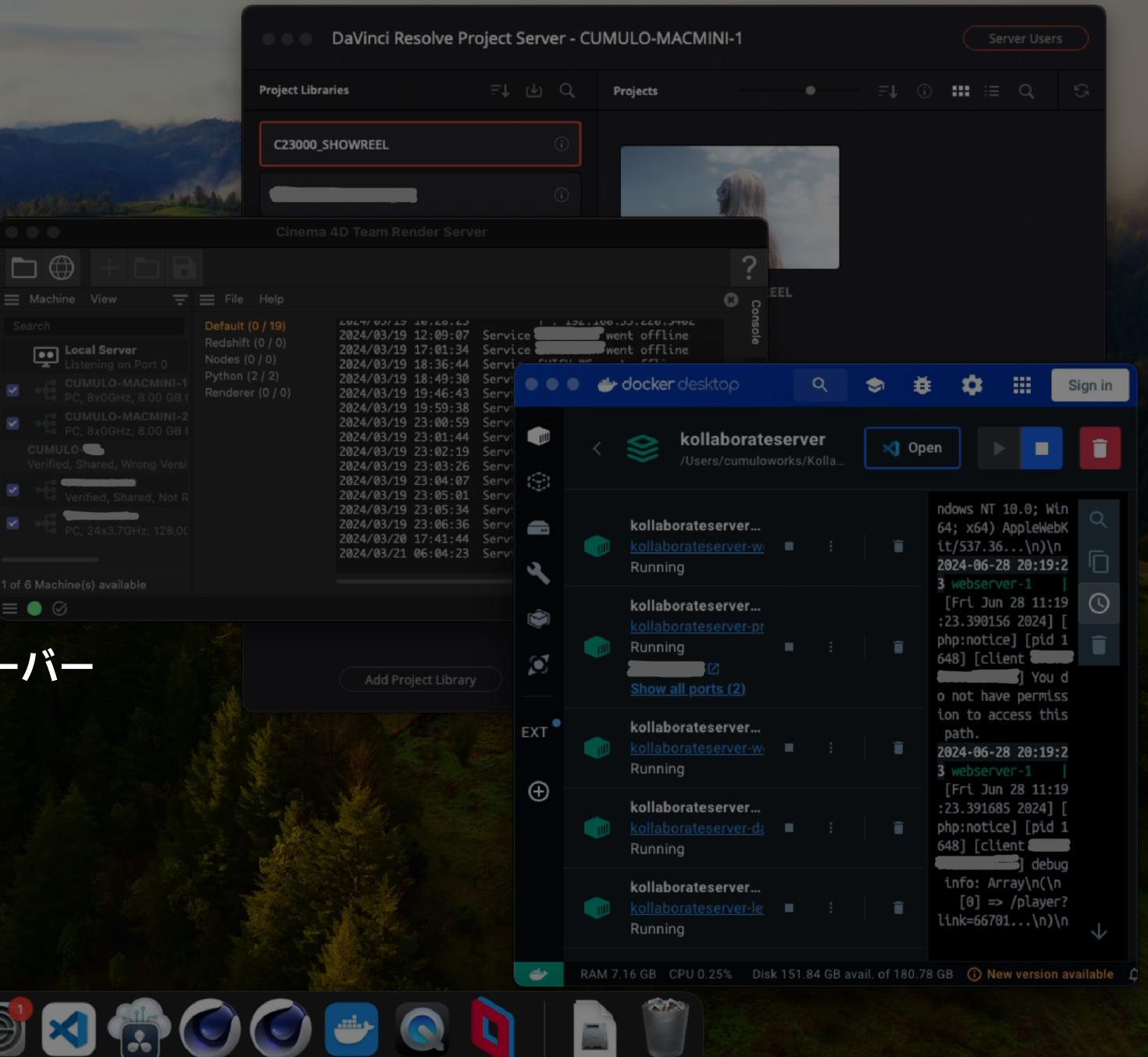
導入のきっかけ

- Mac 環境はやっぱり手元に欲しい
 - たまに送られてくる Mac フォーマット(HFS+)
の HDD への対応
 - ソフトウェアの動作検証がしたい
 - 常時起動のサーバーとして運用してみたい
- Mac Mini だけコストパフォーマンスが異常
 - 当時 10 万円を切っていた

