

卒業論文 2015 年度（平成 27 年）

技術系 Meetup における
プレゼンテーション記録システムの設計と実装

慶應義塾大学 環境情報学部

氏名：高橋 俊成

技術系 Meetup における
プレゼンテーション記録システムの設計と実装

本研究では、技術系 Meetup におけるプレゼンテーションを記録するために、専用のプレゼンテーション記録システムを設計、構築した。

Meetup とは、共通の趣味やテーマで緩やかに繋がる WEB コミュニティのメンバーが、知見共有や人間交流を目的として行うカジュアルな勉強会・交流会である。Meetup におけるプレゼンテーションでは、コミュニティにとって有意義な情報交換が行われている一方で、その記録作業は参加者個々人の自発的・自主的な活動に頼りきっており、成果をオンライン上のコミュニティに十分に還元できていない。

本研究では、まず技術系 Meetup におけるプレゼンテーションをオンラインで共有しやすい形で記録するための要件を整理した。次にそれを満たす専用のプレゼンテーション記録システムを設計し、WebRTC や Electron 等の技術を用いて実装した。終わりに、その評価のために実証実験を実施し、本システムが技術系 Meetup におけるプレゼンテーションの記録に有用であることを確認した。

キーワード：

1. Meetup
2. プレゼンテーション,
3. アーカイブ,
4. WebRTC,
5. Electron

Abstract of Bachelor's Thesis

Design and implementation of a presentation record system
for technical meetup

目次

第 1 章	はじめに	1
1.1.	背景	1
1.2.	本研究の課題	2
1.3.	本研究の目的	2
1.4.	用語定義	3
1.5.	本論文の構成	4
第 2 章	MEETUP におけるプレゼンテーションの分析	5
2.1.	MEETUP におけるプレゼンテーションの特徴	5
2.2.	MEETUP におけるプレゼンテーションの構成要素	6
第 3 章	既存のスライドフォーマットとその問題点	8
3.1.	既存のスライドフォーマットと記録可能なデータ	8
3.2.	既存のスライドフォーマットの問題点	8
3.2.1.	OOXML	8
3.2.2.	MPEG4	9
3.2.3.	HTML	9
第 4 章	MEETUP におけるプレゼンテーション記録システムの設計	10
4.1.	記録システムの要件定義	10
4.1.1.	記録するプレゼンテーション構成要素	10
4.1.2.	遠隔の参加者の行動をスライドタイムラインで記録するための設計	11
4.2.	記録システムの全体設計	11
4.2.1.	システム化プレゼンテーションフローの設計	11
4.3.	発表者用クライアントアプリケーションの設計	13
4.3.1.	発表者用クライアントアプリケーションの振る舞い	15
4.3.2.	発表者用クライアントアプリケーションの流れと画面	16
4.4.	聴衆用クライアントアプリケーションの設計	18
4.4.1.	聴衆用クライアントアプリケーションの振る舞い	18
4.4.2.	聴衆用クライアントアプリケーションの流れと画面	19
4.5.	サーバーアプリケーションの設計	20
4.5.1.	サーバーアプリケーションの振る舞い	20
4.6.	プレゼンテーション記録フォーマットの設計	22
4.6.1.	JSON 出力の記録フォーマットの設計	22
4.6.2.	Markdown 出力の記録フォーマットの設計	25

第 5 章	MEETUP におけるプレゼンテーション記録システムの実装	27
5.1.	実装環境	27
5.2.	実装の前提となる技術	27
5.2.1.	WebRTC.....	28
5.2.2.	Electron (Atom Shell)	28
5.3.	MEETUP におけるプレゼンテーション記録システムの実装の概要	28
第 6 章	実証実験	30
6.1.	輪講会における実証実験	30
6.2.	デベロッパーMEETUP における実証実験.....	30
第 7 章	評価	31
7.1.	評価概要	31
7.2.	計測	31
7.2.1.	映像配信時の参加者のスケーラビリティ	31
7.2.2.	対応するスライド共有サイトの数.....	31
7.3.	考察	31
第 8 章	結論	32
8.1.	まとめ	32
8.2.	今後の課題	32
謝辞.....		33
参考文献.....		34

図目次

図 1	WEB コミュニティの活動と情報の流れ.....	1
図 2	Meetup プレゼンテーション記録フォーマットの概念図.....	3
図 3	一般的なプレゼンテーションと Meetup におけるプレゼンテーションの構造比較.....	6
図 4	Meetup プレゼンテーション記録フォーマットの概念図（再掲）.....	7
図 5	システム化前のプレゼンテーションフロー全体図.....	12
図 6	システム化プレゼンテーションフロー全体図.....	13
図 7	発表者用クライアントアプリケーションの画面遷移.....	17
図 8	聴衆用クライアントアプリケーションの画面遷移.....	20
図 9	プレゼンテーション記録フォーマット(JSON)の例.....	23
図 10	プレゼンテーション記録フォーマット(Markdown)の例.....	26
図 11	Meetup におけるプレゼンテーション記録システムの実装概要.....	29

表目次

表 1	スライドフォーマットと記録可能コンテンツ	8
表 2	プレゼンテーションの記録対象とデータ形式	10
表 3	発表者用クライアントアプリケーションの機能一覧	15
表 4	聴衆用クライアントアプリケーションの機能一覧	18
表 5	サーバーアプリケーションの機能一覧	20
表 6	本研究のプレゼンテーション記録システムの実装環境	27

第1章 はじめに

1.1. 背景

WEB コミュニティは 1990 年代から始まった情報通信技術の発達の中で登場し、今日まで発展してきた。古くはメーリングリストや IRC で、現在では SNS で、活発な非対面・テキスト主体のコミュニケーションが行われてきた。これらは同じ趣味やテーマによって緩やかに繋がる同質性の高いコミュニティである。

WEB コミュニティのメンバーは時々、より密なコミュニケーションを求めて Meetup と呼ばれるオフラインイベントを開催することがある。Meetup は、知見共有・人間交流を目的とした対面・リアルタイムのカジュアルな勉強会・交流会である。Meetup の形態はコミュニティごとに様々だが、とくにエンジニア界隈の技術系 Meetup では設定されたテーマに沿って参加者のうち少人数が発表者としてプレゼンテーションを行うセミナースタイルが主流である。Meetup の特徴は、その母体となるコミュニティの活動が全て WEB 上で行われているという点にある。Meetup 開催前後のやりとりは全て WEB で、Meetup だけがオフラインでのリアルイベントである。そのため、Meetup で共有された情報をできるだけ抜けのないうように記録し WEB に還元することが、コミュニティの継続的な発展に重要だと言える（図 1）。

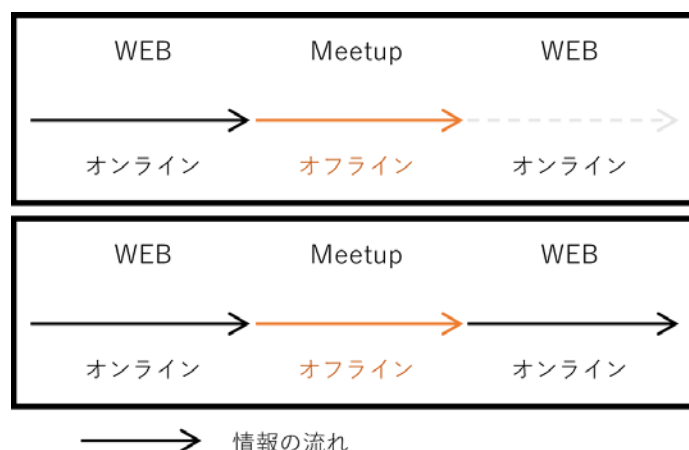


図 1 WEB コミュニティの活動と情報の流れ

しかしながら Meetup の多くは個人が趣味的に開催するため、企業主催のカンファレンスのように発表や質疑応答に専任の撮影者や速記者が用意されることは稀である。現状 Meetup におけるプレゼンテーションの記録手法は、聴衆によるボランティア的な文字起こ

しか、そもそも全く記録していないかのどちらかである。聴衆による文字起こしとは、有志がマイクロブログサービス等にプレゼンテーションの内容やその後の質疑応答を同時進行的にテキストに書き起こし、まるで実況中継のようにマイクロブログサービスへ投稿する手法である。マイクロブログサービスはその性質上、投稿ひとつひとつが独立しており、また時間がたつと投稿内容を参照することが難しくなるので、さらにマイクロブログまとめサービスを用いて投稿された文字起こしを編纂することもある。

