



先駆的科学計算フォーラム2022

# 機械学習を用いた譜面制作支援の研究

# ゲーム産業とHPC

**KLab株式会社**  
エンジニアリング本部  
MLエンジニア  
**濱田 直希**



## 何してきたの？

2010/4 ~ 2013/3

東工大(博士)で進化計算/HPCの研究 

2013/4 ~ 2020/6

富士通研究所で機械学習の研究開発

2020/7 ~ 現在

KLabで機械学習によるゲーム開発支援

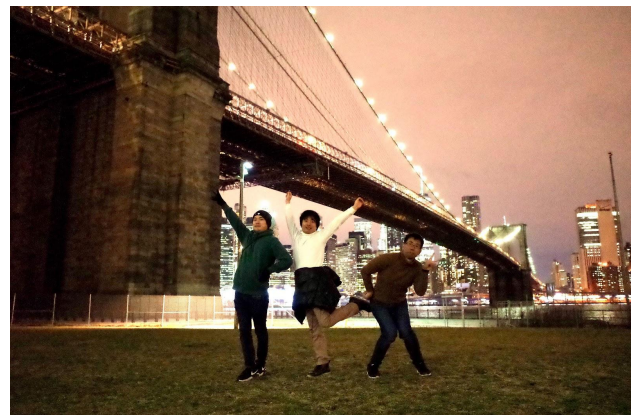
2017/4 ~ 2022/3

理研AIP(客員)で機械学習の研究

## いま何してる？

仕事 機械学習でゲーム制作を支援、大学と共同研究

趣味 ロードバイク、聖地巡礼 





社名	KLab株式会社(クラブかぶしきかいしゃ) KLab Inc.
設立	2000年8月1日
資本金	49億5747万円(2022年1月末現在)
株式公開市場	東京証券取引所・市場第一部(3656)
代表者	代表取締役社長CEO 森田 英克 代表取締役副会長 五十嵐洋介
所在地	本社(東京都港区、六本木ヒルズ森タワー) 大阪事業所、福岡事業所、仙台事業所
主要関連会社	株式会社スパイスマート 株式会社グローバルギア
海外拠点	可来软件开发(上海)有限公司 (KLab China Inc.)
グループ従業員数	正社員 556名(2021年12月末現在)

## App Store 国・地域別最高セールスランキング (抜粋)



Top 50:  
82カ国

香港	1位	シンガポール	1位
マカオ	1位	ミャンマー	1位
ベトナム	1位	スペイン	1位
フランス	1位	スイス	1位
オマーン	1位	バーレーン	1位
レバノン	1位	クウェート	1位
UAE	1位	エジプト	1位
チュニジア	1位	アルゼンチン	1位
ペルー	1位	エクアドル	1位



Top 50:  
74カ国

バーレーン	1位
カタール	1位
ラトビア	1位
イスラエル	3位
クウェート	5位
ギリシャ	5位
フランス	6位
ベルギー	6位
オランダ	8位
UAE	8位
カナダ	9位
ノルウェー	9位
オーストリア	10位
アメリカ	19位
イギリス	24位



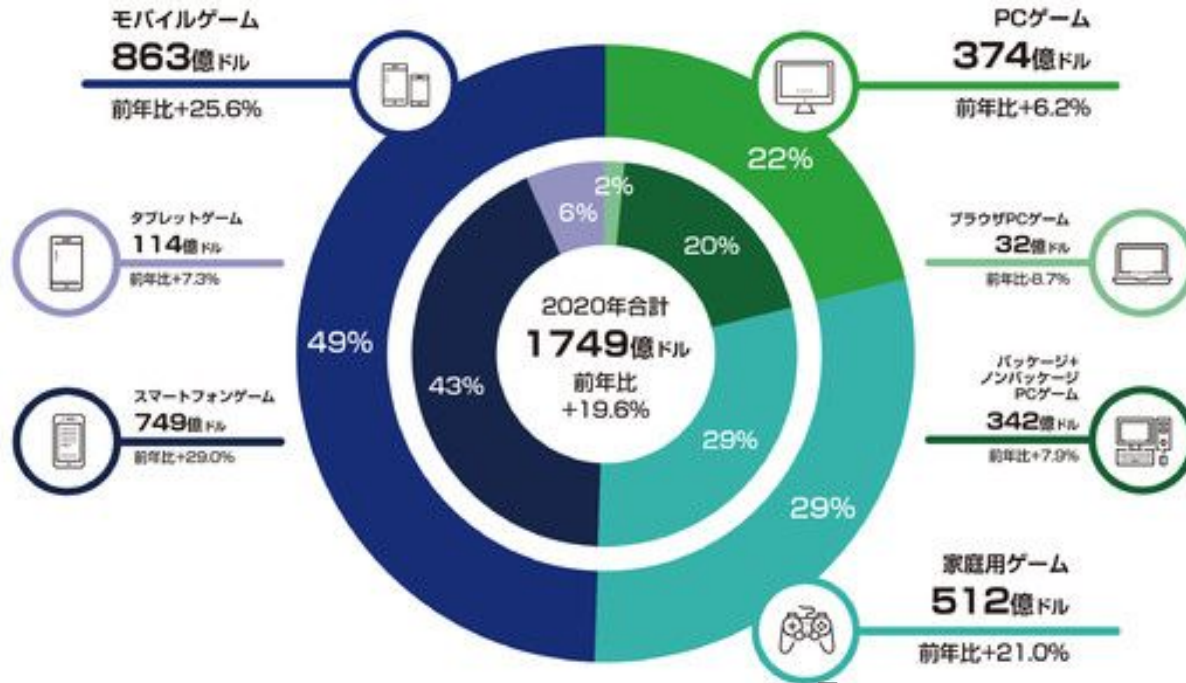
Top 50:  
68カ国

タイ	1位	アイルランド	7位
カンボジア	1位	ハンガリー	8位
インドネシア	3位	香港	9位
マレーシア	3位	イタリア	10位
フィンランド	3位	ニュージーランド	11位
マカオ	4位	フィリピン	11位
ブルネイ	5位	ベトナム	11位
韓国	4位	シンガポール	13位
台湾	6位	イギリス	23位



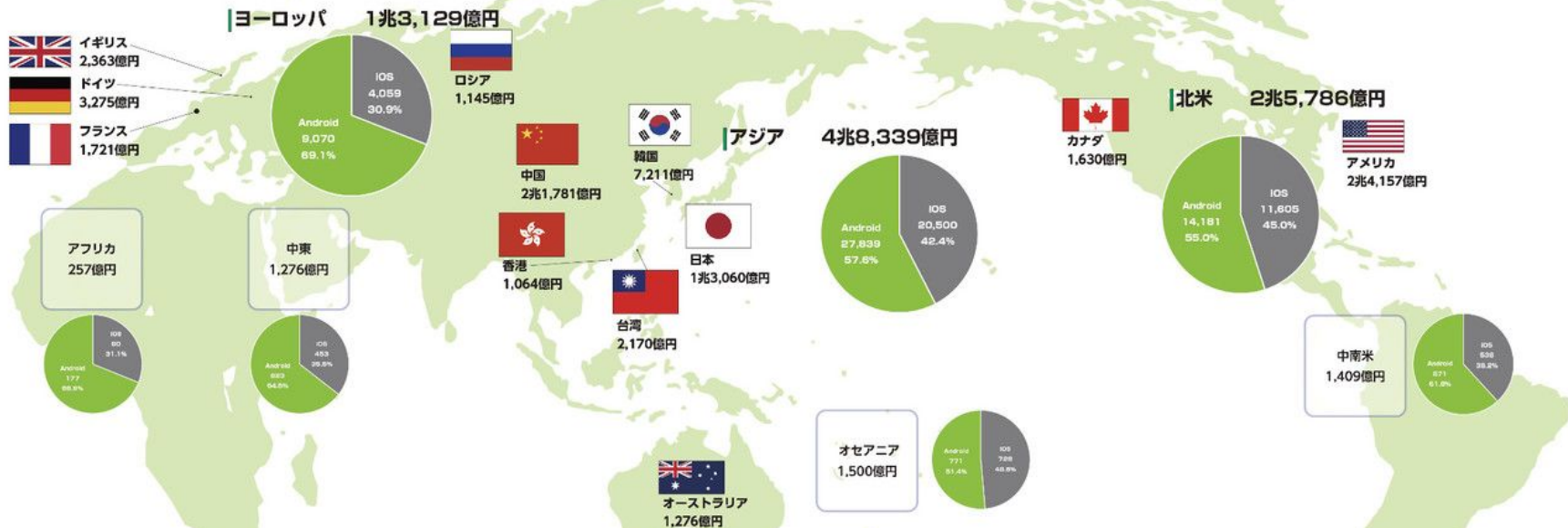


## 2020年 世界ゲーム市場規模 セグメント・デバイス別内訳



## 世界のモバイルゲームコンテンツ市場マップ

2021年の世界の  
モバイルゲーム市場規模 **9兆1,697億円**



モバイルMMORPGの  
ゲーム内ミニゲームの  
AIにAlphaZeroを使用

Revelation Mobile: Yuxi



Revelation Mobile



Revelation Mobile is a MMORPG developed by Netease games. It is a very successful game in China since it was launched on January 8th, 2021.

AlphaZero

AlphaZero and MCTS

Yuxi: a turn-based card  
game in Revelation Mobile

Applying AlphaZero

Result and future



## レーシング・エチケット

GT Sophyがスポーツパーソンシップを身につけるためには、レースの微妙なルールやフェアプレーの精神がGT Sophyの「価値関数」の中に学習されなくてはなりません。そのため、明文化されたルールと暗黙の共通認識であるマナーの両方をGT Sophyの「報酬関数」の中に表現する方法を開発しました。また、GT Sophyが人間との対戦において、アグレッシブになりすぎたり、あるいは逆に消極的になりすぎたりしないように、学習時の対戦相手のバリエーションを調節する方法も開発する必要がありました。

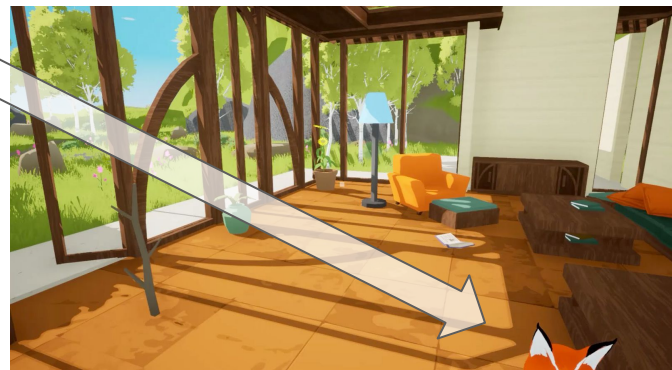
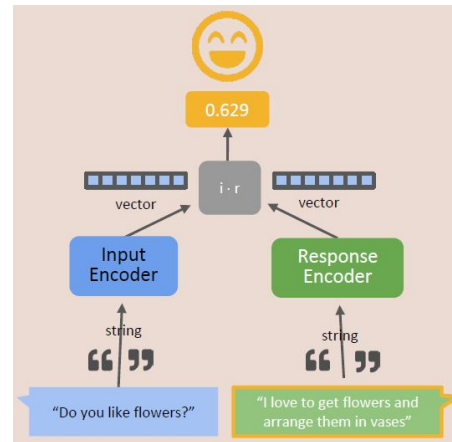
## フェアなオーバーテイク