Mac Mini 1台目

安定動作が期待されるサーバー機能を集約

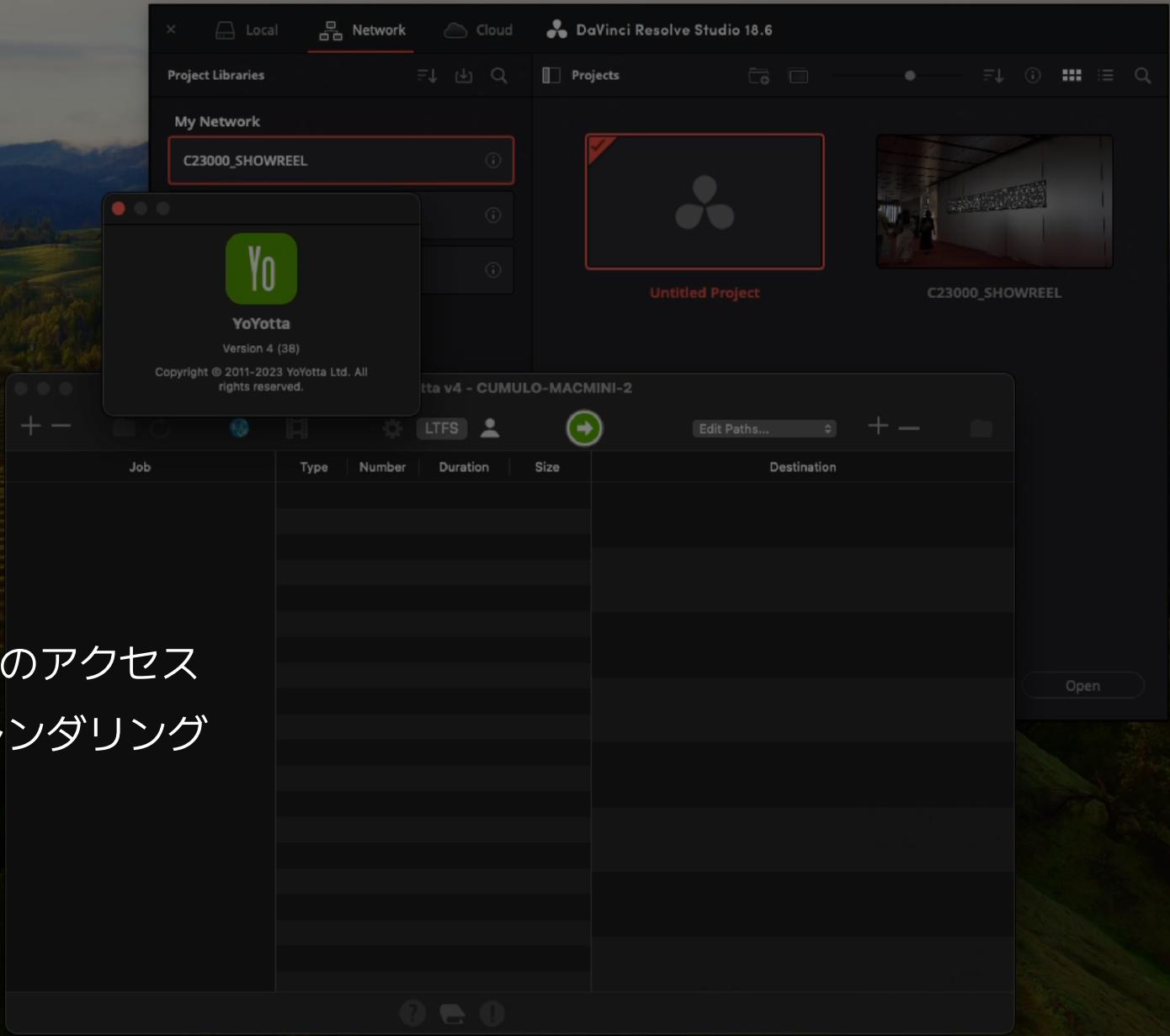
- Docker コンテナの運用
 - **Kollaborate** のサーバー
- **OpenVPN** サーバー(運用休止中)
- **DaVinci Resolve** の データベースサーバー
- **Team Render** の サーバー