この手法の問題点は、文字起こしの負担が大きい点と、聴衆の自主的・献身的な活動に頼りきっている点、そして記録の品質が保証されない点である。リアルタイムに書き起こして投稿する作業は高度なタイピング能力と集中力を求められ、しかもそれは聴衆の自主的・献身的な活動に支えられている。文字起こしができ、かつやりたいという聴衆が名乗り出ない限り記録作業が行われない。また、運よく献身的な聴衆が現れたとしても、その記録フォーマットに統一されたものがないために、個々人の能力や主観によって記録内容が欠けたり、変化したりする恐れがある。

1.2. 本研究の課題

本研究が着目する問題点は、技術系 Meetup のプレゼンテーションにおいて専任の記録担当が用意されないために、そこで共有された知見を正確に記録し WEB コミュニティにきちんと還元できていない点である。

そこで本研究では、技術系 Meetup 専用のプレゼンテーション記録システムを構築することで、参加者の献身性に頼ることなくプレゼンテーションを記録し WEB コミュニティに還元できる環境の実現を目指す。

1.3. 本研究の目的

本研究では、Meetup におけるプレゼンテーションの記録を目的とする。そのためのアプローチとして、プレゼンテーションのリアルタイム WEB 配信機能を備えた専用のプレゼンテーション記録システムを構築する。

このプレゼンテーション記録システムは、Meetup におけるプレゼンテーションを全て WEB 上で配信しながら記録する。これにより、オフラインで行われている Meetup を文字起こし等の変換作業なしに記録できる。記録データは、発表者のスライド資料に聴衆の情報や活動を付加したテキスト主体の「Meetup プレゼンテーション記録フォーマット」とすることで、WEB で活動するコミュニティが参照しやすくする。

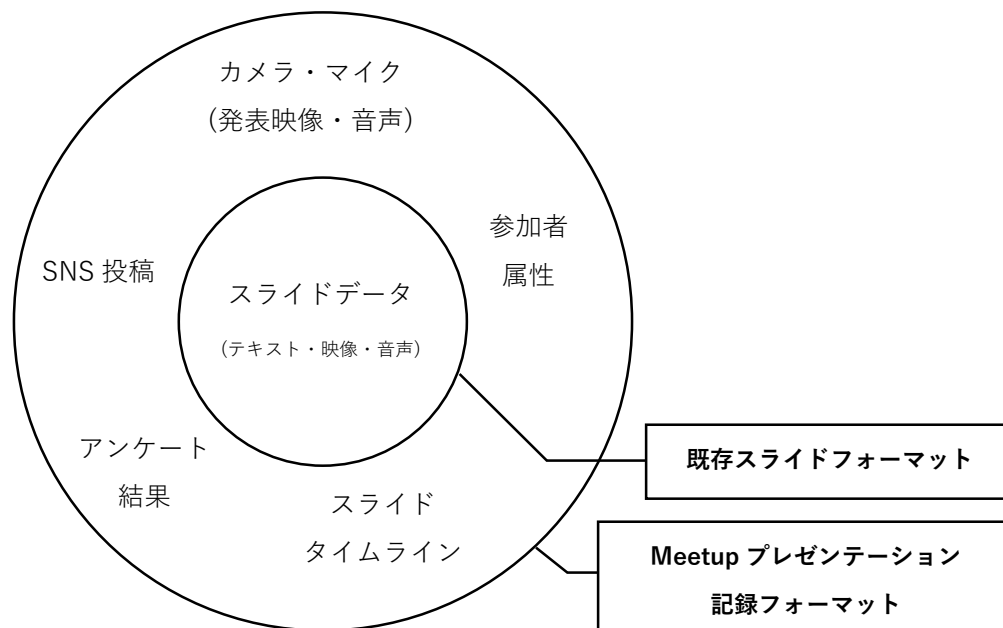


図 2 Meetup プレゼンテーション記録フォーマットの概念図

1.4. 用語定義

本節では、本論文中で用いられる主な用語を定義する。

- Meetup（ミートアップ）
主に WEB 上で活動するコミュニティのメンバーが、知見・意見共有と人間交流を目的として行うカジュアルな勉強会や交流会。参加者のうち数人が発表者となりプレゼンテーションを行う。発表内容は、後日発表者によって任意にオンライン公開され、WEB コミュニティに共有されることがある。
- WEB コミュニティ
主に WEB 上で活動する非組織的なコミュニティ。共通の趣味・テーマなどの情報共有・意見交換・人間交流を主な目的に、緩やかに繋がっている。特定の場所や組織に実際に行かなくともネットワークを経由して知的資産を得られる場として機能する。
- プレゼンテーション
発表者がスライド資料を用いて聴衆に対して情報を掲示する行為と、それに付随する参加者同士の議論や意見交換。

- スライドタイムライン
プレゼンテーションの進行を、経過時間に加えてスライド資料のページ番号に基づいて表したもの。

1.5. 本論文の構成

2 章では技術系 Meetup におけるプレゼンテーションを分析し、問題解決のためのアプローチを提案する。3 章では既存のスライドフォーマットとその問題点について分析する。4 章では前章までの分析に基づいて Meetup におけるプレゼンテーション記録システムを設計し、5 章で実装について述べる。6 章では本システムを用いた実証実験について述べ、7 章でその評価を行う。8 章では結論と今後の課題について述べる。

第2章 Meetup におけるプレゼンテーションの分析

本章では、Meetup におけるプレゼンテーションを分析し、問題解決のアプローチを述べる。

2.1. Meetup におけるプレゼンテーションの特徴

WEB コミュニティによって、開催される Meetup の形態は若干異なる。Meetup を積極的に行う人気のコミュニティには、以下が挙げられる [1]。

- 外国語学習
- ブッククラブ
- 国際交流
- ビジネス、起業家
- ヨガ
- テクノロジー、プログラマー
- 写真撮影
- ママ
- 共通の病
- ランニング

本研究では、プログラマーコミュニティに属する Meetup で行われるプレゼンテーションを Meetup におけるプレゼンテーションとして定義する。

Meetup におけるプレゼンテーションでは、司会と発表者、聴衆の3種類の参加者が同じ時間を共有しているものとする。司会は Meetup 全体の進行管理を行う者とし、発表者は与えられた時間内でテーマに沿ってプレゼンテーションを行う者、聴衆は発表者のプレゼンテーションを視聴する者とする。司会は発表者または聴衆を兼ねることがあり、また発表者は自身のプレゼンテーション時以外では聴衆として振る舞う。

Meetup におけるプレゼンテーションは、スライド資料を用いて行われる比較的短時間の発表並びに質疑応答である。Meetup ごとにテーマが定められ、参加者のうち数人が発表者としてそれに則ったプレゼンテーションを順に行う。この時、聴衆は発表後の質疑応答のほ

か、発表中に同時進行的にマイクロブログサービスへ感想や意見を投稿するなどして発表者および他の聴衆とコミュニケーションをとる。

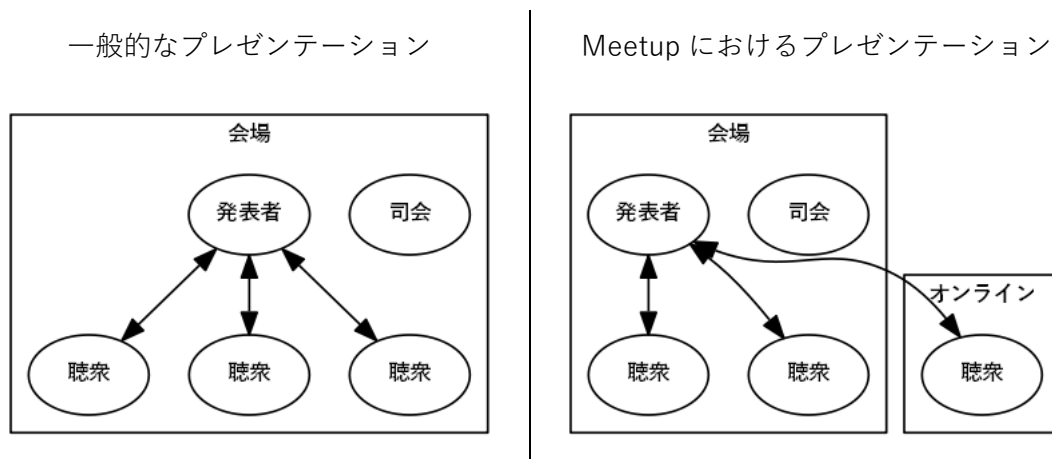


図 3 一般的なプレゼンテーションと Meetup におけるプレゼンテーションの構造比較

図 2 に示すように、Meetup におけるプレゼンテーションの特徴はインターネット越しに視聴する遠隔の参加者の存在である。一般的なプレゼンテーションはすべての参加者が同一の物理的な空間内に存在するのに対して、Meetup におけるプレゼンテーションでは参加者の一部がインターネット越しに遠隔で視聴する。主催者が会場の様子をストリーミング配信したり、発表者が事前に発表資料をスライド共有サービスに投稿したり、現地の参加者が進行や感想をマイクロブログにリアルタイムに投稿したりすることで、このような遠隔での参加を実現している。