相手をブロックせず、走行ラインを残しながらコーナーでの追い抜きを見せるGT Sophy。激しいバトルシーンながら、互いにフェアなプレイが行われている。





“I’m thirsty.” や “Something to drink?” など  
類似する意味をもつ文にも対応できる

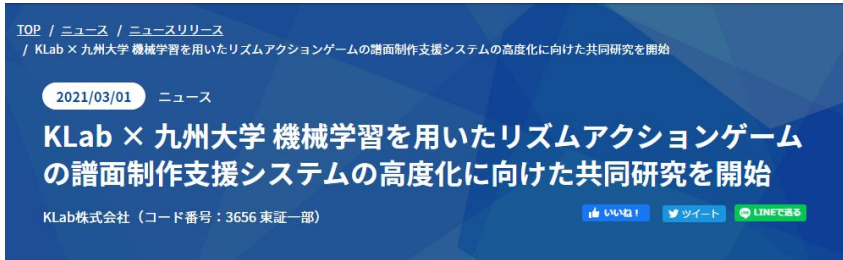


## モバイルゲームは機械学習とHPCの重要な応用先

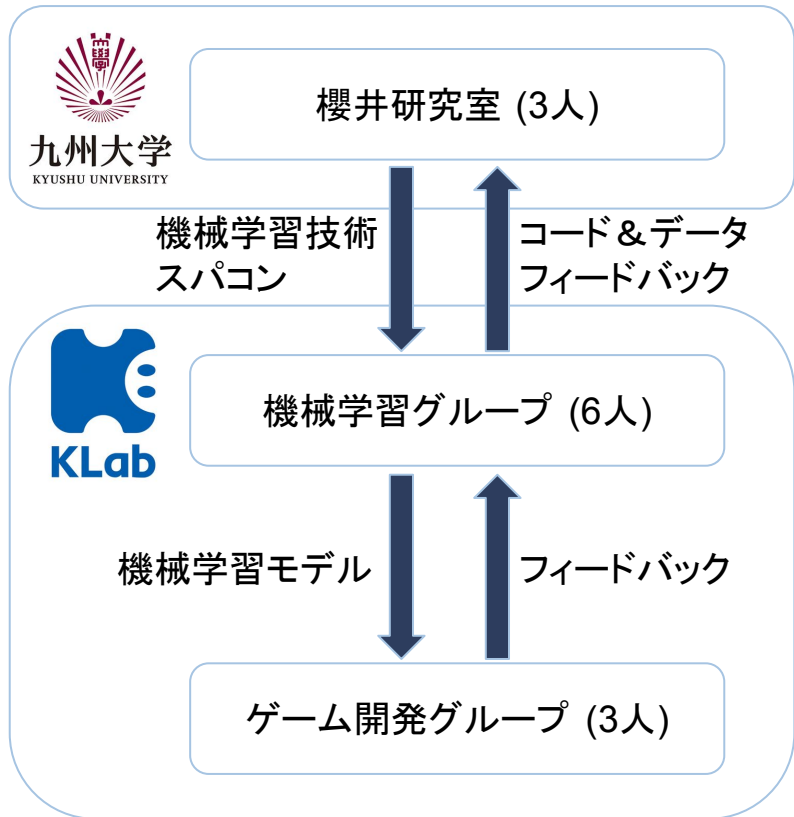
- 豊富なデータ
  - 100万人を超えるアクティブユーザー
  - 操作するたびにサーバアクセス → 数テラバイト/日のログ
- 定型タスクに当てはまらない問題が多い
  - まだまだ未開拓な応用(ゲームAI, ゲーム開発・運用・広告, ...)
  - ゲーム特有の制約(スマホで60FPS処理, 省電力, 不安定な通信)
- 機械学習の学術研究～ビジネス応用まで全てが求められる
  - しかもビジネス的にペイするコストと時間で
  - 産学共同研究を有効活用することが必須

# 共同研究について

# KLab九州大学共同研究



KLab株式会社 (本社: 東京都港区、代表取締役社長: 森田英克、以下「KLab」) と九州大学 情報基盤研究開発センター 附属汎オミクス計測・計算科学センター (所在地: 福岡県春日市、センター長: 小野謙二、以下「九州大学」) は、機械学習を用いたリズムアクションゲームの譜面制作支援システムについて共同研究を開始いたします。



## 研究成果サマリ(2021年4月～2022年3月)




- 譜面生成のSOTAモデルを改良し、国内外の会議/論文誌で発表
  - **CEDEC 2021**: 『ラブライブ！スクールアイドルフェスティバルALL STARS』の機械学習によるコンテンツ制作支援～譜面作成へのディープランニング活用～
  - **IBIS 2021**: 深層生成モデルによるリズムアクションゲームのチャート生成
  - **SIGGRAPH Asia 2021** Machine Learning Aided Content Creation In "Love Live! School Idol Festival ALL STARS" ~ Automatic rhythm game charts generation with deep learning
  - **JOTA**: Free Disposal Hull Condition to Verify When Efficiency Coincides with Weak Efficiency
  - **arXiv**: All unconstrained strongly convex problems are weakly simplicial
  - **arXiv**: GenéLive! Generating Rhythm Actions in Love Live!
  - ほかメディア記事・大学での講演等
- 開発したモデルを当時運用中のゲームに導入
  - これまでに**110曲 x 4難易度**の譜面制作に使用し、譜面制作時間を合計**約2,000時間**削減
  - 他タイトルにも横展開中