Mac Mini 2台目

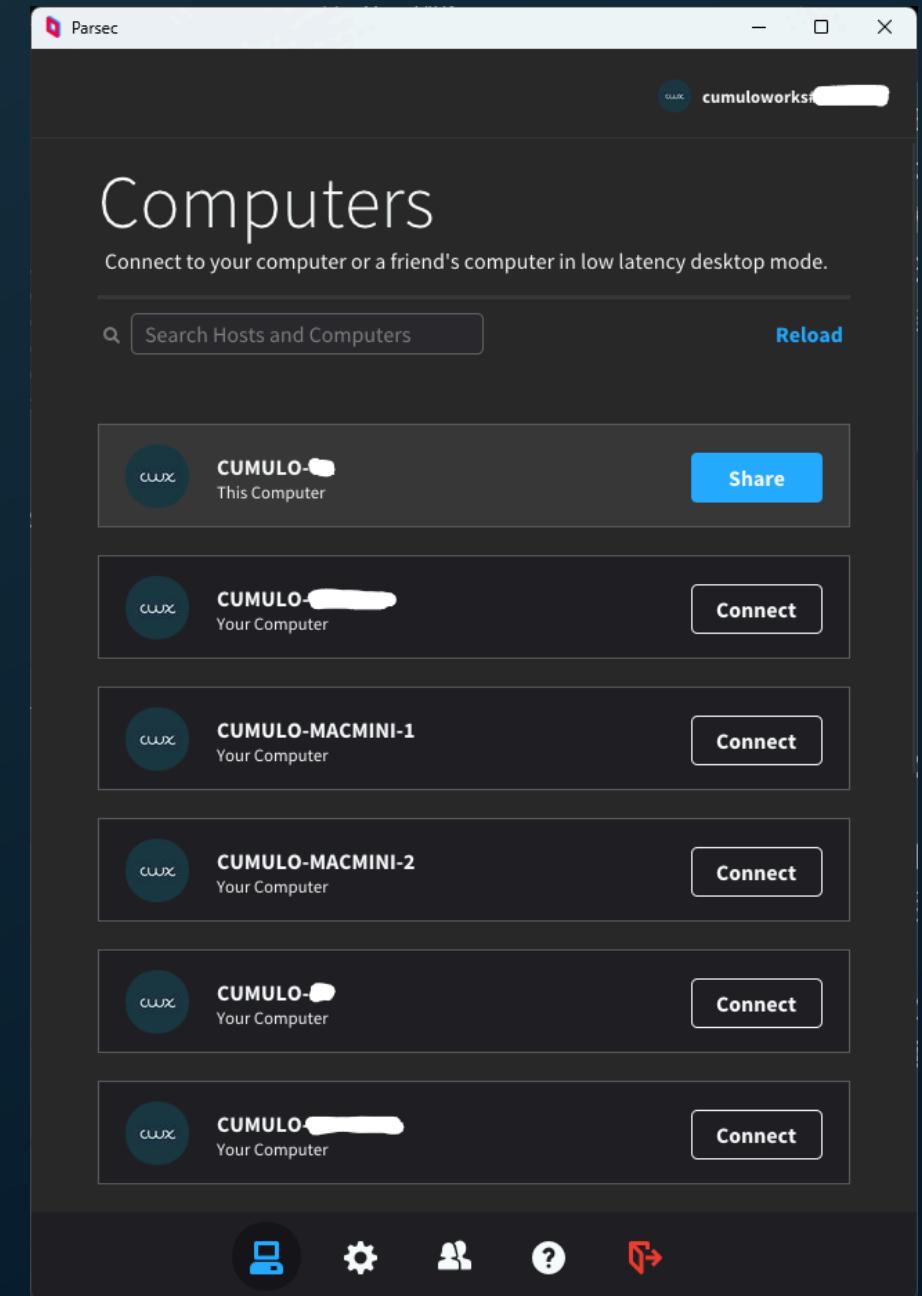
Mac 環境必須のアプリケーションなどを実行

- Mac 環境でのツールなどの検証
- Thunderbolt 接続された LTO ドライブへのアクセス
- DaVinci Resolve でのリモート ProRes レンダリング



リモートデスクトップ環境 (Parsec)

- リモートデスクトップで、運用しているマシンにアクセス可能
- ラックマウントの Mac へもリモートアクセスで、KVM 的に使う
- 遅延が小さく、広色域・固定フレームレートにも対応していて、映像作業にも使える
- Chrome Remote Desktop** も併用して二重化している



メインマシン

- 2024 年始めに導入した Threadripper マシン
 - AE での編集作業がストレスなく行えるよう に、メモリを 384GB に
- 現状 VRAM が 24GB で足りないので、将来的に換 装予定

CPU: AMD Ryzen Threadripper 7980X
M/B: ASUS Pro WS TRX50-SAGE WIFI
GPU: MSI GeForce RTX 4090 SUPRIM LIQUID X
RAM: Kingston 384GB (4x DDR5-5600 RDIMM ECC 96GB Micron Die)
SSD: 2x Nextorage 2TB NVMe SSD PCIe Gen5×4
PSU: SUPERFLOWER LEADEX VII GOLD 1300W
CPU_FAN: Arctic Freezer-4U-M
CHA: Geometric Future Model 4 Caliburn
CHA_FAN: 3x Thermaltake TOUGHFAN 12 Pro



レンダリングマシン

- 以前まで使っていた水冷のメインマシンを小型化したもの
- 電力的なコストパフォーマンスは悪いが、まだまだ現役
- 見た目がかなり気に入っている、史上最高傑作

CPU: AMD Ryzen 9 5950X
M/B: MSI Prestige X570 Creation
GPU: 2x Zotac GeForce RTX 3090 Trinity
RAM: Kingston 384GB (4x DDR5-5600 RDIMM ECC 96GB Micron Die)
SSD: Corsair Force Series MP600
PSU: SUPERFLOWER LEADEX VII GOLD 1300W
CHA: Jonsbo TK-1 White
CHA_FAN: 2x Noctua NF-F12 industrialPPC-3000 PWM
CHA_FAN: 2x Noctua NF-A12X15
WATER COOLING: EKWB

ワークフロー

進行管理・コミュニケーションツール

- 社内では基本的に **Notion, Discord** に集約
- Discord は、チャネルごとに細かい権限設定が出来て、複数プロジェクトが同時に動くような状況でも使いやすい
- その他、プロジェクトに合わせて柔軟に対応

基本ルール

- ファイル・フォルダ名は半角英数字と、本ドキュメントで指定する記号 `()-_` のみを使用する。
- 本ドキュメントで指定する例外を除いて、2バイト文字は使用しない。
- アプリによって自動生成されるフォルダ・ファイル名、及びキャメルケース使用時以外は、全て小文字で命名する。ただし、シーン名などで A, B, C を使用する場合は必ず大文字とする。
e.g. `scene_A` (not `Scene_A` nor `scene_a`)
- クライアントから受け取ったファイル・フォルダ名については、変更せずそのまま使用する。2バイト文字によって支障がある場合は、コピーして英数字で命名し直す。
- 各階層において、使わなくなったファイルやフォルダがある場合は、その同階層に `(old)` フォルダを作って移動する。
- ISO 形式に従い、日付は `yyyy-mm-dd` 、時刻は `yy-mm-ddThhmmss` (: はファイル名として使用できないため省略する)で命名する。
e.g. 日付: `2025-01-31` , 時刻: `2025-01-31T153049`