2.2. Meetup におけるプレゼンテーションの構成要素

図 3 は、Meetup におけるプレゼンテーションの構成要素を表したものである。Meetup におけるプレゼンテーションでは、まず参加者がいて、次に発表者のスライド資料やそのコントロール情報（スライドタイムライン）があり、そして放送者や参加者の表情や身振りなどの視覚情報、発話や拍手などの音声情報がある。これらの基本的な構成要素に加えて、主に遠隔の参加者による SNS 投稿など会場の外の情報も構成要素として挙げられる。

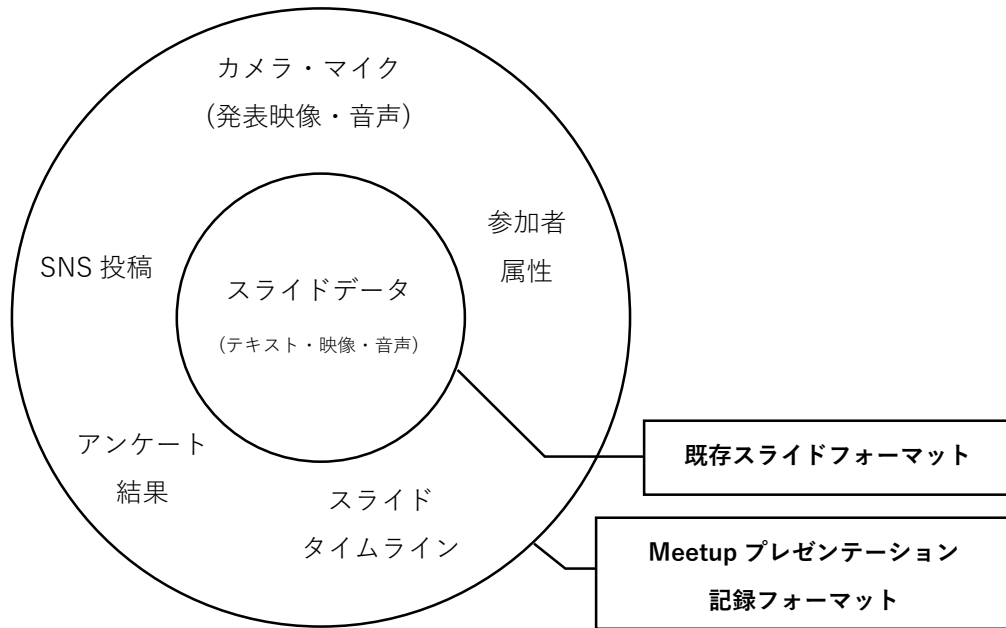


図 4 Meetup プレゼンテーション記録フォーマットの概念図 (再掲)

第3章 既存のスライドフォーマットとその問題点

本章では、スライド資料における既存フォーマットを分析し、それぞれの問題点を整理する。

3.1. 既存のスライドフォーマットと記録可能なデータ

スライド資料に用いられるフォーマットと表現可能なデータの組み合わせの可否を表 1 に示す。

表 1 スライドフォーマットと記録可能コンテンツ

	OOXML	MP4	HTML
テキスト	○	×	○
静止画	○	○	○
映像/音声	○	○	×

3.2. 既存のスライドフォーマットの問題点

本節では、既存のスライドフォーマットの問題点についてそれぞれ述べる。

3.2.1. OOXML

OOXML (Open Office XML, ISO/IEC 29500:2012)は、Microsoft PowerPoint などのスライドツールで用いられる XML ベースのファイルフォーマット。

テキストや画像、映像などのマルチメディア資料に対応し、表現力も普及度も随一のスライドフォーマットである。しかし、高機能ゆえに非常に複雑で、OOXML フォーマットを扱えるアプリケーションは少ない。

3.2.2. MPEG4

MPEG4 は WEB で共有される動画データの中で最も一般的なファイルフォーマットのひとつである。

MPEG4 はどんな要素でもビットマップ化・映像化すれば記録できるので、見た目の表現力は OOXML 以上といえる。しかし、元がテキストデータの場合は映像化の際にその意味を失うことになるため、同じテキストデータを用意しても内容を検索することはできない。

3.2.3. HTML

WEB で最も普及している文書フォーマットのひとつ。構造化されたテキストデータなので内容の検索性が高い。video タグを用いればマルチメディアも参照できるが、あくまで参照であり単体での表現力には欠ける。

第4章 Meetup におけるプレゼンテーション記録システムの設計

本章では、これまでの分析を踏まえた上で、Meetup におけるプレゼンテーションの記録を実現するために必要なシステムを設計する。

4.1. 記録システムの要件定義

本節では、Meetup におけるプレゼンテーション記録システムに必要な機能要件を整理する。

- 2.2 節で述べた Meetup におけるプレゼンテーションの構成要素を記録できる
 - 構成要素のセマンティック（意味解釈可能）さを保ったまま記録できる
 - Meetup 会場にいない遠隔参加者の活動を記録できる
- スライドタイムラインに基づいてプレゼンテーションの構成要素を記録できる
- 作成した記録を WEB で容易に共有できる

4.1.1. 記録するプレゼンテーション構成要素

本システムで記録対象とするプレゼンテーション構成要素を以下に挙げる。

表 2 プレゼンテーションの記録対象とデータ形式

記録対象	データ形式	付随するメタ情報
スライド資料	テキスト(URL)	発表者
参加者属性(連絡先情報)	テキスト	
スライドコントロール (ページめくり、発表時間)	テキスト	
カメラ映像	ビデオ	動画コーデック、撮影対象、撮影者
マイク音声	サウンド	音声コーデック、録音対象、録音者
SNS 投稿	テキスト(URL)	投稿者、投稿内容、投稿時刻、ページ番号
アンケート	テキスト	質問内容、質問時刻、選択肢ごとの回答者数

スライド資料は、スライド共有サイトの利用が一般化していることを鑑みて、データそのものを持たずに外部 URL への参照という形で記録する。

4.1.2. 遠隔の参加者の行動をスライドタイムラインで記録するための設計

発表者と会場にいない遠隔の参加者との間で伝送遅延が大きい場合、遠隔の参加者の行動をスライドタイムラインで記録することが難しくなる。例えば遠隔の参加者がスライドの 4 ページ目にコメントをしたつもりで、その情報が中央サーバーあるいは発表者の端末に届く前に発表者がスライドを 5 ページ目にめくった場合、システムは遠隔の参加者が 5 ページコメントしたものと誤って判断してしまう。これを防ぐには、発表者がスライドのどのページを現在表示しているかをすべての参加者にブロードキャストする必要がある。

この時、スライドのページ番号だけでなくプレゼンテーションのすべての構成要素を参加者にブロードキャストすることで、システムは肥大化するものの別途プレゼンテーションの配信システムを用意する必要がなくなり、遠隔の視聴者の視聴環境が改善される。