# ラブライブ！スクールアイドルフェスティバルALL STARS

各楽曲(90秒程度)につき最大4難易度

- 初級
- 中級
- 上級
- 上級+

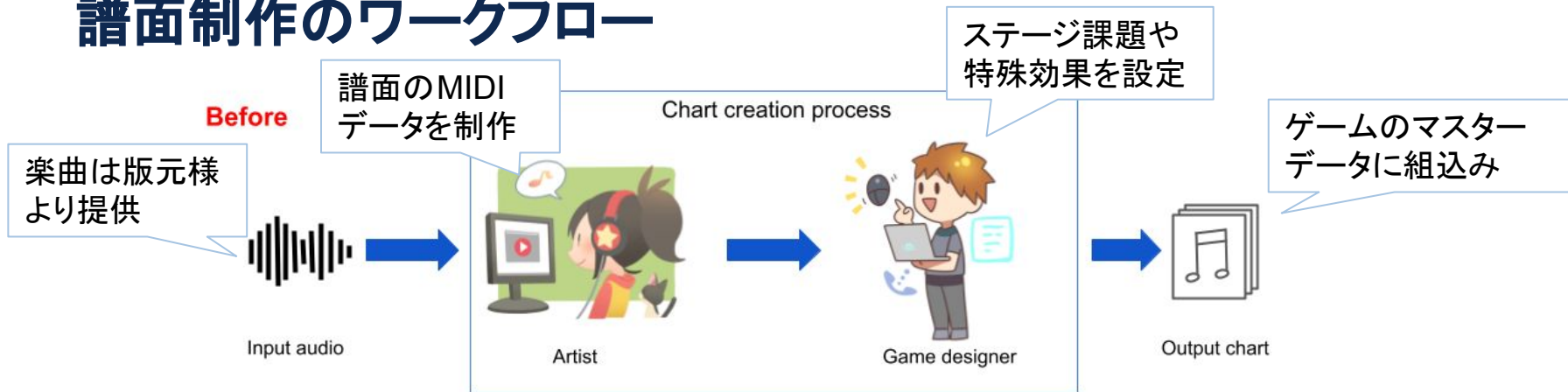
ノーツ種類

- 通常タップ 
- 上下左右スワイプ 
- ロングノーツ 
- その他ギミック

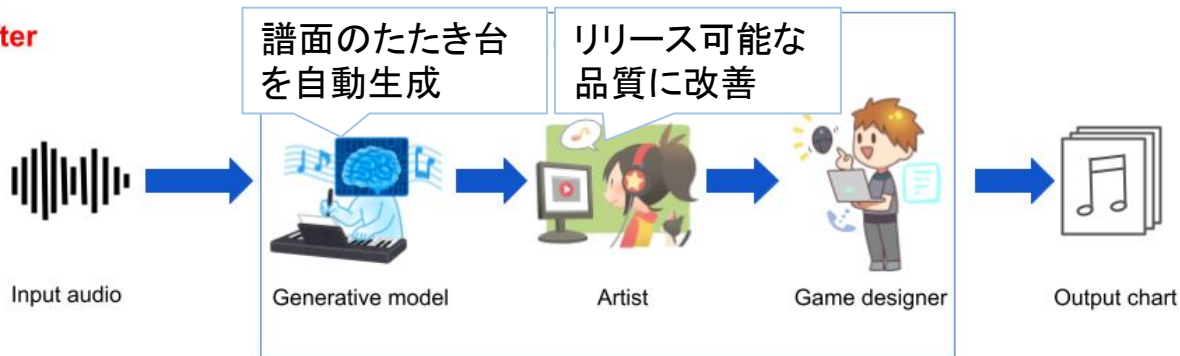


動画 <https://www.klab.com/ip/blog/tech/2021/cedec2021.html>

# 譜面制作のワークフロー

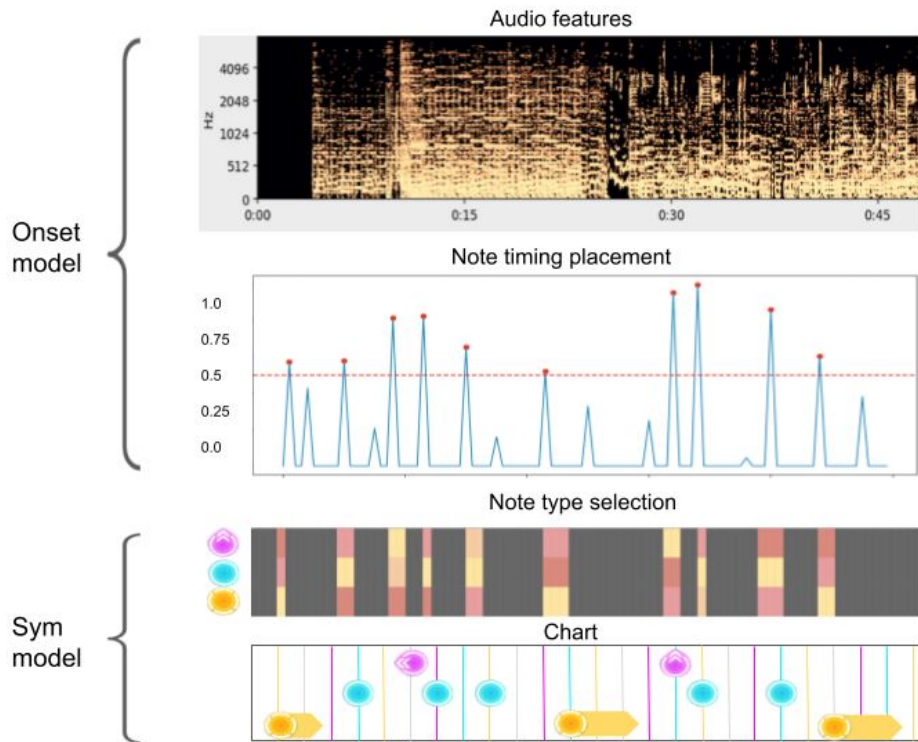


**After**





# 深層生成モデルによる譜面生成



入力:メルスペクトログラム



ノーツの存在確率



ノーツの種類



出力:譜面

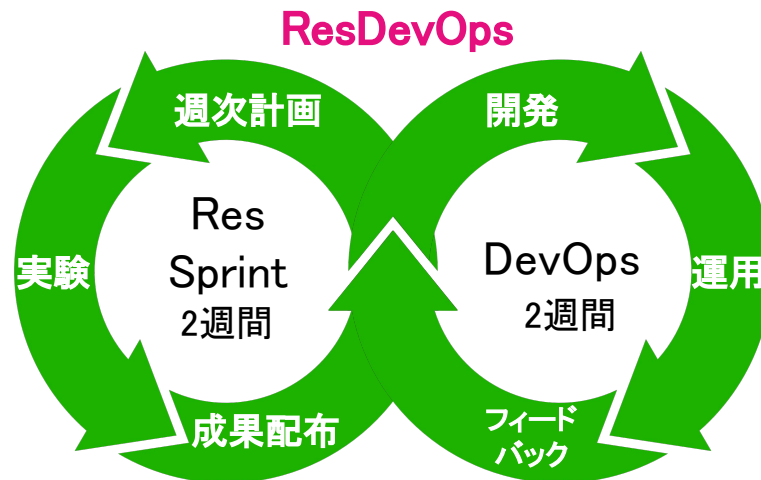
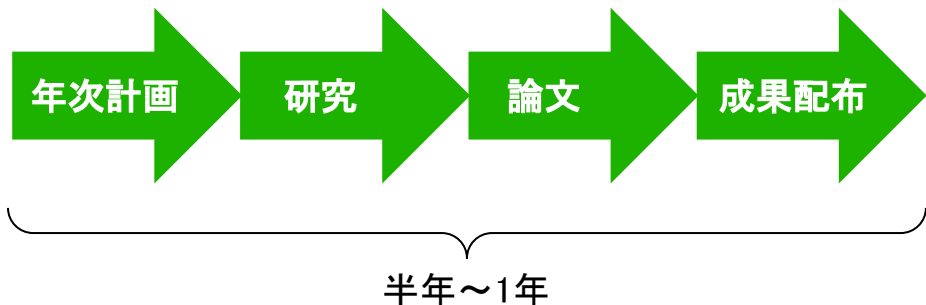
## KLabのリズムゲームにおける譜面制作の課題

- 毎月 3タイトル x 5曲 x 4難易度 を配信(当時)
- 譜面制作に1曲あたり40時間 ➡ 計600時間/月
- 既存モデルでは低難易度の譜面に課題 ➡ 要研究
- ML研究(半年)とゲーム運用(2週間)に13倍の時間スケールギャップ ➡ ResDevOpsが必要

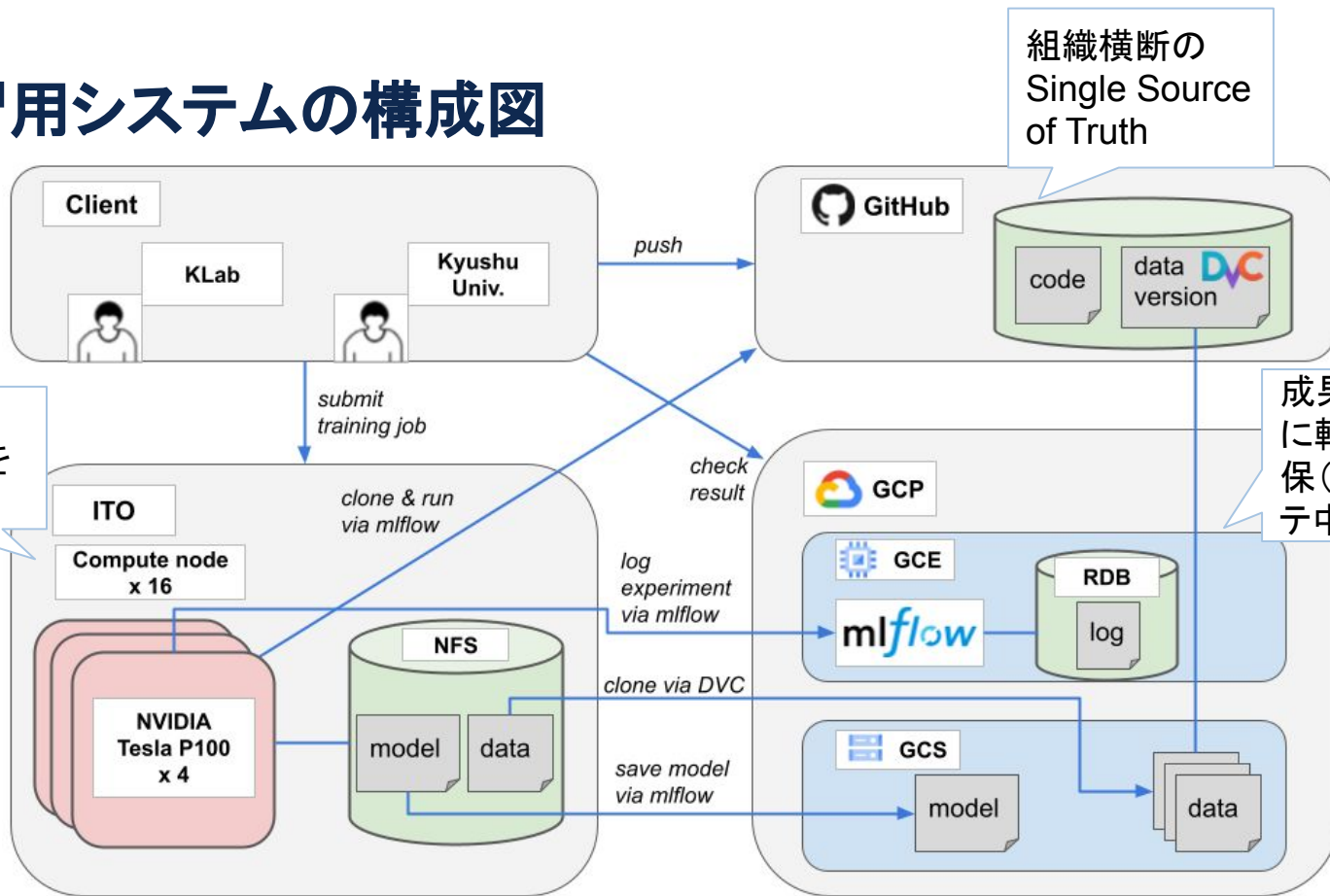
## ResDevOps = Research Sprint + DevOps (MLOps)

- 研究～開発～運用の時間スケールギャップを解消するプラクティス
  - 現場ニーズに直結する研究成果をタイムリーに提供
  - 現場からの継続的フィードバックで研究を加速
- 研究サイクルをスプリント化し, CI/CDと接続

### 通常の研究



# 学習用システムの構成図



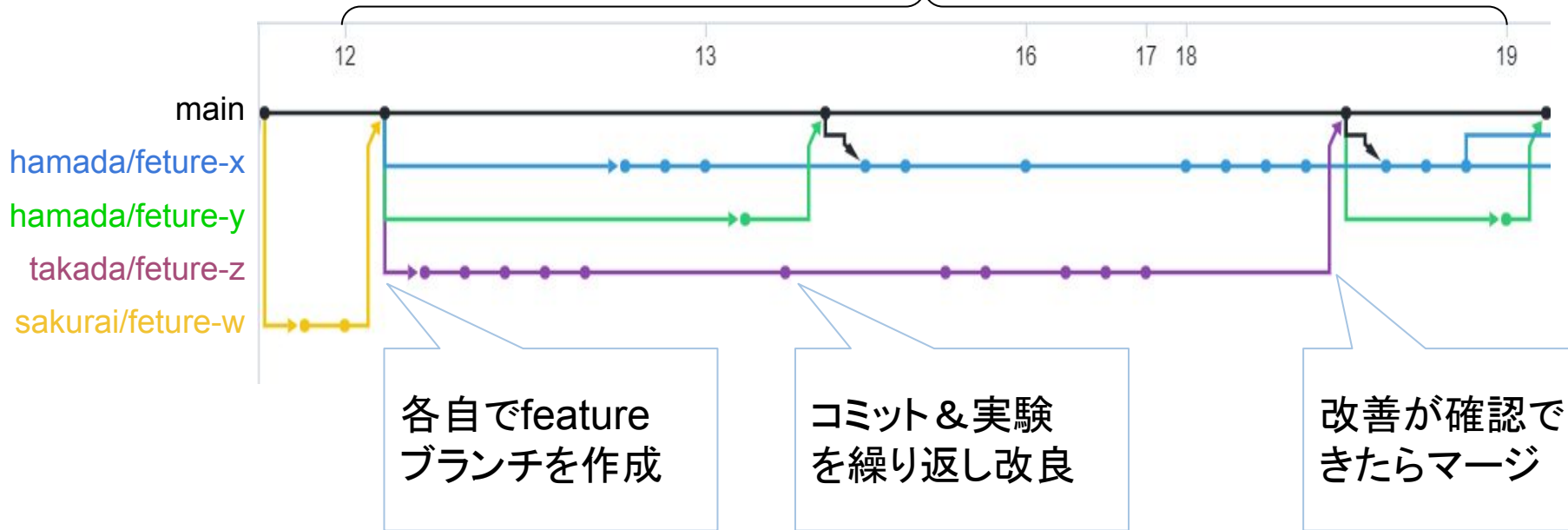
スパコンで  
大量の実験を  
安価に実行

成果物をクラウド  
に転送し可用性確  
保(スパコンのメン  
テ中も利用可)

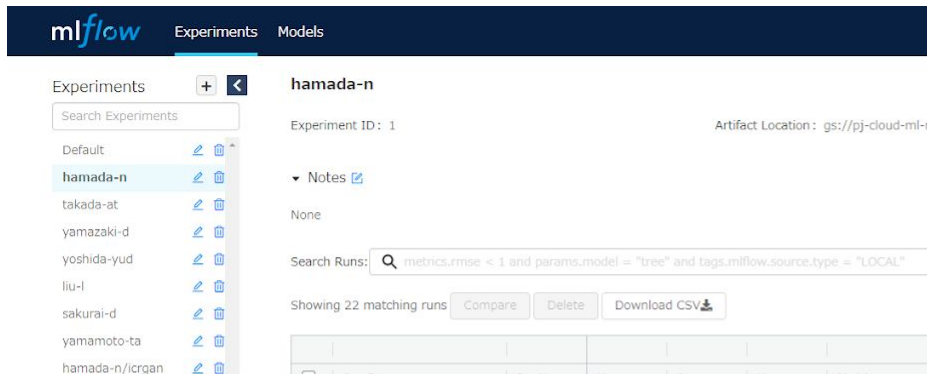
# GitHub Flowにより各自が並列に研究アイデアを検証

2021年4月

1週間



# MLflowで実験を共有し議論



MLflow Experiments Models

Experiments **+** **-**

Search Experiments

- Default
- hamada-n**
- takada-at
- yamazaki-d
- yoshida-yud
- liu-l
- sakurai-d
- yamamoto-ta
- hamada-n/icrgan

