

置いてけぼりを防ぐミーティングシステムの提案

仙道航^{†1} 瀬川典久^{†2} 澤本潤^{†2} 杉野栄二^{†2}

大学や大学院の学生の教育研究活動において PowerPoint などを用いたゼミナールが広く行われている。ゼミナールでは、専門性の高いテーマについての活動報告を行うことから、実際に当該テーマについて研究活動を行っている発表者とそうでない聴講者との間に知識量の差が生じやすい。聴講者が発表に対し理解できないままゼミナールが進行していく状態はゼミナールの研究テーマをより良いものにするという目的を妨げる一つの要因になりうる。本稿では、発表資料に記載されているキーワードについて事前に調査を行い、発表時に聴講者へと提示する、聴講者がわからないキーワードを発表者に知らせて説明を促すことにより理解を促すためのシステムを提案する。

Proposal of meeting system to prevent left behind

WATARU SENDO^{†1} NORIHISA SEGAWA^{†2}
JUN SAWAMOTO^{†2} EIJI SUGINO^{†2}

Seminar using PowerPoint and in educational and research activities of students and graduate university is widely performed. In the seminar, from doing activity report of topics highly specialized, the difference between the amount of knowledge is likely to occur between the audience not the case with the announcement who is doing the research activities on the theme actually. State to continue to progress seminar audience remains incomprehensible to the announcement can be a factor of one to prevent the goal of how to improve the research theme of seminar. In this paper, we propose a system for promoting understanding by encouraging explanations as to tell the presenter conducted a survey in advance for the keywords listed in the press release is presented to the audience at press time and keywords that do not know the audience proposed.

1. はじめに

現在、大学や大学院の学生の教育研究活動においてゼミナール形式による講義が広く行われている。

大学のゼミナール形式は講座や研究室毎に指導教官と少人数の学生によって行われることが多い。ゼミナールにおいて学生は自身の研究テーマを設定し、それに対する研究活動の成果を報告する。この報告に対し、指導教官やその他の学生を交えて議論を行い学生の研究テーマをより良いものにしていく。

今日では、ゼミナールにおける報告で PowerPoint や Keynote といったプレゼンテーションツールが使用されることが多い。また、安価なラップトップ型のパソコンやプロジェクタの登場により、作成した発表資料を聴講者全体に提示し、発表者がその資料を基にプレゼンテーションを行うスタイルが一般的なものになった。

このように、PowerPoint などのプレゼンテーションツールは大学の教育研究活動におけるゼミナールを大きく進化させたと言える。

5 章で詳しく触れるが、近年では、プレゼンテーションに関する研究も広く行われるようになった。しかし、PowerPoint などのプレゼンテーションツールが登場してすでに 20 年以上が経過したが、発表者がスライドを基に聴講

者の前でプレゼンテーションを行うスタイルから何一つとして変化が見られない。

従来のプレゼンテーションのスタイルでは、実際に研究活動を行っている発表者とゼミナールでの報告を定期的に受けるだけの聴講者とは知識量に差が開きやすいと考えている。このように、発表者と聴講者との間に知識量の差があり、その状態がゼミナール毎に開いていくことは活動報告後に行う議論の質を下げる一つの要因になりうると思われる。

そこで、本研究では発表者が事前に準備した発表資料から発表を理解するために必要な専門用語を自動的に抽出することを行う。次に、抽出した専門用語の意味を自動的に検索する。発表時にはスクリーンを 2 枚用意し、1 枚に発表資料ともう 1 枚に発表資料のスライドに合わせて専門用語とその意味を提示する。また、聴講者の所有するコンピュータの Web ブラウザを通じて後者と同じ内容を提示する。聴講者は発表と提示された専門用語の意味を見てもわからない場合、提示された専門用語をクリックすることで発表者に分からない専門用語を知らせる。発表者は適切なタイミングで発表を中断し、専門用語の補足を行うことが可能になる。

これらにより、聴講者が置いてけぼりになったまま発表が進行することを防ぐ効果が期待できる。

本稿の構成は次の通りである。

2 章では、ゼミナールを成功させるために重要な要因に

^{†1} 岩手県立大学大学院
Iwate Prefectural University Graduate School
^{†2} 岩手県立大学
Iwate Prefectural University

について述べる。3章では、本研究における提案システムの概要について述べる。4章では、提案システムの構成と実装方法について述べる。5章では、関連研究を提示し、本研究の新規性について述べる。6章では、今後の予定と考察を行う。