- 命名は<https://www.abbreviations.com/>などを参考に、**分かりやすさを損なわない範囲**で極力短くする。一般的でない省略法を使う場合は、単語の最初を抜き出すほうが好ましい。（よく使用するものはドキュメント最後に記載）

do: `development` → `dev` `schedule` → `sched`

don't: `shading` → `sdg`

- フォルダ階層は可能な限り浅く構築し、フォルダで細かく分類するよりは検索機能を使うことを優先する。

プロジェクト ID

- 各プロジェクトにはアルファベット1文字+受注年（西暦下2桁）+通し番号（3桁）を付与する。
e.g. Cumuloworks, Inc.で 2025 年に受注した 5 番目のプロジェクト = `c25005`
- この ID は、プロジェクトフォルダ名・請求書 ID などとリンクしている。
- 既存のプロジェクトの継ぎとなるプロジェクトが発生した場合に新規の ID を付与するかどうかは、既存のプロジェクトの内容をどれくらい継承するかを鑑みて判断する。※軽微な修正等の場合は以前の ID をそのまま使用してもよい。

プロジェクトフォルダ

ファイルの命名規則・整理のルールなど

- すべてのプロジェクトでファイルの命名規則を統一
- 最近、ルールを明文化したGithub で公開中

カラーマネジメント

- ハードウェアキャリブレーション可能なモニターを導入
- 最近では、ACESへの統一を進めている
- After Effects の新しい OCIO 機能も活用している

各ソフトウェアのカラースペース

- Octane Render: ACEScg
- After Effects: ACEScg でコンポジット、ACEScct で書き出し
- DaVinci Resolve: ACEScct
- 納品: 基本的には Rec.709 で、必要に応じて他のカラースペースに変換



After Effects でタイムコードのオーバーレイの作成

- 各ショット・フレームを正確に把握するために独自のオーバーレイを導入
- 日時・マシン名・タイムコードなどがすぐ把握でき、CG ソフトとのやり取りや、エラーフレームの除去などで活用
- プレビズやオフラインデータを起点として制作する場合、特に有用
- 16:9 の上下に余白を作る形で挿入している



REC TC/Fm: 00:04:01:02/05786

angeilc_v16 | CUT340

C3080.MP4 TC/Fm: 09:30:22:05/821333

Record TC

Project File Name

Composition Name

Source TC



Current Time Machine Name

2024-07-07 T 16:06:00 +09:00 CUMULO-TR

cumuloworks

Frame Range (C4D)

05774 f - 05934 f (00012/00160 f)

スプレッドシートでカットの情報を一元管理する

- オーバーレイと同時に、スプレッドシートで各ショットの進捗状況を把握
- 複数人で作業する際に効果を発揮（レンダリング作業の分担など）

	A	B	C	D	E	F	G	H	I	O	AA	AC	AD	AF	AG	AH	AI
1		IMG	START	END	DUR	DUR+48f	RESOLUTION	STILL	TAKE NAME	HOUDINI	CINEMA4D	RNDR				AE RENDER	
2										SIM	CRITICAL ISSUE	ORBX EXPORT	RNDR UP	RNDR Q	RNDR DL	RNDR ERR CHK	
4	CUT020		0728F	0857F	0130F	0178F	2560x1440	0740F	E002	C							5/13 OK
5	CUT030		0858F	1040F	0183F	0231F	2560x1440	0890F									5/13 OK
6	CUT040		1041F	1168F	0128F	0176F	1920x1080	1160F									
7	CUT050		1169F	1269F	0101F	0149F	1920x1080	1210F									
8	CUT060		1270F	1358F	0089F	0137F	1920x1080	1345F	D004	C		5/7 READY	5/13 OK				
9	CUT070		1359F	1445F	0087F	0135F	1920x1080	1410F	A002	5/05 COMPLETE		5/7 READY	5/7 READY			LOCAL RENDER	
10	CUT080		1446F	1614F	0169F	0217F	1920x1080	1540F		5/05 COMPLETE		5/7 READY	5/7 READY			LOCAL RENDER	
11	CUT090		1615F	1753F	0139F	0187F	1920x1080	1700F									5/13 OK

レビューシステム

Docker で動作する **Kollaborate** を導入

- オンプレミスで動く frame.io みたいな
- レビューのコメントを DaVinci Resolve と EDL で連携
- クライアントとの共有でも有用

 CM

Version 18 ▾

Files

Dashboard

Files

Team

Tasks

Share

Sessions

Workflows

CM

File preview: A woman with long blonde hair, wearing a blue flower in her hair, is shown from the back, looking up at a bright sky filled with falling petals or confetti. She is wearing a light-colored top.