**hamada-n**

Experiment ID: 1

Artifact Location: gs://pj-cloud-ml-

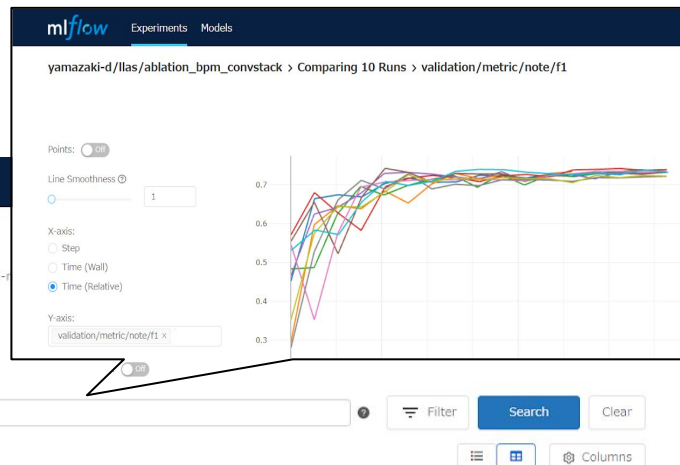
Notes

None

Search Runs:

Showing 22 matching runs Compare Delete Download CSV

お互いの実験を  
リアルタイムに  
閲覧できる

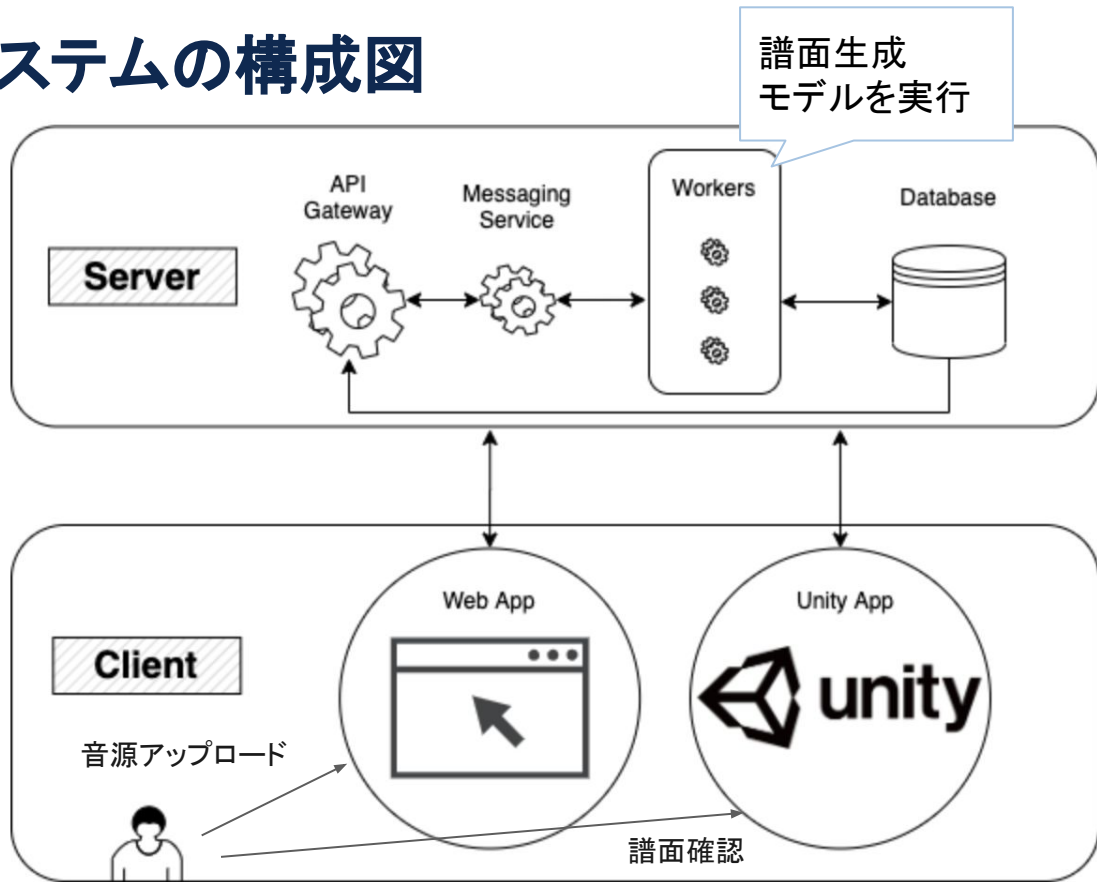


	Start Time	Run Name	User	Source	Version	Models	Parameters >			Metrics >			Tags
							aug_config	aug_count	batch	loss	loss/bCR	loss/discrimina	host
<input type="checkbox"/>	2021-05-13 21:35:27	14057626	hamada-n	notes_gener	ad059d	pytorch	./aug_config...	5	16	4.054	0.026	0.097	scb0049.ky...
<input type="checkbox"/>	2021-05-13 20:56:49	14052931	hamada-n	notes_gener	93866c	pytorch	./aug_config...	5	16	4.926	0.035	0.112	scb0048.ky...
<input type="checkbox"/>	2021-05-13 19:12:16	-	hamada-n	notes_gener	edd894	-	./aug_config...	5	16	-	-	-	-
<input type="checkbox"/>	2021-05-12 10:55:32	-	hamada-n	notes_gener	83fd95	-	aug_config...	5	16	-	-	-	-
<input type="checkbox"/>	2021-05-10 14:17:14	13749068	hamada-n	notes_gener	36b4e8	pytorch	aug_config...	5	16	4.909	0.029	0.107	scb0040.ky...
<input type="checkbox"/>	2021-05-06 21:39:52	13726473	hamada-n	notes_gener	36b4e8	pytorch	aug_config...	5	16	3.977	0.012	0.085	scb0028.ky...
<input type="checkbox"/>	2021-05-06 12:19:59	13722824	hamada-n	notes_gener	f3ff60	-	aug_config...	5	16	4.213	0.01	0.055	scb0041.ky...
<input type="checkbox"/>	2021-05-06 03:45:52	13719685	hamada-n	notes_gener	f3ff60	-	aug_config...	5	16	4.113	0.016	0.074	scb0034.ky...
<input type="checkbox"/>	2021-05-06 03:41:40	13719669	hamada-n	notes_gener	f3ff60	-	aug_config...	5	16	4.233	0.015	0.082	scb0033.ky...
<input type="checkbox"/>	2021-05-06 03:36:40	13719641	hamada-n	notes_gener	f3ff60	-	aug_config...	5	16	5.186	0.184	0.32	scb0032.ky...
<input type="checkbox"/>	2021-05-06 03:34:52	13719625	hamada-n	notes_gener	f3ff60	-	aug_config...	5	16	5.046	0.024	0.083	scb0028.ky...
<input type="checkbox"/>	2021-05-06 00:55:14	13718635	hamada-n	notes_gener	83fd95	-	aug_config...	5	16	7.213	0.039	1.319	scb0034.ky...
<input type="checkbox"/>	2021-05-04 13:37:00	13701365	hamada-n	notes_gener	729b34	pytorch	aug_config...	5	16	4.571	0.014	0.066	scb0038.ky...
<input type="checkbox"/>	2021-05-04 13:37:00	13701366	hamada-n	notes_gener	729b34	pytorch	aug_config...	5	16	4.44	0.027	0.084	scb0039.ky...

## 爆速の研究開発を実現

- 2021年4月～2021年12月に
  - 試したアイデア 234件 (MLflow Experiments数)
  - 投入したジョブ 5,807件 (pjstat -H day=275)
  - のべ計算時間 20,625ノード時間 (qrepuse)  
≒ 82,500GPU時間

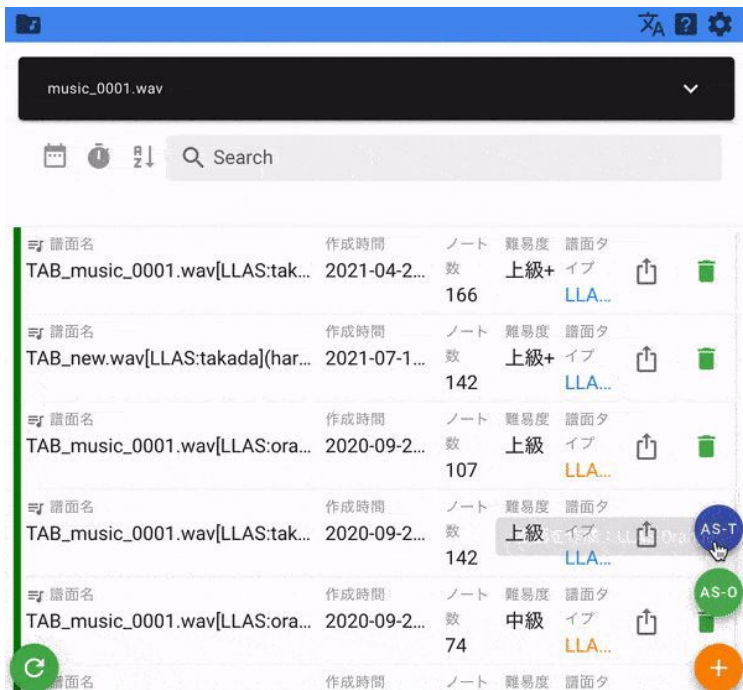
# 推論用システムの構成図



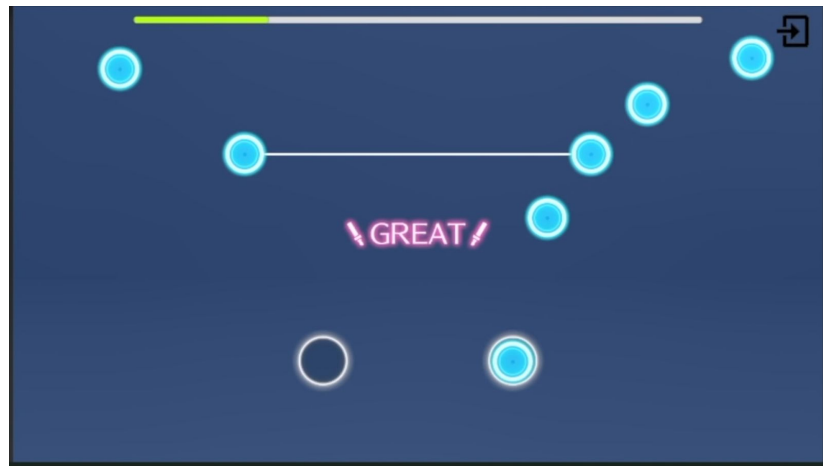
Google  
Kubernetes Engine



# サウンドチーム自身で譜面生成&確認を可能に



譜面名	作成時間	ノート数	難易度	譜面タイプ	操作
TAB_music_0001.wav[LLAS:tak...	2021-04-2...	166	上級+	イブ LLA...	📄 🗑️
TAB_new.wav[LLAS:takada](har...	2021-07-1...	142	上級+	イブ LLA...	📄 🗑️
TAB_music_0001.wav[LLAS:ora...	2020-09-2...	107	上級	イブ LLA...	📄 🗑️
TAB_music_0001.wav[LLAS:tak...	2020-09-2...	142	上級	イブ LLA...	📄 🗑️ AS-T
TAB_music_0001.wav[LLAS:ora...	2020-09-2...	74	中級	イブ LLA...	📄 🗑️ AS-0
...	...	...	...	...	+