よって、本システムはプレゼンテーションの記録のために、その配信も行えるよう設計する。

4.2. 記録システムの全体設計

本節では、前節で整理した機能要件を満たす設計について述べる。

4.2.1. システム化プレゼンテーションフローの設計

図 5 で、本システムを用いたプレゼンテーションの大まかな流れを表す。

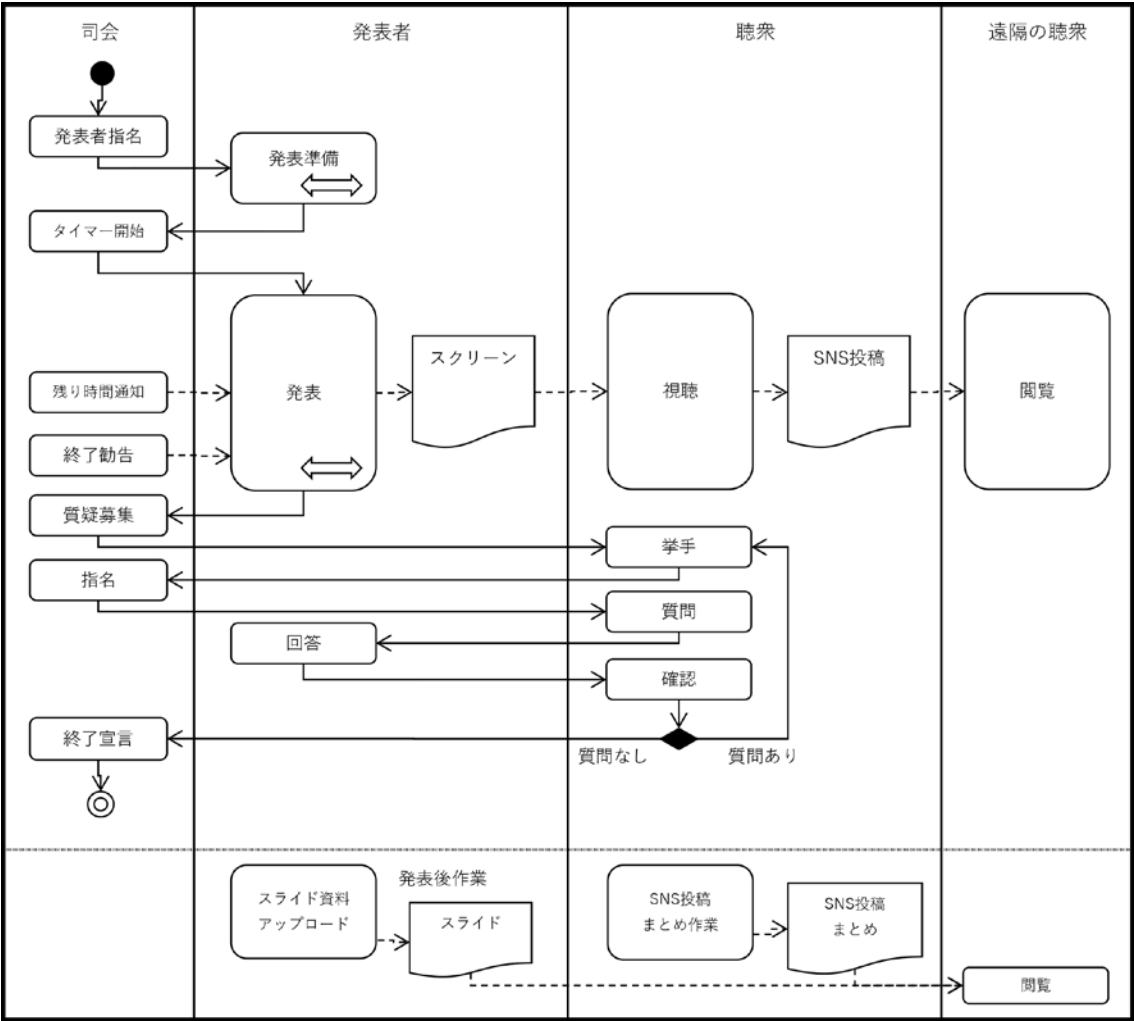


図 5 システム化前のプレゼンテーションフロー全体図

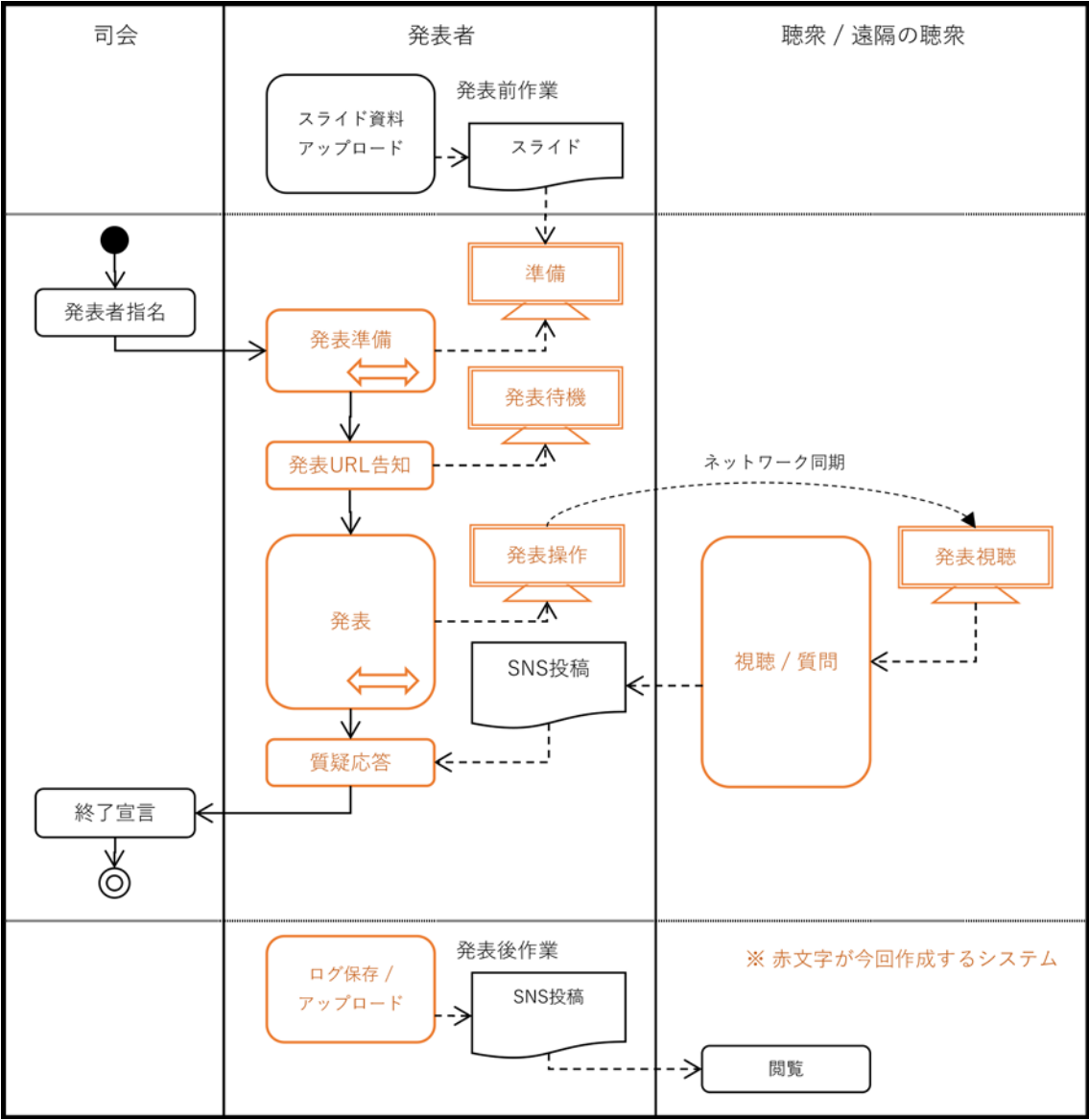


図 6 システム化プレゼンテーションフロー全体図

あああ

4.3. 発表者用クライアントアプリケーションの設計

本節では、本システムのうち発表者用クライアントアプリケーションの設計について述べる。

4.3.1. 発表者用クライアントアプリケーションの振る舞い

本項では、本研究で実装する発表者用クライアントアプリケーションの機能及び動作の概要を述べる。まず、表 3 で本アプリケーションの機能の一覧を示す。

表 3 発表者用クライアントアプリケーションの機能一覧

No.	大機能	小機能	機能概要
P1	発表者登録	発表者情報取得	保存された発表者のアカウント情報を取得する
P2		GitHub 認証要求	サーバーから GitHub アカウントの認証用 URL を取得し、標準ブラウザに表示する
P3		GitHub PIN 入力	GitHub PIN コードの入力を受け付ける
P4		GitHub 認証	入力された GitHub PIN コードをサーバーに送り、GitHub アカウント情報を取得する
P5		Google 認証要求	サーバーから Google アカウントの認証用 URL を取得し、標準ブラウザに表示する
P6		Google PIN 入力	Google PIN コードの入力を受け付ける
P7		Google 認証	入力された Google PIN コードをサーバーに送り、Google アカウント情報と API アクセストークンを取得する
P8		Twitter 認証要求	サーバーから Twitter アカウントの認証用 URL を取得し、標準ブラウザに表示する
P9		Twitter PIN 入力	Twitter PIN コードの入力を受け付ける
P10		Twitter 認証	入力された Twitter PIN コードをサーバーに送り、Twitter アカウント情報を取得する
P11		認証エラー	認証や情報取得の失敗時にエラーを通知する
P12		発表者情報保存	発表者の GitHub, Google, Twitter アカウント情報を保存する
P13	発表準備	発表情報入力	発表概要の入力を受け付ける
P14		スライド手動設定	スライド資料を URL 手動入力で設定する
P15		Google スライド一覧表示・設定	発表者の Google スライドに登録されているスライド資料の一覧を表示し、指定されたものを資料として設定する
P16		SlideShare スライド一覧表示・設定	任意の SlideShare アカウントに登録されているスライド資料の一覧を表示し、指定されたものを資料として設定する
P17		Web カメラ設定	発表者 PC に接続された Web カメラの配信有無を設定する
P18		発表情報送信	入力された発表情報を WEB サーバーに送信する
P19	発表告知	視聴用 URL 取得	Web サーバーから視聴ページの URL を取得する

P20		視聴用 URL 表示	視聴ページの URL を表示する
P21		視聴用 URL 投稿	発表者の Twitter アカウントに視聴ページの URL を投稿する
P22	発表操作	スライド操作	スライド資料の表示ページを変更する
P23		ポインタ操作	ポインタでスライド資料の一点を指し示す
P24		デスクトップ配信 切り替え	スライド操作とデスクトップ配信を切り替える
P25	発表配信	スライド操作配信	発表者のスライド操作を WEB サーバーに送信する
P26		ポインタ操作配信	発表者のポインタ操作を WEB サーバーに送信する
P27		Web カメラ配信	発表者 PC に接続された Web カメラ映像をメディアサーバーに送信する