00:03:45:07 / 00:05:11:17

Download Share Subscribe Favorite Actions

Information Metadata Users History

Date: May 27 2024 15:53:52

Uploaded by: Tomoya Eguchi

Views: 30

Downloads: 1

Comments Transcript

Type a comment and press Enter to submit

Tomoya Eguchi 00:00:40:01 a month ago

Tomoya Eguchi 00:01:20:03 a month ago

Tomoya Eguchi 00:01:41:22 a month ago

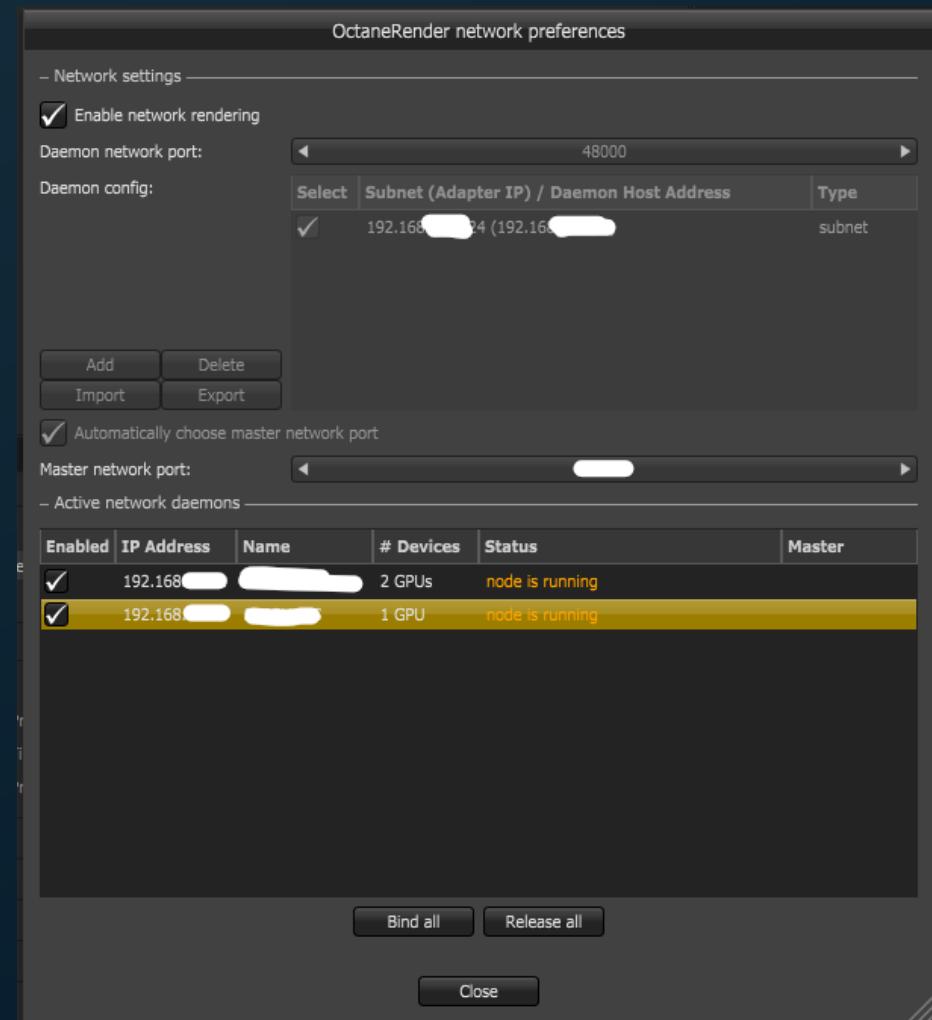
Tomoya Eguchi 00:02:27:04 a month ago

11.96 GB 40.03 GB

CG レンダリング

GPU (Octane Render)

- ほぼ全てのプロジェクトで Octane Renderer を使用
- 基本的には **RNDR** を使用したクラウドレンダリング
- プレビューや、短時間のレンダリングは Render Node を使用
- それぞれのマシンの GPU をネットワーク経由で利用可能簡単にスケールできる

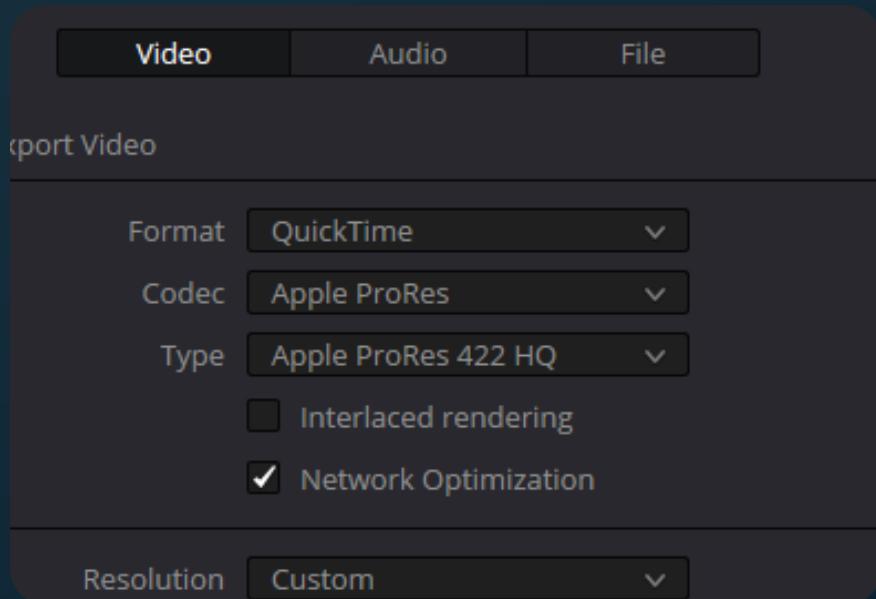


CPU (Cinema 4D Standard など)