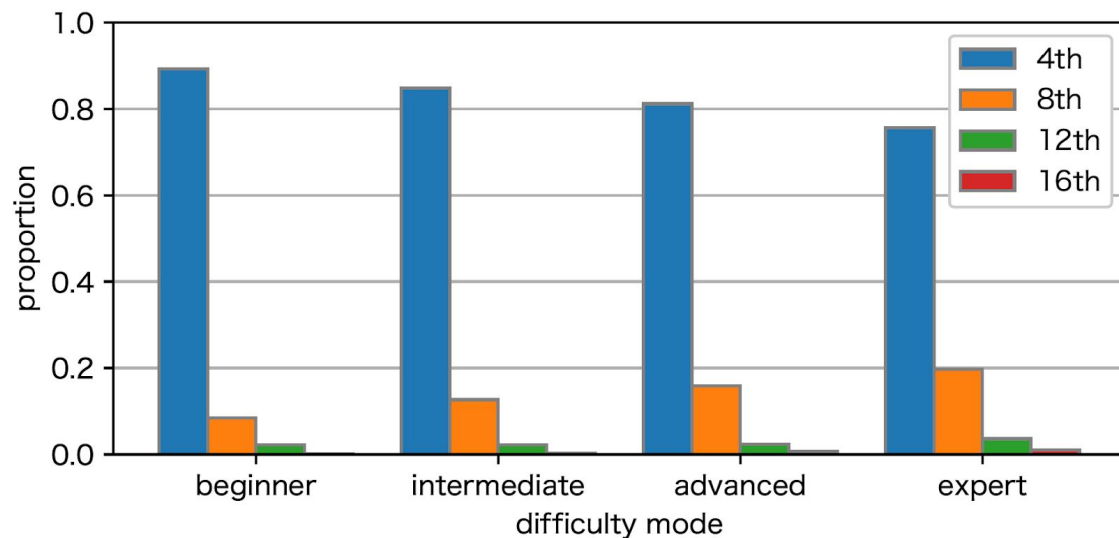
## 譜面制作の工数を50%削減

- 2021年4月～2021年12月に
  - 生成した譜面 110曲 x 4難易度  
≒ 440譜面
  - 削減したコスト 1曲40時間→20時間  
≒ 2,200時間

## 学習データ

Datasets	# songs	# charts
Love Live!	163	501
Utapri	140	579
Stepmania Fraxtil	90	450
Stpmania ITG	133	652

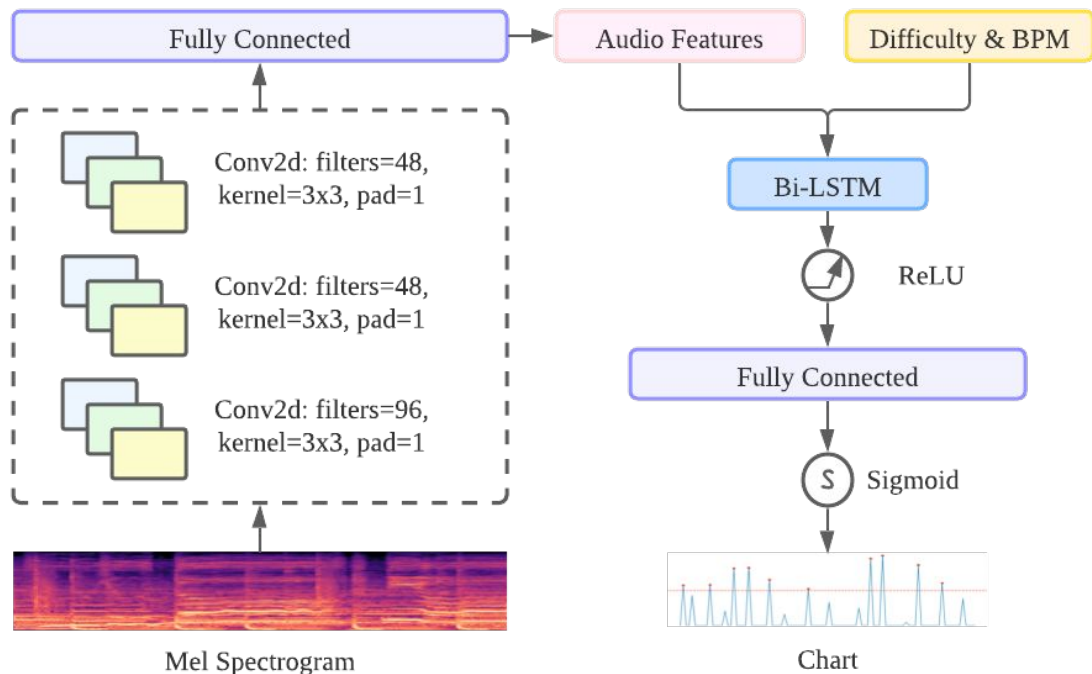
## 性質1: ノーツ位置は4分音符と8分音符がほとんど



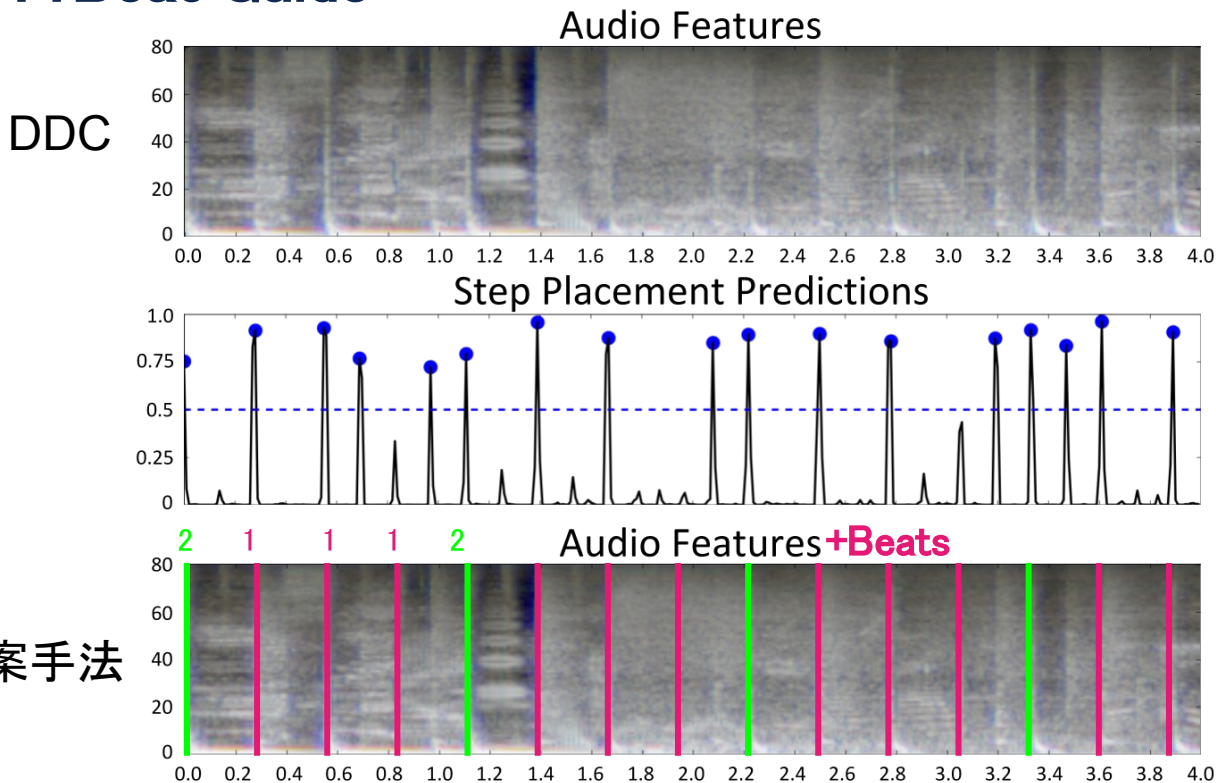
## 性質2: 低難易度譜面は高難易度譜面にほぼ含まれる

$s \backslash t$	Beginner	Intermediate	Advanced	Expert
Beginner	100%	97.4%	97.4%	98.6%
Intermediate	64.7%	100%	98.8%	98.2%
Advanced	49.8%	76.2%	100%	98.6%
Expert	39.0%	59.4%	73.5%	100%