P28		デスクトップ配信	発表者 PC のデスクトップキャプチャ映像をメディアサーバーに送信する
P29	質疑応答	コメント受信	聴衆からのコメントを受信する
P30		コメント一覧表示	聴衆からのコメントを新着順に一覧表示する
P31		コメントリンク表示	コメントへのリンクを関連するスライド資料の特定ページ・特定座標に表示する
P32	記録	ログ受信	Web サーバーからログデータを受信する
P33		ログ表示	ログデータを表示する
P34		ローカル保存	ログデータをテキスト形式(TSV, Markdown)にして保存する
P35		Gist 投稿	ログデータを発表者の Gist に投稿する

発表者用クライアントは、Meetup におけるプレゼンテーションの構成要素のうち発表者が発信する情報を処理し記録用 Web サーバーに送信する機能と、Web サーバーから受信した参加者の SNS 投稿を表示する機能を持つクライアントアプリケーションである。

発表者があらかじめスライド共有サイトに投稿しておいたスライド資料の URL を本アプリケーションに入力することで、即座にプレゼンテーションの配信と記録が可能になる。プレゼンテーションを開始するとそのプレゼンテーション固有の URL が生成される。ほかの視聴者がこの URL を Web ブラウザーで開けば、後述する聴衆用 Web アプリケーションですぐにプレゼンテーションへの参加が可能となる。

発表用クライアントの画面にはスライド資料と自身のビデオ映像、そして参加者の SNS 投稿が表示され。マウスやキーボードでスライドコントロールが可能である。

4.3.2. 発表者用クライアントアプリケーションの流れと画面

図4で、本システムの発表者用クライアントの画面遷移を示す。

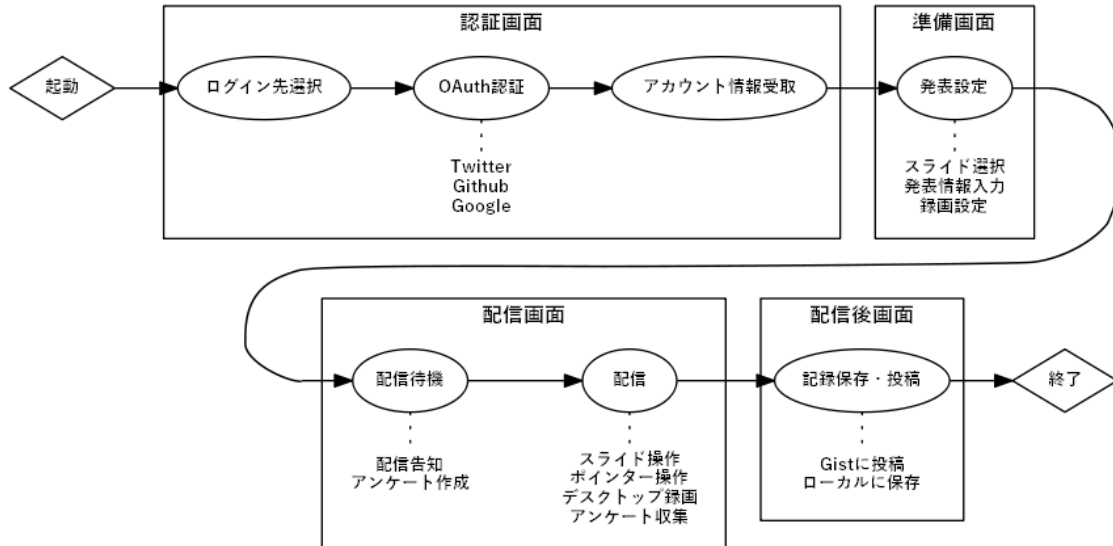


図7 発表者用クライアントアプリケーションの画面遷移

本システムの発表者用クライアントは、認証画面、準備画面、配信画面、配信後画面の4つの画面によって構成される。本論文では、画面を画面レイアウトの単位とする。それぞれの画面は1つ以上のステップで構成され、0個以上のビューを内包する。本論文において、ステップはユーザーインターフェース（入力可能なユーザー操作の組み合わせ）の単位とし、ビューはユーザー操作をきっかけに同一画面にオーバーレイ表示されるコンテンツの単位とする。

4.3.2.1. 認証画面

（画面レイアウト）

発表者のユーザー情報を入力する画面で、3つのステップで構成され、3つのビューが含まれる。

4.3.2.2. 準備画面

（画面レイアウト）

発表者が配信のための準備をする画面で、3つのビューが含まれる。

4.3.2.3. 配信画面

(画面レイアウト)

発表者が配信を行う画面で、2つのステップで構成され、6つのビューが含まれる。

4.3.2.4. 配信後画面

(画面レイアウト)

発表者が配信の記録を保存や共有するための画面で、2つのビューが含まれる。

あああ

4.4. 聴衆用クライアントアプリケーションの設計

本節では、本システムのうち聴衆用クライアントアプリケーションの設計について述べる。

4.4.1. 聴衆用クライアントアプリケーションの振る舞い

本項では、本研究で実装する聴衆用クライアントアプリケーションの機能及び動作の概要を述べる。まず、表4で本アプリケーションの機能の一覧を示す。

表4 聴衆用クライアントアプリケーションの機能一覧

No.	大機能	小機能	機能概要
A1	認証	聴衆情報取得	聴衆のアカウント情報を取得する
A2		Twitter 認証要求	サーバーから Twitter アカウントの認証用 URL を取得し、遷移する
A3		Twitter 認証	リダイレクトを解析して得たアクセストークンをサーバーに送り、Twitter アカウント情報を取得する
A4		認証エラー	Twitter 認証や情報取得に失敗したときにエラーを通知する
A5		発表者情報保存	発表者のセッション情報、Twitter アカウント情報を保存する
A6	入退室	入室	発表 URL に紐づいた配信への接続をサーバーに要求する