2. ゼミナールとは

ゼミナールとは、学生の設定した研究テーマに対する活動の報告とこれに対する議論を行う場である。その目的は、学生の報告に対して議論を交わし研究テーマをより良いものにする事である。この目的を果たすためには、闇雲にゼミナールを行うのではなく秩序だったものにする必要があると考える。

そもそも、ゼミナールも一つの会議の形態である。

現在、会議をデザインし、より良いものにする目的の書籍は数多く出版されており、そのノウハウをシステム開発にも活かすことができると考えた。

山田らは著書[1]において会議を円滑に進めるためには表1に示す7つのステップが重要であると記している。

表1: 会議を円滑に進めるための7つのステップ

ステップ	役割
導入	メンバー全員で目的、議題、ゴールを共有する。
情報共有	議題の前提となる情報をメンバーで共有する。
議題や論点の設定	議論すべきことを決める。
発散	互いの意見やアイデアを出す。
収束	メンバーから出された意見やアイデアをまとめる。
意思決定	会議で決まったことをメンバーで確認する。
行動化	誰がいつまでに何をするかを明確にする。

表1におけるメンバーとはつまり、指導教官と学生を示している。また、議論すべきことは学生による活動報告や立てた計画に対する指摘やより良い案にはどのようなものがあるのかといった事柄である。行動化における誰がいつまでに何をするかとは、発表を行った学生が今後何をすべきかの他、必要であれば、指導教官やその他の学生が何をすべきかを明確にする。

従来の発表者が発表資料をゼミナールの期日までに準備し、発表資料を基にした発表を聴講者が聞くというスタイルではステップ2の「情報共有」において聴講者が置いてきぼりになることが懸念される。これは、事前に発表資料

に記載した事柄について研究活動をしている発表者とそうでない聴講者との間に知識量の差が生じるためである。ステップ2において置いてきぼりになる聴講者が生じるとその後のステップの進行に着いて行けず十分な議論が交わされないままゼミナールが終了してしまうという事態が発生してしまう。これは、当該ゼミナールだけでなく、次回以降のゼミナールにおいて開始直後からすでに置いてきぼりになっている聴講者が発生することにも繋がる。こうした状況はゼミナールの目的を妨げる要因となりうるため望ましくない。

本研究では、この7つのステップに基づき、ゼミナールを支援することを考えている。本稿では、その実現の足がかりとして、7つのステップのうちはじめの「導入」「情報共有」の2ステップを円滑に行うための支援システムの提案を行う。

3. システム概要

本稿では、ゼミナールにおける発表者と聴講者との間に存在する知識量の差を埋めるための新しいプレゼンテーション支援システムを提案する。

本システムでは、発表者が作成した発表資料の中から発表を理解するためのキーワードを抜き出し、キーワードが持つ意味について事前に調査を行う。ゼミナールにおいて当該資料を用いて発表を行う際、発表資料と共にキーワードに関する意味を聴講者に提示することで知識量の補填を促す。

また、聴講者が発表および資料と提示内容の3つを踏まえても専門用語が理解できなかった際に、聴講者が発表者にどの専門用語がわからなかったかを知らせるためのインタラクションを持たせた。このことによって、発表者が発表中に随時専門用語について、説明をすることが可能になり、事前に用意した発表資料を読み上げるだけの静的なゼミナールでなく、必要に応じて説明をする動的なゼミナールになると考えられる。

本システムをゼミナールに適用した際のプレゼンテーションスタイルは図1の通りである。図1では、発表者が聴講者の前に立ち、プロジェクタを通じて発表資料をスクリーンに映し出し、発表資料を基にした発表を机に座っている聴講者が聞いているという様子を示している。一般のゼミナールと異なる点は、プロジェクタが1台ではなく2台使用され、これに伴いスクリーンも2枚使用されている点である。本システムでは、スクリーンAには従来のゼミナール同様、発表資料を映し出す。スクリーンBにはスクリーンAに表示されている発表資料に対応するキーワード群とそれらの意味を映し出す。スクリーンBに映し出される内容の一例を図3に示す。

また、聴講者の所持しているノートPC、タブレット等に

も Web ブラウザを介してスクリーン B と同様の内容を映し出す。聴講者は、発表を聞いている時にわからない専門用語があれば、その専門用語をクリックする。この時、発表者には、どの専門用語がクリックされたかが、リアルタイムで伝えられる。発表者は、そのクリックされた情報を参考に、適切に発表を中断し、専門用語を説明することが可能になる。