# NNの基本構造は先行研究DDCとほぼ同じ



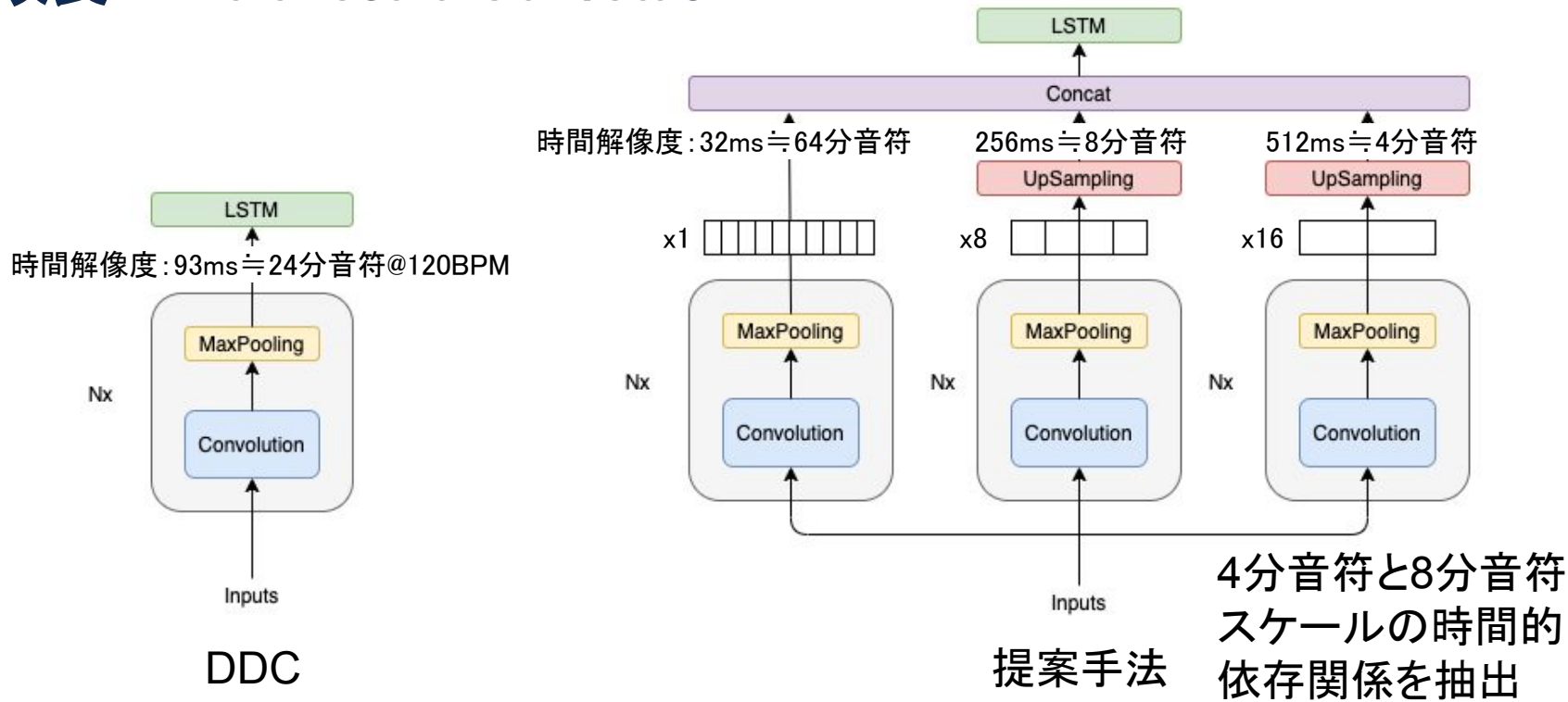
# 改良1: Beat Guide



提案手法

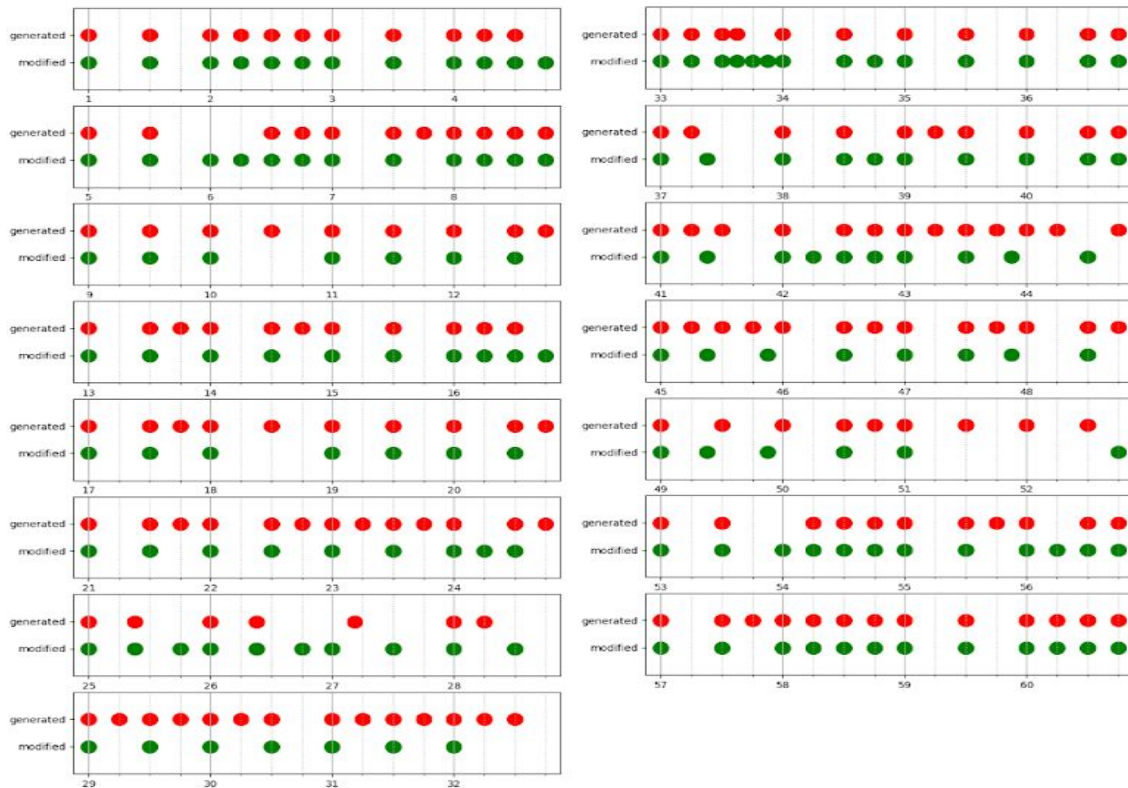
小節頭と拍の位置  
を教えることで  
ノーツ位置の  
ヒントになる

## 改良2: Multi-scale Convstack





# 生成譜面の例: わずかな手直しで製品品質に



- 生成譜面
- 製品譜面

※製品譜面は生成譜面を手直して作成

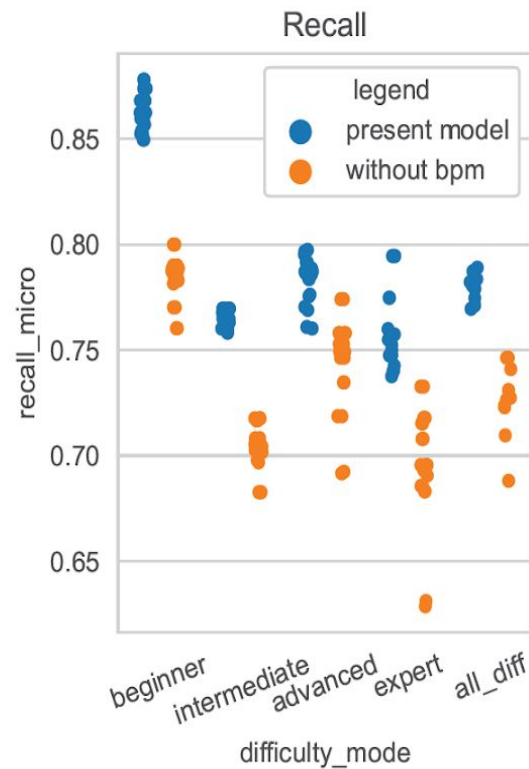
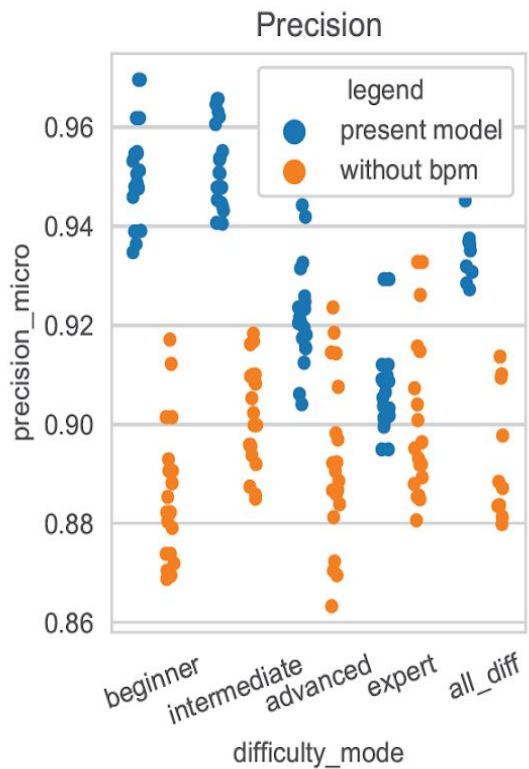
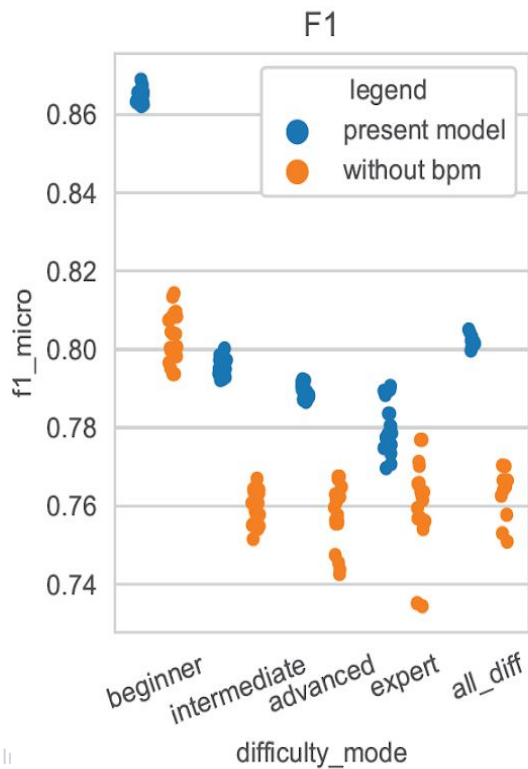
混同行列  
(8分音符で量子化)

	P	N	
P	119	28	.81
N	47	294	.86
	.72	.91	.85

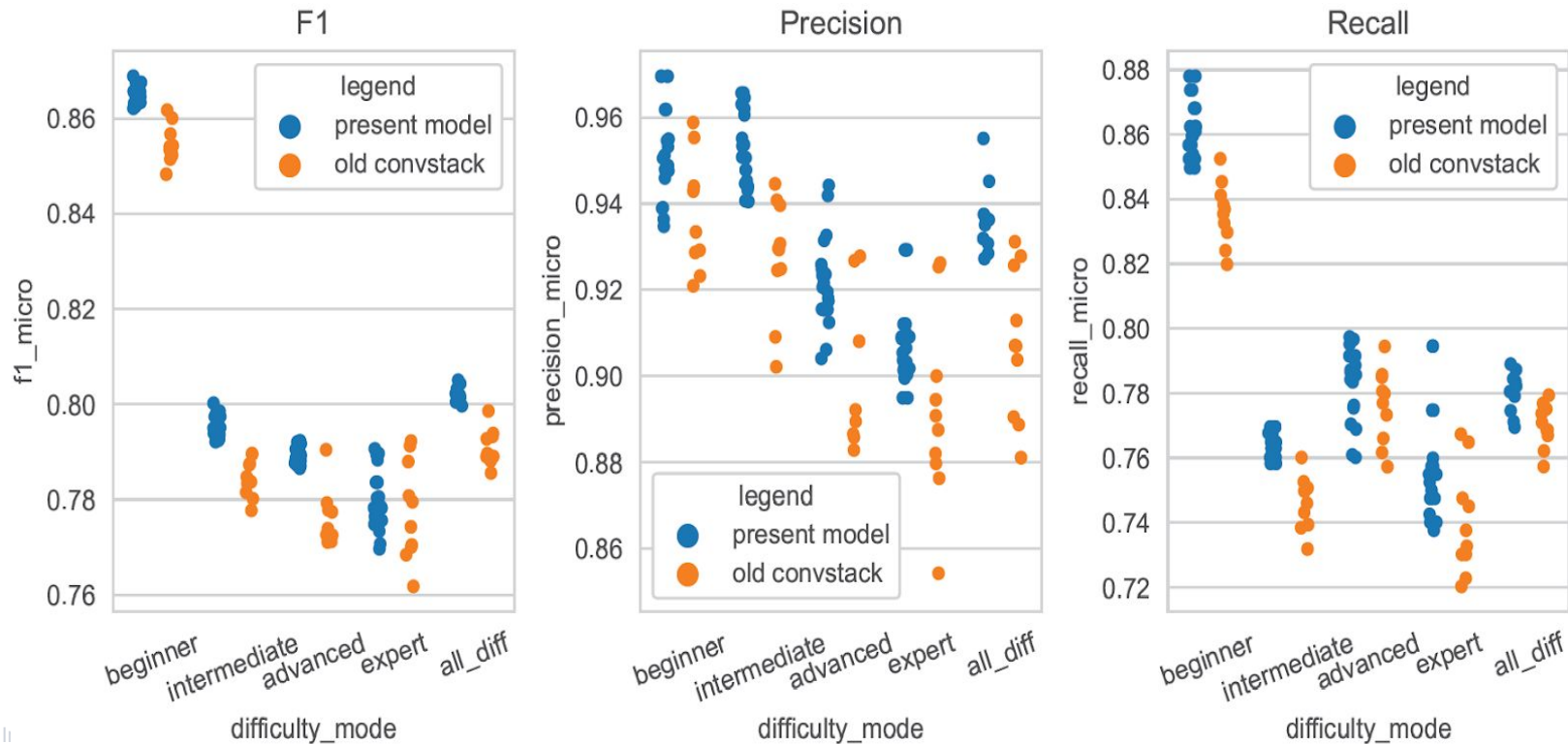
## 結果:すべての難易度でF値が改善(特にBeginner)