A7		入室エラー	配信に接続できないときにエラーを通知する
A8		退室	視聴画面を閉じたときにサーバーへ離脱を通知する
A9	視聴	発表情報取得	発表情報を取得する
A10		スライド資料表示	発表に紐づいたスライドデータを表示する
A11		スライド同期	発表者のページ更新通知をサーバーから受け取り同期する
A12		スライド別窓表示	スライドを別窓で、聴衆が任意に操作可能な状態で開く
A13	コメント	コメント表示	サーバーからコメント通知を受け取り同期・表示する
A14		コメント入力/投稿	入力されたコメントを表示中のスライドページまたはその特定座標(矩形)と紐づけてサーバーに送る
A15		マイク録音/投稿	聴衆端末のマイク音声を録音し、表示中のスライドページまたはその特定座標(矩形)と紐づけてサーバーに送る
A16		スライド領域選択	スライドの特定座標(矩形)を選択する
A17	ポインタ	ポインタ同期	サーバーからポインタ通知を受け取り同期・表示する
A18	アンケート	アンケート表示	サーバーからアンケート通知を受け取り表示する
A19		アンケート受付	サーバーから✓切通知を受け取るまで、回答を受け付ける
A20		アンケート送信	アンケートの回答をサーバーに送信する
A21		アンケート結果表示	アンケート結果をグラフにして表示する
A22	映像	Web カメラ表示	サーバーから発表者の Web カメラ映像を受信し表示する
A23		デスクトップ表示	サーバーから発表者のデスクトップ映像を受信し表示する

聴衆用アプリケーションは、Meetup におけるプレゼンテーションの構成要素のうち聴衆の SNS 投稿を処理しサーバーに送信する機能と、発表者の発表情報と他の参加者の SNS 投稿をサーバーから受け取り表示する機能を持つアプリケーションである。

発表者から伝えられたプレゼンテーション URL を Web ブラウザーで開くと、この聴衆側アプリケーションで即座にプレゼンテーションに参加できる。聴衆用アプリケーションの画面にはスライド資料と発表者のビデオ映像、そして参加者の SNS 投稿が表示され、マウスやキーボード、タッチインターフェースで SNS 投稿やアンケートへの回答が可能である。

4.4.2. 聴衆用クライアントアプリケーションの流れと画面

図 7 で、本システムの聴衆用クライアントの画面遷移を示す。

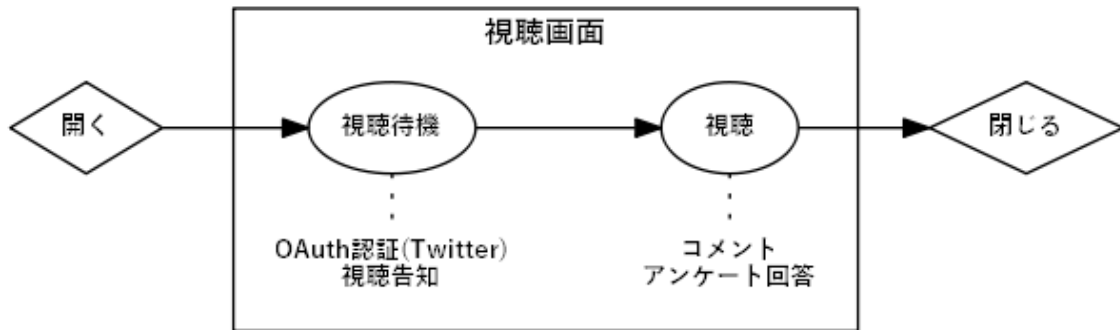


図 8 聴衆用クライアントアプリケーションの画面遷移

本システムの聴衆用クライアントは、視聴画面のみによって構成される。

4.4.2.1. 視聴画面

(画面レイアウト)

あああ

4.5. サーバーアプリケーションの設計

本節では、本システムのうちサーバーアプリケーションの設計について述べる。

4.5.1. サーバーアプリケーションの振る舞い

本項では、本研究で実装するプレゼンテーション記録システムのサーバーアプリケーションの動作概要を述べる。まず、表 5 で本アプリケーションの機能の一覧を示す。

表 5 サーバーアプリケーションの機能一覧

No.	大機能	小機能	機能概要
S1	発表者認証	GitHub 認証開始	発表者クライアントに GitHub 認証用 URL を渡す
S2		GitHub 認証	発表者クライアントから GitHub PIN コードを受け取り、生成したアクセストークンで GitHub アカウント情報を取得し返す
S3		Google 認証開始	発表者クライアントに Google 認証用 URL を渡す
S4		Google 認証	発表者クライアントから Google PIN コードを受け取り、生成したアクセストークンで Google アカウント情報を取得し返す

S5		Twitter 認証開始	発表者クライアントに Twitter 認証用 URL を渡す
S6		Twitter 認証	発表者クライアントから Twitter PIN コードを受け取り、生成したアクセストークンで Twitter アカウント情報を取得し返す
S7	聴衆認証	Twitter 認証開始	聴衆クライアントに Twitter 認証用 URL を渡す
S8		Twitter 認証	聴衆クライアントからアクセストークンを受け取り、Twitter アカウント情報を取得し返す
S9	発表告知	Twitter 共有	各クライアントからの要求に応じて、発表タイトル及び URL を Twitter アカウントに投稿する
S10	発表管理	発表生成	発表者クライアントから発表情報を受け取り、一意の発表 URL を生成して返す
S11		発表接続	発表 URL に応じて、発表者クライアントと聴衆クライアントを接続する
S12		発表終了	発表者クライアントからの終了通知を受け取り、関連する接続の切断と発表 URL の無効化を行う
S13	発表配信	基本情報同期	接続した聴衆クライアントに発表情報を送信する
S14		スライド同期	発表者クライアントからのページめくり信号を関連する全聴衆クライアントにブロードキャストする
S15		ポインタ同期	発表者クライアントからのポインタ信号を関連する全聴衆クライアントにブロードキャストする
S16		映像配信	発表者クライアントからのリアルタイム配信映像を、関連する全聴衆クライアントにブロードキャストする
S17		映像切り替え同期	発表者クライアントからの映像切り替え信号を関連する全聴衆クライアントにブロードキャストする
S18	アンケート	アンケート開始	発表者クライアントからのアンケート要求とその内容を関連する全聴衆クライアントにブロードキャストする
S19		アンケート終了	アンケートの締め切り通知を関連する全聴衆クライアントにブロードキャストする
S20		アンケート回収	聴衆クライアントからのアンケート回答を受け取り、集計する
S21		アンケート集計	締め切り時点での集計結果を関連する全てのクライアントにブロードキャストする
S22	コメント	コメント同期	聴衆クライアントからコメントを受け取り、関連する発表者クライアントと他の聴衆クライアントにブロードキャストする
S23		コメント投稿	Twitter 認証された聴衆クライアントからコメントを受け取り、聴衆の Twitter タイムラインにコメントを投稿する

S24		音声コメント	聴衆クライアントから録音を受け取り、関連する発表者クライアントと他の聴衆クライアントにブロードキャストする
S25	発表記録	コメント記録	同期した全てのコメントデータを記録する
S26		映像記録	配信した全ての映像データを記録する
S27		スライド記録	スライド資料の URL 及びページ数、各ページのサムネイル画像を記録する
S28		文字起こし	記録されたコメントデータとスライドデータを元に、ページごとにコメントをまとめたテキストログを作成し、発表者クライアントに送信する
S29		Gist 投稿	発表者クライアントからの要求に応じて、文字起こしで生成したテキストログを発表者の Gist に投稿する
S30		ログ URL 通知	発表者クライアントにコメントデータ及び映像データをダウンロードできる URL を通知する

サーバーアプリケーションは、本システムにおいてクライアント間の情報の受け渡しとその記録を担う。Meetup における全てのプレゼンテーション構成要素を発表者用・聴衆用クライアントから受け取り、サーバー内に記録したうえで参加者全員に同じ情報をブロードキャストして伝える。