- Cinema 4D の Team Render を使用することも

編集・納品

- 編集は主に DaVinci Resolve を使用
- Mac Mini サーバーで運用中のデータベースサーバーを使って、複数人でタイムラインを共有できる
- Mac でリモートレンダリングを設定することで、Windows マシンから ProRes でレンダリングできる



バックアップ体制

バックアップの考え方

データの削除は行わず、永続的に保存することを基本方針としている

ファイルを 3 つのカテゴリに分けて考える

1. 進行中のプロジェクト
2. 終了直後のプロジェクト
3. 終了して一定期間が経過したプロジェクト

1. 進行中のプロジェクト

- メインサーバーの HDD は、RAID6 (SHR-2) で構成され、2 つのディスク障害に耐えられる
- メインサーバーの障害時に備え、オフサイトサーバーにリアルタイムでバックアップを取る
- プロジェクトが長期化する場合は、定期的に LTO-8 テープにバックアップを取る

参考: 3-2-1 ルール

- 3 つのコピー (e.g. メインサーバー + オフサイトサーバー + LTO-8 テープ)
- 2 つの異なるメディア (e.g. HDD + LTO-8 テープ)
- 1 つのオフサイト (e.g. オフサイトバックアップサーバー, LTO-8 テープは分散保管)



オフサイトバックアップサーバー (Synology DS1618+)

- メインサーバー導入前に使用していた 6 ベイ NAS を流用 (40TB)
- 進行中のプロジェクトなど、重要なデータをメインサーバーとリアルタイム同期
- メインサーバーが使えない状況でも最低限のデータにアクセスできるようにしている

**Intel Atom C3538 4-core 2.1 GHz
RAM: 16GB DDR4 ECC UDIMM**

6× 8TB WD Red

2. 終了直後のプロジェクト (ホットアーカイブ)

- 終了直後のプロジェクトは容量が許す限りは、メインサーバーで保存
- オフサイトサーバーにバックアップする代わりに、LTO-8 テープにバックアップを取る
- 完パケの動画ファイルのみ別途フォルダに保存し、いちいち LTO からリトリーブする必要が無いようにしている

3. 終了して一定期間が経過したプロジェクト (コールドアーカイブ)

- 終了して一定期間が経過したプロジェクトは、メインサーバーから削除
- LTO-8 テープのコピーを 2 つ作り、別々の場所に保管する

LTOについて

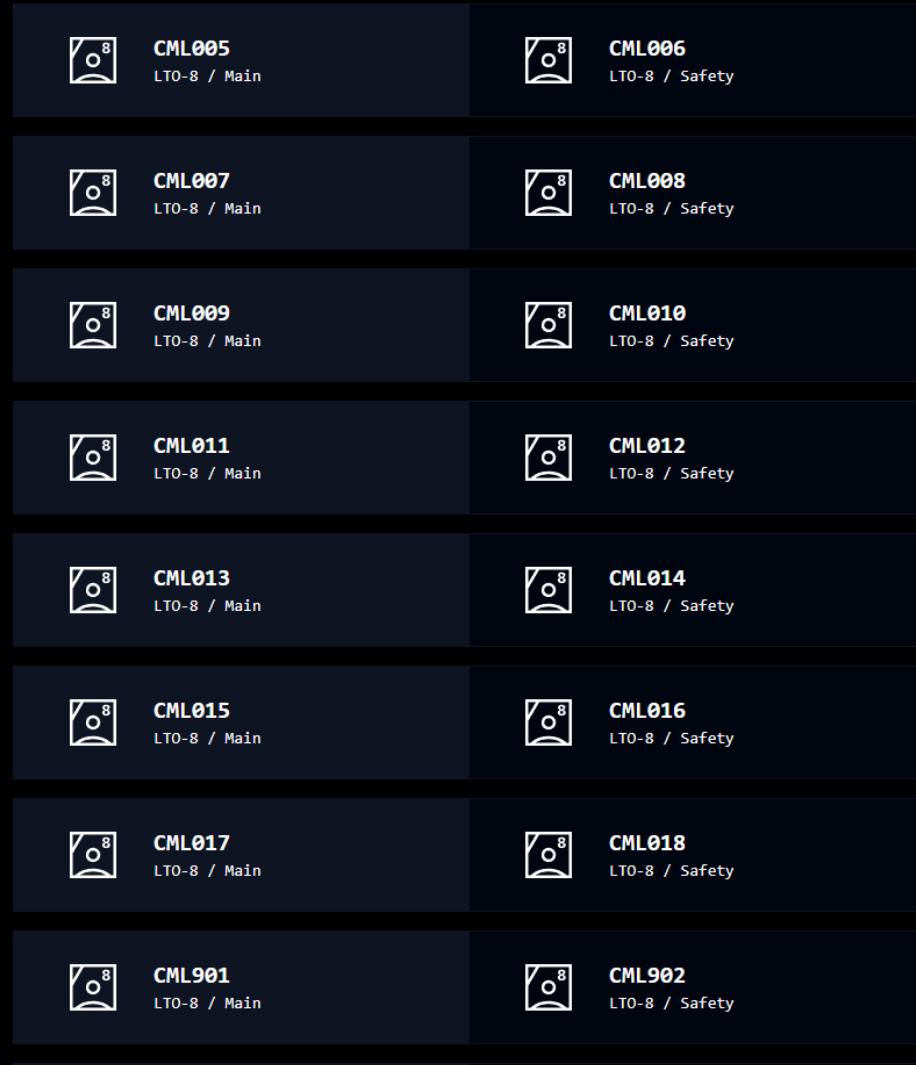
- LTOは磁気テープ型のデータストレージで、HDDと比べて耐久性が高く、データの保存期間も長い
- 頻繁なアクセスが必要なデータには向かないが、アーカイブ用途には適している
- 完全にオフラインで保存することができるため、ランサムウェアなどのセキュリティ対策としても有効と思われる