Difficulty	F-score <sup>m</sup> (std.)		F-score <sup>c</sup> (std.)	
	GenéLive!	DDC	GenéLive!	DDC
Beginner	0.8643 (0.0018)	0.7823 (0.0081)	0.8627 (0.0018)	0.7832 (0.0080)
Intermediate	0.7953 (0.0024)	0.7452 (0.0039)	0.7949 (0.0020)	0.7465 (0.0045)
Advanced	0.7885 (0.0017)	0.7494 (0.0044)	0.7878 (0.0021)	0.7503 (0.0042)
Expert	0.7784 (0.0062)	0.7610 (0.0059)	0.7796 (0.0060)	0.7607 (0.0058)
all	0.8021 (0.0016)	0.7524 (0.0046)	0.8104 (0.0013)	0.7559 (0.0049)

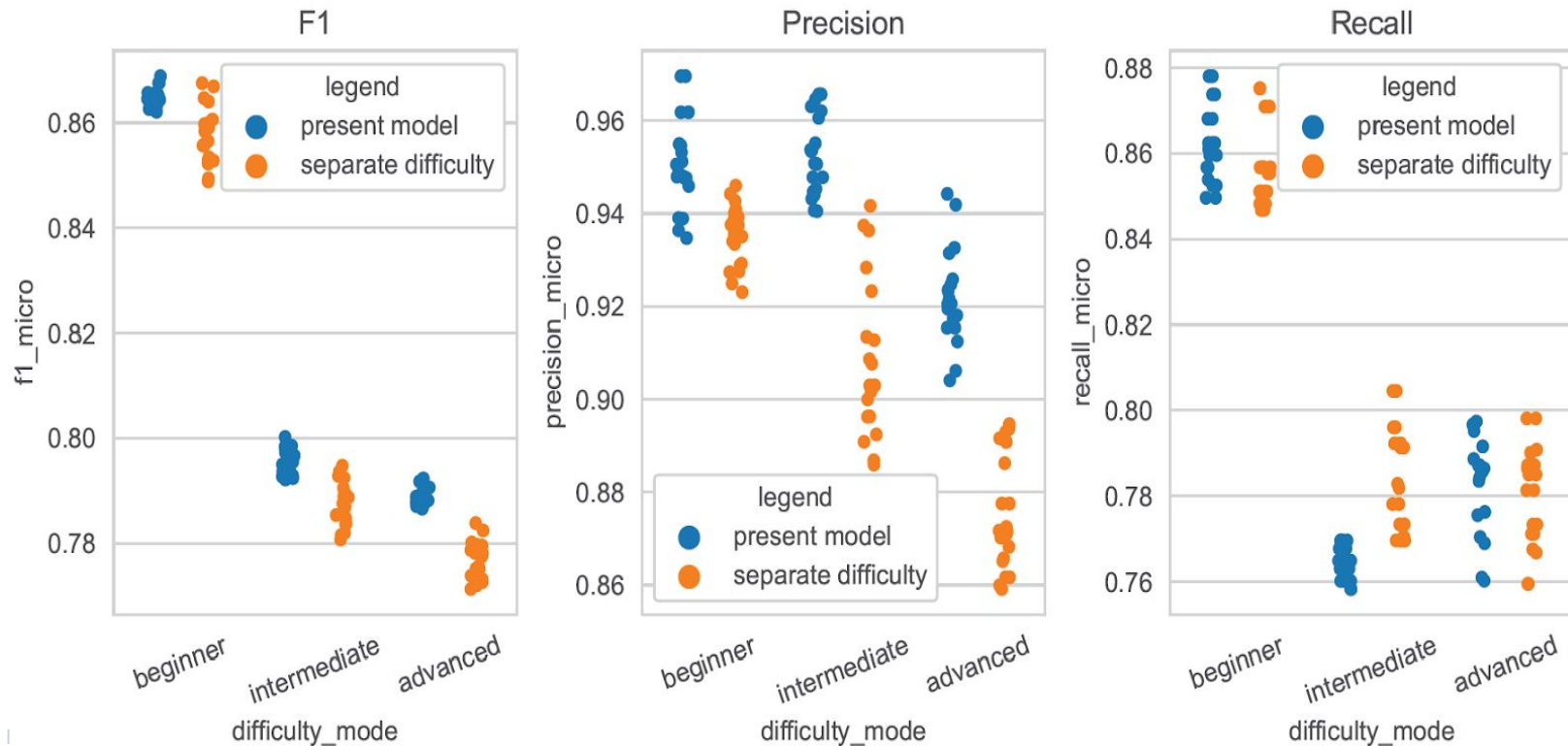
## 考察: Beat GuideでF値が改善(特にBeginner)



## 考察: Convstack改良でF値が改善 (Expert以外)



## 考察: 全難易度1モデル学習でF値が改善(分散→小)



## まとめ

- 深層生成モデルで譜面生成
- ResDevOpsで研究を加速
- 譜面の性質を考慮してSoTAモデルを改良
- 譜面制作時間を50%削減
- プレプリント: <https://arxiv.org/abs/2202.12823>

# HPCとゲームの未来

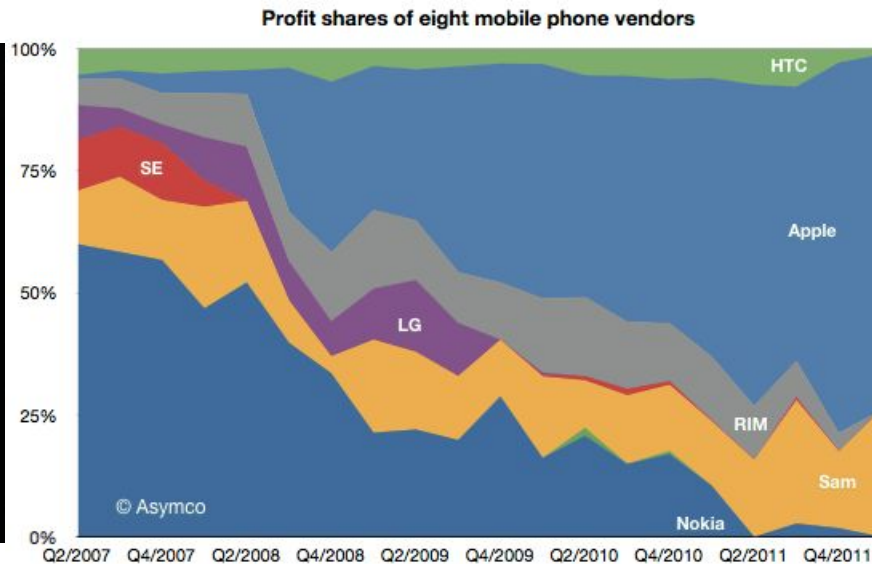
(ディスカッション用補足資料)

## HPC界でゲームチェンジが起こり始めている

- 2019年頃からAI用HPCクラスタをKubernetes (k8s)で構築する企業が登場
  - NVIDIA: [RDG: Kubernetes Cluster Deployment for ML and HPC Workloads with NVIDIA GPU Virtualization and VMware PVRDMA Technologies.](#)
  - Uber: [Creating Kubernetes Based UberCloud HPC Application Clusters Using Containers](#)
  - Open AI: [Scaling Kubernetes to 7,500 Nodes](#)
- 国内企業も続々とk8sでAI用HPCクラスタを構築しはじめた
  - Yahoo!: [ヤフーのAIプラットフォーム紹介 ～ AI開発をより手軽に](#)
  - PFN: [PFN's Supercomputers](#)
  - DeNA: [DeNAのMLopsエンジニアは何をしているのか。機械学習基盤「Hekatoncheir」を解説](#)
  - CyberAgent: [AI事業本部におけるGPU活用の取り組みとKubernetes](#)
- クラウド各社もHPCサービスを開始
  - AWS: [AWS Elastic Fabric Adapterの通信速度評価](#)
  - Azure: [Azure ハイ パフォーマンス コンピューティング](#)
  - GCP: [Introducing HPC VM images—pre-tuned for optimal performance](#)



## イノベーションのジレンマ



“We didn’t do anything wrong, but somehow, we lost”.