4.6. プレゼンテーション記録フォーマットの設計

本節では、本システムが記録するプレゼンテーションデータのフォーマット設計について述べる。技術系 Meetup におけるプレゼンテーションの記録データは WEB コミュニティでの利用が想定されるため、WEB と親和性の高い JSON および Markdown をコンテナフォーマットとして採用する。異なる 2 つのフォーマットを用いるのは、前者はプレゼンテーションの完全な記録に適し、後者はプレゼンテーションの Gist 等 SNS 上での容易な共有に適しているためである。

4.6.1. JSON 出力の記録フォーマットの設計

本項では、JSON をコンテナとする記録フォーマットの設計について述べる。まず、この記録フォーマットでの記録例を図 9 に示す。

```
{
  "title": "プレゼンテーションのタイトル",
  "description": "このプレゼンテーションはサンプルです。",
  "time_begin": 14518146780000,
  "time_end": 14518163010000,
  "presenter": {
    "name": "山田太郎",
    "twitter": "@example_yamada",
    "google": null,
    "github": "exampleyamada"
  },
  "audience": [{
    "name": "竹田花子",
    "twitter": "@example_hanako "
  }, {
    "name": "たまちゃん",
    "twitter": "@example_azarashi"
  }],
  "slide": [
    "http://example.com/slide.html",
    "http://example.com/slide.html#2",
    "http://example.com/slide.html#3"
  ],
  "attachment": [{
    "type": "video/webm",
    "url": "http://example.com/recerd.webm",
    "name": "録画映像"
  }],
  "activity": [{
    "time": 14518148620000,
    "type": "slide_update",
    "page ": 1
  }, {
    "time": 14518149240000,
    "type": "comment",
    "page": 1,
    "area": null,
    "text": "よろしくー",
    "audience_id": 0,
    "url": "http://twitter.com/status/9999999999"
  }, {
    "time": 14518182720000,
    "type": "finish"
  }
]
```

中略

図 9 プレゼンテーション記録フォーマット(JSON)の例

記録データには、プレゼンテーションのタイトルや時刻などの基本情報に加え、発表者情報、聴衆のリスト、スライド資料と各ページの URL、スライド資料以外の添付資料の URL、そして時系列順に整理された発表中のアクティビティのリストが含まれる。

各属性の詳細な構造を表 6 に示す

表 6 プレゼンテーション記録フォーマット(JSON)の属性詳細

属性名	型	親要素	必須	説明/備考
title	string	-	Y	発表者が指定した発表タイトル
description	string	-	N	発表者が指定した発表の説明文
time_begin	integer	-	Y	プレゼンテーションの開始時刻(ミリ秒単位の UNIX 時刻)
time_end	integer	-	Y	プレゼンテーションの終了時刻(ミリ秒単位の UNIX 時刻)
presenter	object	-	Y	発表者情報を格納するオブジェクト
name	string	presenter	Y	発表者の氏名またはニックネーム
twitter	string	presenter	N	発表者の Twitter スクリーンネーム
google	string	presenter	N	発表者の Google ID
github	string	presenter	N	発表者の GitHub ID
audience	array<object>	-	Y	聴衆情報を格納する配列。index は聴衆の ID と同義
name	string	audience	N	聴衆の氏名またはニックネーム
twitter	string	audience	N	聴衆の Twitter スクリーンネーム
slide	array<string>	-	Y	スライド資料とそのページごとの URL を格納する配列。index はページ番号と同義。ページごとに固有の URL が割り振られていないスライド資料の場合は要素が重複する場合がある
attachment	array<object>	-	N	スライド資料以外の添付資料を格納する配列。index は添付資料の ID と同義
type	string	attachment	Y	添付資料の MIME Type
url	string	attachment	Y	添付資料の参照先 URL
name	string	attachment	N	添付資料の名称
activity	array<object>	-	Y	プレゼンテーション内のイベントを時系列順に格納する配列。index は要素の ID と同義
time	integer	activity	Y	イベントの発生時刻 (ミリ秒単位の UNIX 時刻)
type	string	activity	Y	イベントの種類。begin, end, comment 等がある

page	integer	activity	N	イベントと紐づいたページ(slide 配列の index)
area	object	activity	N	ページの特定座標・領域を示す場合に用いる
x	integer	area	Y	スライド資料の横幅を 1.0 としたときの X 座標
y	integer	area	Y	スライド資料の縦幅を 1.0 としたときの Y 座標
width	integer	area	N	スライド資料の横幅を 1.0 としたときの領域幅
height	integer	area	N	スライド資料の縦幅を 1.0 としたときの領域高さ
text	string	activity	N	聴衆のコメント
url	string	activity	N	イベントに紐づいた URL
audience_id	integer	activity	N	イベントに紐づいた聴衆の ID。0 以上
attachment_id	integer	activity	N	イベントに紐づいた添付資料の ID。0 以上
activity_id	integer	activity		イベントに紐づいた他のイベントの ID。0 以上
question	object	-	N	アンケート情報
type	string	question	Y	アンケートの形式。single または multiple
text	string	question	Y	アンケートの質問文
option	array<object>	question	Y	アンケートの選択肢を格納する配列。2 要素以上
text	string	option	Y	アンケート選択肢の文章
number	integer	option	Y	アンケート選択肢の選択数。0 以上

4.6.2. Markdown 出力の記録フォーマットの設計

本項では、Markdown をコンテナとする記録フォーマットの設計について述べる。まず、この記録フォーマットでの記録例を図 10 で示す。


```
# プレゼンテーションのタイトル

## 発表概要
* 発表時刻: 2016/01/02 11:32 ~ 11:40
* [スライド資料] (http://example.com/slide.html)
* [添付資料 1 - 録画映像] (http://example.com/recerd.webm)

## 参加者
### 発表者
山田太郎 ([Twitter] (https://twitter.com/example\_yamada)),
[GitHub] (https://github.com/exampleyamada))
### 聴講者
* [竹田花子] (https://twitter.com/example\_hanako)
* [たまちゃん] (https://twitter.com/example\_azarashi)
ほか 4 名

## コメントログ
* [2 ページ目] (http://example.com/slide.html#2)
  * [よろしくー (竹田花子)] (https://twitter.com/exmple\_hanako/status/9999999999)
```

後略

図 10 プレゼンテーション記録フォーマット(Markdown)の例

第5章 Meetup におけるプレゼンテーション記録システムの実装

本章では、前章で設計したプレゼンテーション記録システムのうち、以下の実装について述べる。

- 聴衆用 WEB アプリケーション
- 発表者用 ネイティブアプリケーション
- WEB サーバー
- メディアサーバー

5.1. 実装環境

本研究で実装した Meetup におけるプレゼンテーション記録システムの実装環境を表 3 に示す。

表 7 本研究のプレゼンテーション記録システムの実装環境

名称	詳細
WEB サーバーOS	Ubuntu 14.04.3 LTS
WEB サーバー	Apache 2.4.7
WEB サーバーアプリケーション使用言語	Haxe (Node.js)
メディアサーバーOS	Ubuntu 14.04.3 LTS
メディアサーバー	Kurento Media Server 6.2.0
ネイティブクライアントエンジン	Electron 0.36.0
ネイティブクライアント使用言語	Haxe (JavaScript), HTML, CSS
WEB クライアント使用言語	Haxe (JavaScript), HTML, CSS

5.2. 実装の前提となる技術

5.2.1. WebRTC

本節では、本研究で実装した映像配信プロトコルとして使用する WebRTC のアーキテクチャについて述べる。WebRTC は、W3C が提唱する Web ブラウザー用のリアルタイムコミュニケーション API で、特別なプラグインなしでピアツーピアの映像伝送を可能とする。

5.2.2. Electron (Atom Shell)

本節では、本研究で実装した発表者用のネイティブクライアントのアプリケーションエンジンとして使用する Electron のアーキテクチャについて述べる。Electron は、GitHub 社が開発するデスクトップアプリのためのクロスプラットフォーム実行環境で、内蔵する Chromium ブラウザーによって Node.js/HTML アプリケーションをクロスプラットフォームに動作させることを可能とする。

5.3. Meetup におけるプレゼンテーション記録システムの実装の

概要

本節では、本研究で実装したプレゼンテーション記録システムの実装概要を述べる。実装の概要を図 4 に示す。本システムの実装は、発表者が使用するネイティブアプリケーションと、聴衆が使用する Web アプリケーション、ならびにそれらから発信される情報を受け取り記録・他の参加者に仲介する Web サーバー・Kurento メディアサーバーに大別される。