LTO の使用方法・メンテナンスなど

- LTO の保存条件は、温度・湿度・磁気の影響を受けやすいため、適切な環境で保管する必要がある
- 温度は、基本的な日本の屋内であれば問題ないが、湿度が高い部屋の場合はドライボックスなどを使用するのが良さそう
- データが消える程の磁気の影響は、普段の生活では問題ないと思われる
- クリーニングカートリッジというのがあり、ドライブでクリーニングのサインが点灯した段階で使用する
- 災害などへの対策として、同一のコピーを 2 つ作成、別々の場所で保管する





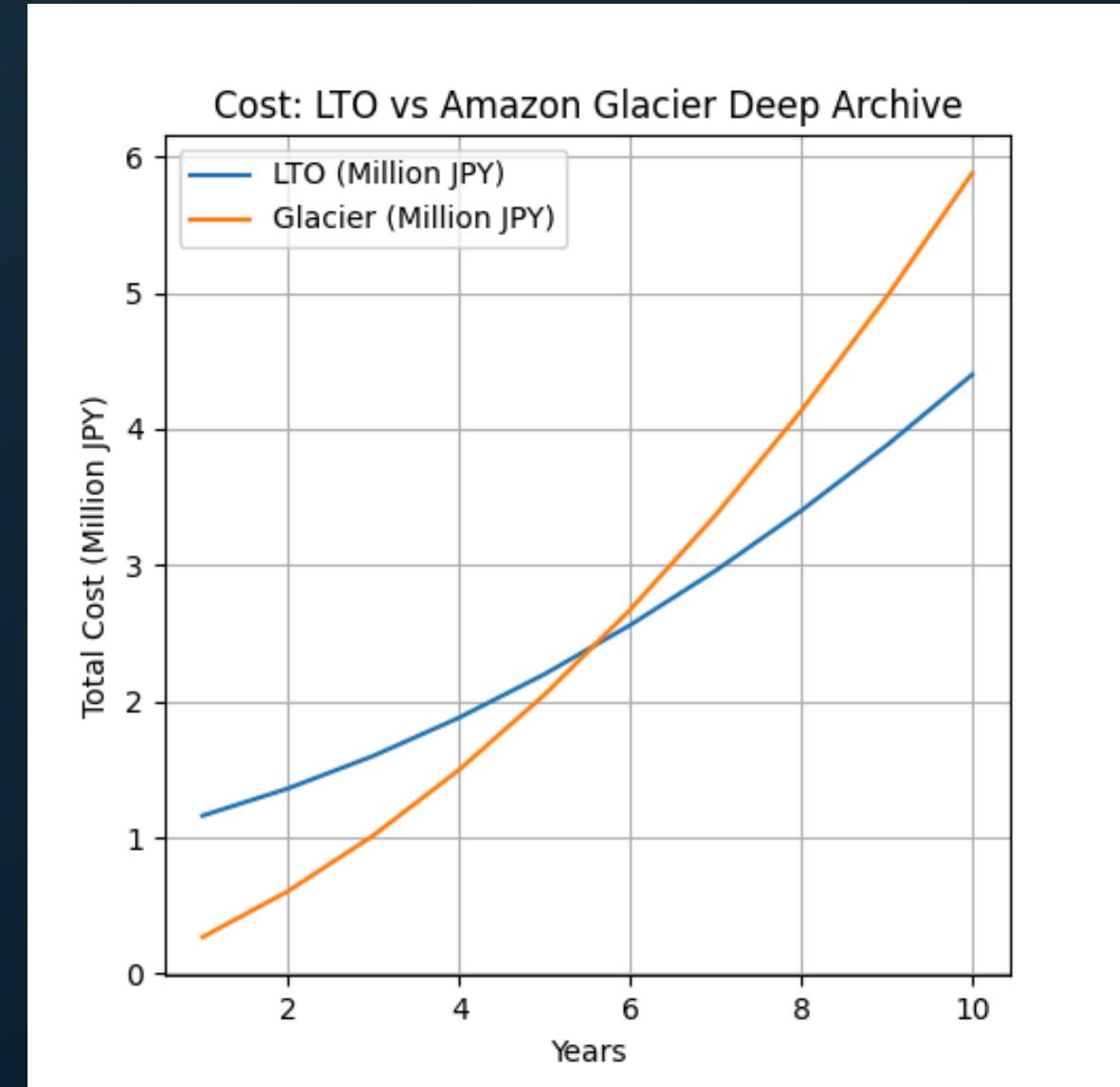
LTO 検索システム

- YoYotta を使用してアーカイブ・リトリーブ操作を行う
- YoYotta は、データ検索が若干遅い
- データコピーのログを SQL に登録して、ブラウズできるサービスを制作
- Next.js たのしい

LTO のコストパフォーマンス

- LTO は、ドライブが非常に高価だが、テープは安価
- 一度導入してしまえば、以降テープの購入費用だけで済み、長期的に高いコストパフォーマンス
- Amazon の Glacier などのクラウドストレージでは、保存しているデータ量のみならず、データのアップロード・ダウンロードにコストがかかる
- 計算 by ChatGPT (python)

150TB + 毎年 40TB 増加 + 10 年間保存



今からできる! バックアップストラテジー

- 重要なデータはクラウドストレージで同期を取る
 - ローカルのドライブの障害に備える
- HDD は、2つ以上のコピーを作る
 - 可能であれば、1つはオフサイトに保管する(自宅 + 職場など)

バックアップ体制のまとめ

進行中	終了直後	終了後
メインサーバー	メインサーバー	LTO-8 テープ
オフサイトサーバー	LTO-8 テープ	-
LTO-8 テープ(長期プロジェクト)	-	-

今後の展望 (最近興味があること)

アプリケーション開発 (Python, Node.js)

- 開発中のアプリケーション (**Seequer**)
- 連番ファイルの一覧表示、プレビュー、破損ファイル検出など

カットごとにリアルタイムで進捗状況をトラッキングすること

- フライトストリップでの管理?
- 専用のアプリケーション開発?

git でのバージョン管理

- 複数人で作業するとき、データをバージョン管理できなか (特に AEP など) ?
- 複数の AEP、複数のコンポジションなどを複数人で作業した後にマージする方法はないか?



興味がある方、是非ご連絡ください

アイデア交換や共同開発など...勉強させてください

ありがとうございました

Thank you very much for listening!

ご質問などがあればお気軽にどうぞ