---Nokia CEO Stephen Elop, 2016

写真: <https://www.linkedin.com/pulse/nokia-ceo-ended-his-speech-saying-we-didnt-do-anything-rahul-gupta/>

図: <https://www.cultofmac.com/164712/iphone-4s-helps-apple-rake-in-73-of-mobile-phone-profits/>

## k8sのバッチスケジューラとしての実力は？

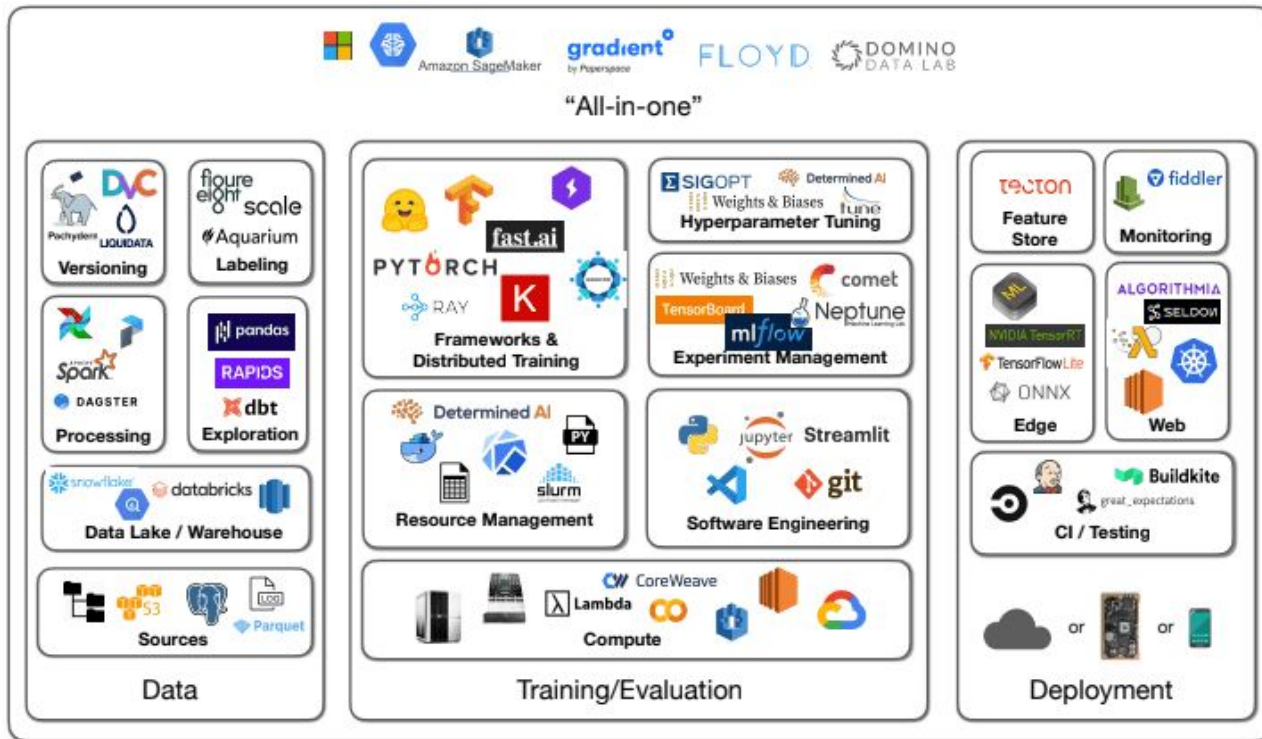
- k8sのデフォルトスケジューラはバッチジョブに不向き
  - [Topology Manager](#)はノード内のリソーストポロジしか考慮できない
  - フェアシェア等の機能がない
- [Volcano](#)を使えば既存のバッチスケジューラ相当の機能
  - Gang, Fair-share, Queue, Preemption, Topology-based scheduling
  - Reclaim, Backfill, Resource reservation
- k8sマニフェストの肥大化
  - 何層もの仮想化の上からHW性能を引き出すために複雑な設定が必要
  - ちょっとしたジョブを投げるにも大量の決まり文句を書かなければならない
- k8s上にバッチスケジューラを再発明しただけで、明確な優位性がないようにも思えるが…(コンテナ使いたいだけならスパコンにもSingularityがあるし…)



## なぜk8sが選ばれるのか？

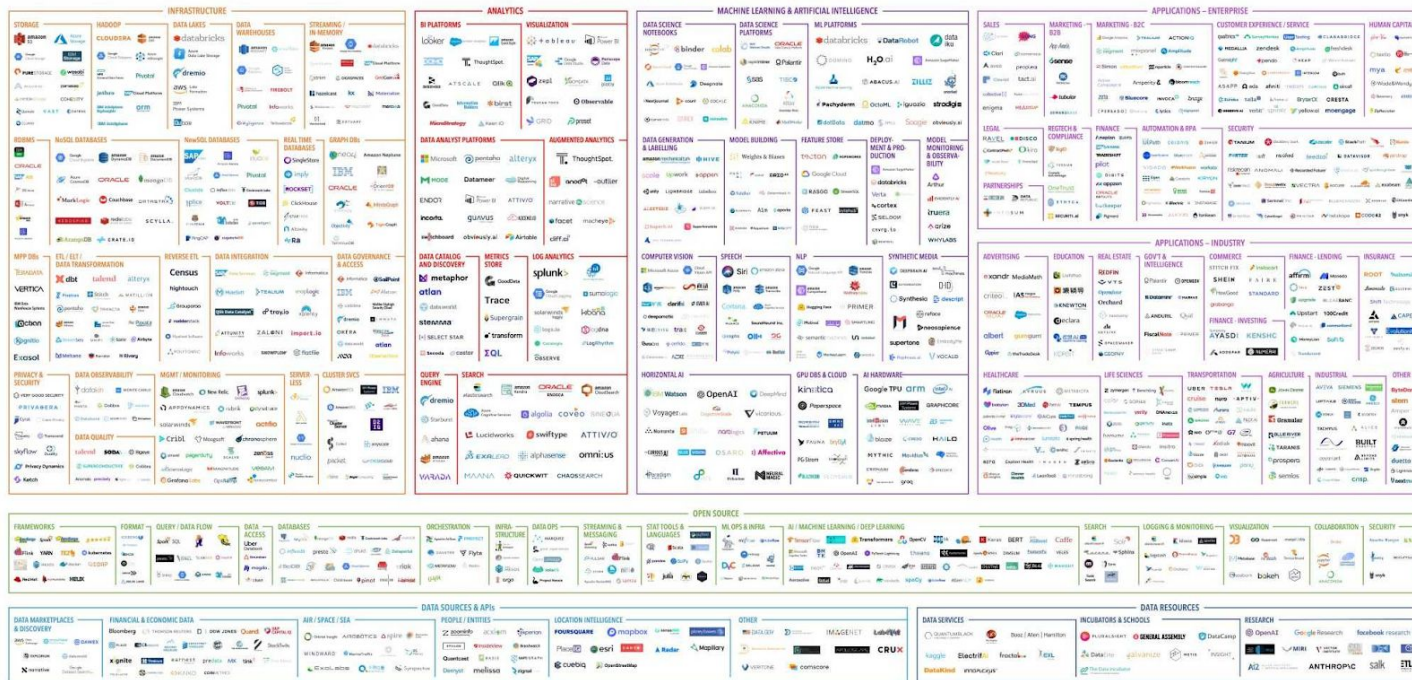
- 多くの企業はすでにk8sで各種サービスをホストしている
  - 同一のツールでバッチ処理もできればクラスタ利用率を高められる
- ハードウェアよりエンジニアの方が希少で高価
  - k8sさえ覚えれば何でもできるようにして学習コストを下げたい
- k8sのスキルはエンジニアの市場価値を高める
  - みんなk8sを覚えたがる
- 仕様が標準化・オープン化されておりポータビリティが高い
  - 同じコードがオンプレでもGCPでもAWSでもAzureでも動く
  - サードパーティ製の周辺ツールが充実
- ボイラプレートの問題はラッパーを作れば解消できる
  - Vertex AI Training@GCPはGKE上のバッチ処理サービス
  - pfkube@PFNやacloud@Yahoo!はkubectlコマンドを簡易化

# k8sエコシステムには多数のMLOpsサービスが登場



# 従来型スケジューラはエコシステムから締め出される!?

MACHINE LEARNING, ARTIFICIAL INTELLIGENCE, AND DATA (MAD) LANDSCAPE 2021



Version 3.0 - November 2021

© Matt Turck (@matturck), John Wu (@john\_d\_wu) & FirstMark (@firstmarkcap)

matturck.com/data2021

FIRSTMARK  
EARLY STAGE VENTURE CAPITAL

## この先生きのこるには:スパコン as a Service

- クラウドネイティブなサービスとの相互運用性
  - Web API(トークン認証, ロールベースアクセス制御(RBAC))
  - クラウドサービスとの連携 (Webhook, クラウド各社のCLIツール)
  - クラウドのバッチ処理サービスとの互換性
- インターフェースの標準化
  - CLIの統一 & サブコマンド化
  - スパコン版Write once, run anywhere
  - スパコン間でのジョブ & データ転送
- ユーザマニュアルのWeb公開

## Web API(トークン認証, RBAC)

- ニーズ: 多数のユーザや外部サービスからスパコンを利用したい
  - 10人超の研究チームによるGitOps, 数十のサービスと連携, ...
- SSH鍵認証の欠点
  - 認証 = 全権認可 ⇒ 漏洩時の被害大 ⇒ 多数ユーザは危険
  - 各自で環境構築するとバラバラになる ⇒ 他人の実験が再現不能に
- Web API(トークン認証, RBAC)の利点
  - グループ内で管理者と利用者に分けられる
  - ユーザ毎に最小限のロールを与えてアクセス制御
  - 短命 & 再発行が容易なトークンで認証
  - 管理者が全員ぶんの環境を構築
  - 利用者はジョブ投入・閲覧権限のみ

## クラウドサービスとの連携 (Webhook, クラウドCLI)

- ニーズ:スパコンの成果物をクラウドで利用したい
  - ログ, 訓練済モデル, 計算結果
  - 可用性が必要な部分はクラウドで保存したい
  - 年度末に環境移行で慌てたくない
- 連携プログラムを自作することの欠点
  - サービス間のI/Fがバラバラ ⇒ エコシステムが育たない
  - ベストプラクティスが分からない ⇒ 誤用でセキュリティ事故に
- WebhookやクラウドCLIを運営側で用意することの利点
  - 他人のプラクティスが再利用できる(OSS, ブログの活性化)
  - エコシステムが成長⇒スパコン維持費の削減
  - 新規利用者の増加&クイックスタート



## クラウドのバッチ処理サービスとの互換性

- ニーズ: ワークロードをスパコン・クラウド間で簡単に移行したい
  - モデル開発はスパコンで, 再学習はクラウドで
  - 閑散期はスパコンで, 卒論シーズンはクラウドで
- シームレスに使えると新規利用者も増える

## CLIの統一 & サブコマンド化

- ニーズ: CLIがわかりにくい
  - スパコン間でCLIコマンドに互換性がなく、毎回コマンドもオプションも覚え直し
  - サブコマンド化されていないので補完が効かない
  - サブコマンド化されていないので何がバッチスケジューラのコマンドかわからない
- 階層化すれば何ができるかわかりやすくなる(例: `pj job[TAB]`でジョブ操作が一覧に)
  - `pj sub job.sh` ⇒ `pj job submit job.sh`
  - `pj show rsc` ⇒ `pj node ls`
- 標準化でジョブスクリプトもポータブルに
  - HPCIの全スパコンでwrite once, run anywhere
  - スパコン間でジョブやデータが転送できるとクラウド並みの可用性に

## ゲーム産業からHPCへの期待

- HPCリソースをエッジデバイスに届けようとする試みが盛んに
  - [NVIDIAとGoogle Cloud、あらゆる場所でのAI開発を加速するため業界初の「AI-on-5G Lab」を設立](#)
- 今後もスパコン＝研究限定でいいのか？ 研究・開発・運用は一体化する時代に
  - [スパコン富岳、日本の産業強化に貢献できないジレンマ](#)
- クラウドゲーミングへのHPC応用に期待
  - ゲーム処理をクラウドで計算し、画面をスマホなどに転送
  - 注目の技術だが、遅延に課題([ある環境では1080p/30fpsで166msの遅延](#))
  - CDNのようにエッジデバイス付近のHPCクラスタを自動で使う仕組みができあがれば、スマホで据え置き器並みにハイクオリティなゲームができるかも!?



Copyright (c) KLab Inc. All Rights Reserved.