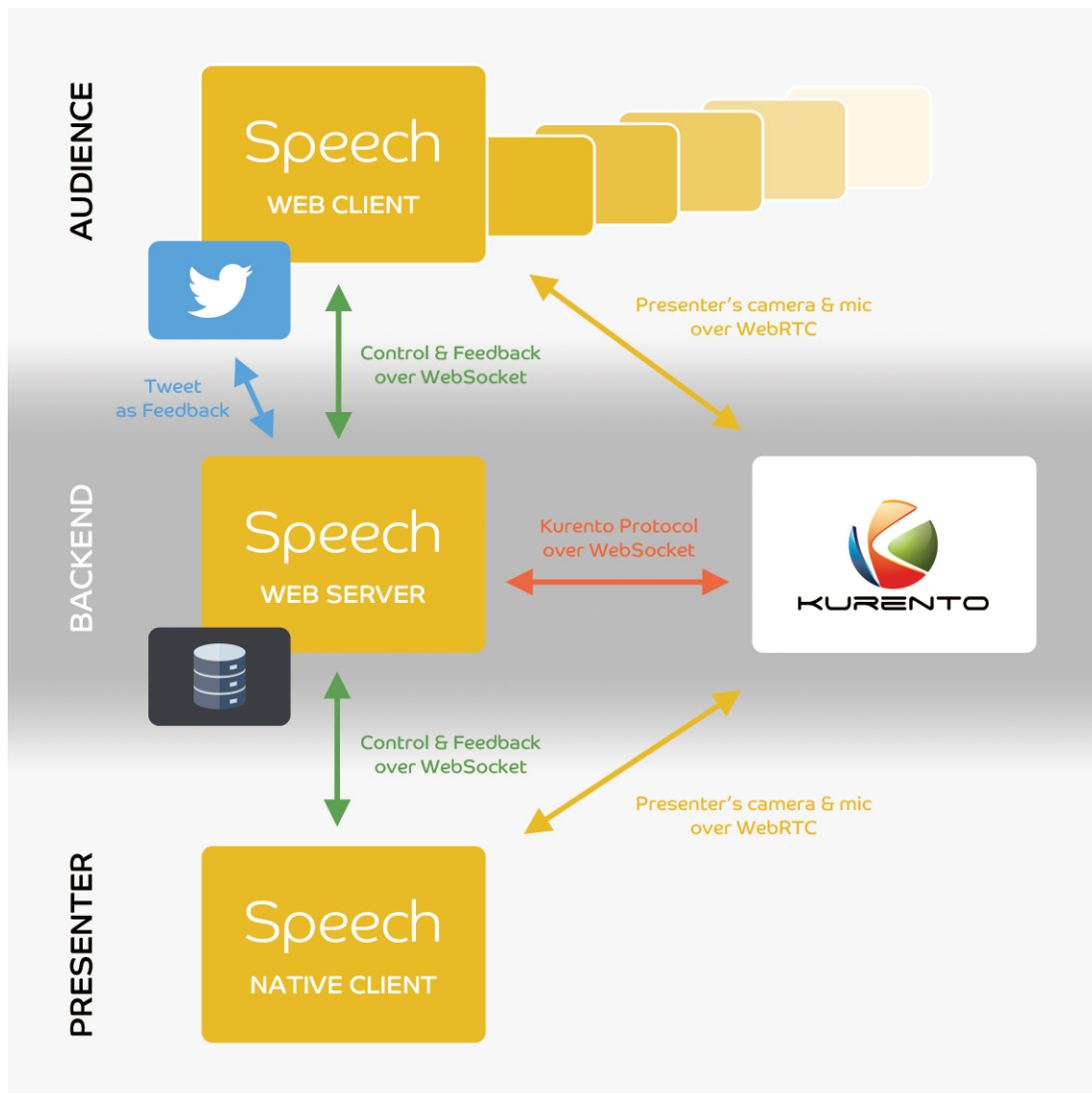


図 11 Meetup におけるプレゼンテーション記録システムの実装概要

第6章 実証実験

実際のプレゼンテーションにおいて本システムを用いた 2 つの実証実験について述べる。

6.1. 輪講会における実証実験

本節では、村井純研究室 Arch グループの有志で行っている「モダンオペレーティングシステム輪講会」において、本システムを用いたプレゼンテーション記録の実証実験を行った。

6.2. デベロッパーMeetup における実証実験

本節では、都内某所にて行われたエンジニア向け Meetup イベントに登壇し、本システムを用いたプレゼンテーション記録の実証実験を行った。

第7章 評価

本章では、本研究で設計・実装したプレゼンテーション記録システムの評価を行う。

7.1. 評価概要

- 映像配信時の参加者のスケーラビリティ
WebRTC を用いて映像記録・配信を行った際に安定して動作する同時参加者数の上限値。
- 対応するスライド共有サイトの数
プレゼンテーションを行う際に、スライド資料参照先として利用可能なスライド共有サイトの数。

7.2. 計測

7.2.1. 映像配信時の参加者のスケーラビリティ

15 秒間に一人ずつ参加者が増える環境を構築し、新規参加者が映像を受信できなくなったときの同時参加者数を計測した。これを 20 回繰り返し、その平均値を結果とした。計測の結果を図に示す。

7.2.2. 対応するスライド共有サイトの数

著名なスライド共有サイトならびに個人サーバー上に置いた異なるスライドデータの URL を発表者用アプリケーションに入力し、実際に配信が可能だったか否かを記録した。計測の結果を表に示す。

7.3. 考察

第8章 結論

本章では、結論として本研究の成果を明らかにするとともに、今後の課題について述べる。

8.1. まとめ

本研究では、Meetup におけるプレゼンテーションの記録を目的とした。その問題解決のために、Meetup におけるプレゼンテーション記録システムを設計・実装した。

Meetup におけるプレゼンテーションは、オンライン・オフラインの参加者が入り乱れる複雑な環境である。こうした状況を踏まえ、本研究ではプレゼンテーションの構成要素を配信しながら記録する WEB システムを実装して実証実験を行った。実証実験では、研究室で行われた輪講会と、外部のエンジニア Meetup において記録の実証実験を行った。

実証実験の結果より、本研究で構築したプレゼンテーション記録システムで Meetup におけるプレゼンテーションが正確に記録されることを確認し、本研究のアプローチが有効であることが確認された。

8.2. 今後の課題

実証実験の結果から、本研究のアプローチは Meetup におけるプレゼンテーションの記録に有効であることが明らかになった。しかし実証実験で構築した環境には幾つかの問題が見受けられた。以下に今後の課題を述べる。

- スライド資料が外部参照であるため、オフライン時に記録を再生することができない

謝辭

参考文献

- [1] “Meetup とは,” 19 12 2015. [オンライン]. Available: <http://help.meetup.com/customer/ja/portal/articles/637187-meetup%E3%81%A8%E3%81%AF>.
- [2] 高木義和, “情報文化 VIII. インターネットコミュニティの特性,” 28 5 2015. [オンライン]. Available: <http://www.nuis.ac.jp/~takagi/IB/2015/ic08.pdf>. [アクセス日: 20 12 2015].
- [3] T. NGO, “OFFICE OPEN XML OVERVIEW,” 23 10 2006. [オンライン]. Available: http://www.ecma-international.org/news/TC45_current_work/OpenXML%20White%20Paper.pdf. [アクセス日: 21 12 2015].
- [4] 総務省, “デジタルアーカイブの構築・連携のためのガイドライン,” 26 3 2012. [オンライン]. Available: http://www.soumu.go.jp/main_content/000153595.pdf. [アクセス日: 20 12 2015].
- [5] 市川裕康, “『ソーシャル&リアルがポイント。今求められる新しい出会い、学び、コミュニティの形～「ミートアップ」とは』 | ソーシャルビジネス最前線 | 現代ビジネス [講談社], ” 2 5 2011. [オンライン]. Available: <http://gendai.ismedia.jp/articles/-/3422>. [アクセス日: 21 12 2015].
- [6] 一. 三井, 誠. 内田, 晋. 白山, “SNS における情報伝播に対するコミュニティの影響,” 横断型基幹科学技術研究団体連合, 2005.