本講演の資料は **Github** で公開しています



GITHUB

TWITTER



INSTAGRAM



MARSHMALLOW



Q&A

質問です。クラウドに関して、Google drive からオンプレではなく Flame.io に移行することは検討されなかったのでしょうか？また、それはなぜですか？

📌 データが映像関連のみでないことと、今後 200TB 以上にデータが膨れ上がることを想定すると、frame.io のようなクラウドサービスを使い続けることは現実的ではないと判断しました

NAS のバーチャルマシンとして使う具体例お聞きしたいです！メインマシンの方がパワーがあるのでは？と疑問でした。ファイル管理などで Windows では不向きな面があるのでしょうか？

📌 負荷が高い作業というよりは、**負荷が低いが時間がかかる作業**を回すことが多いです。それらをメインマシンで行うと、再起動などによっていちいち中断されてしまうからです

e.g. 遅いサーバーからのデータのダウンロードや、時間がかかるファイルの変換処理など

データのバックアップに関して、過去の作品等データを、うわつよ
ingなテープ媒体(正式名称忘れた；；)にまで徹底したバックアップ
・保管をなされていますが、そこまで徹底なされている理由は何な
のでしょうか。例えばこれが学術規模の研究や史料的価値のあるデー
タやアーカイブ(i.e. 国益等に係る機密情報や準ずる研究データ)なら
まだしも、正直、「そこまでやる必要ある？？」というのが率直な感
想です。

📌 ご指摘ごもっともだと思います！1つは、過去のデータが必要になるケースというのは結構存在する
ということです。例えば、類似の案件があり過去のプロジェクトデータを参照したいとか、改訂版を作
る必要があるとかです。そういうときに、消してしまってからでは遅いので、原則としてすべてのデ
ータを保存しておくということです。あとは単純に高度なストレージ管理をするのが楽しいからです！

昔、AviUtl ユーザーだった積雲仕事氏ですが、根本的な機能不足・機能差や、商業業界で AUL が非標準であることといったソフトウェアとしての根源的な差を除いて、AviUtl と(素人目にも凄い)貴環境を比較して、何か思うところ等ありますか？例えば、「無償という点以外で、もう完全に AUL を使う利点がない」でも良いですし、「こういう作業は AUL の方が良かったりする」等なんでもよいです。是非、意見をお聞きしたく存じます。

📌 After Effects といった業界スタンダードのほかに、フリー・国産で高度なモーショングラフィック制作ソフトウェアがあるという意義はめちゃくちゃ大きいと思います！（例えば学生や初心者でも手をつけやすい、モーショングラフィックの基本的な概念は十分に学べる、という点。自分もそうでした。）