図 1: システム概念図



図 2: スクリーン A の動作図



図 3: スクリーン B の動作図

スクリーン B と聴講者の所有するコンピュータに提示される内容はスクリーン A に映しだされている発表資料のスライド進行に伴って対応する専門用語とその意味に再描画される。

図 4~6 では、左側のスクリーン A の進行とともに右側のスクリーン B の内容が動的に変化していく様子を示す。

図 4 では専門用語が登場していないため、スクリーン B にも内容が表示されていない。

図 5, 6 ではスクリーン A に合わせてスクリーン B の内容が変化していることが分かる。



図 4: 1 枚目のスライドの様子



図 5: 2 枚目のスライドの様子

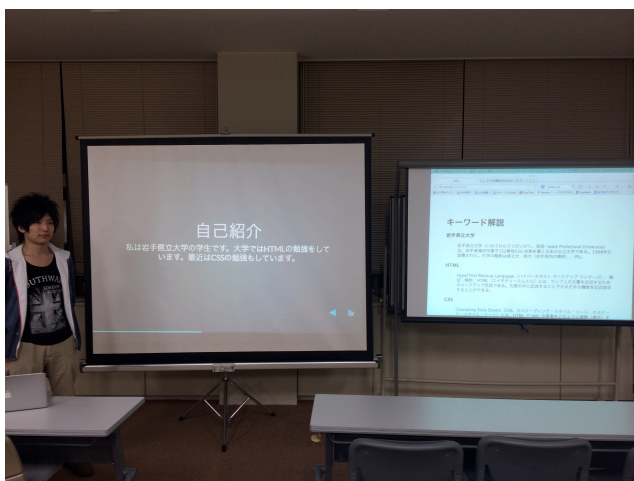


図 6: 3 枚目のスライドの様子

4. システム構成及び実装

本章では、提案システムに関して本稿執筆時点で構築したプロトタイプシステム α 版の構成と実装について述べる。

α 版では、発表資料に登場したキーワードを事前に手動で検索しデータベースへと登録する手法を取っている。また、発表者のコンピュータをサーバとしている。これに関しては後にシステム構成図とともに詳細について触れる。

本システムの構成を図 7 に示す。

本システムは図 1 におけるスクリーン A を映し出す Master とスクリーン B 及び聴講者へスクリーン B と同様の内容を映し出す Slave によって構成されている。

α 版では、Master は発表者のコンピュータにサーバサイド JavaScript として有名な Node.js[2] をインストールしている。また、Node.js のパッケージ管理マネージャである npm を利用し、Web アプリケーションフレームワークとして有名な Express、Web サーバとクライアントの双方向通信を実現する技術である WebSocket[3] を利用するライブラリで

ある socket.io、MySQL データベースへ接続するための mysql パッケージをインストールしている。これらを利用することで Slave 側では特別な環境を用意することなく各コンピュータにインストールされている Web ブラウザから図 1 におけるスクリーン B と同じ内容を閲覧することができる。

α 版の Master に当たるコンピュータの性能を表 2 に示す。なお、今回のシステム開発も同じコンピュータを用いて行った。

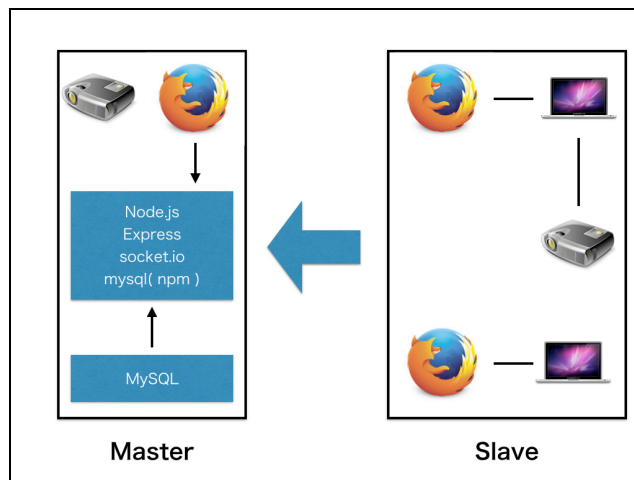


図 7: α 版システム構成図

表 2: コンピュータの性能及び環境

名称	概要
OS	Mac OS X 10.9
CPU	Intel Core i7 2.8GHz
RAM	DDR3 1600MHz 16GB
Server	Node.js
DB	MySQL 5.6.16

α 版のシステムフローについて図 8 を用いて説明する。また、図 8 中の各矢印が持つ意味を表 2 に示す。

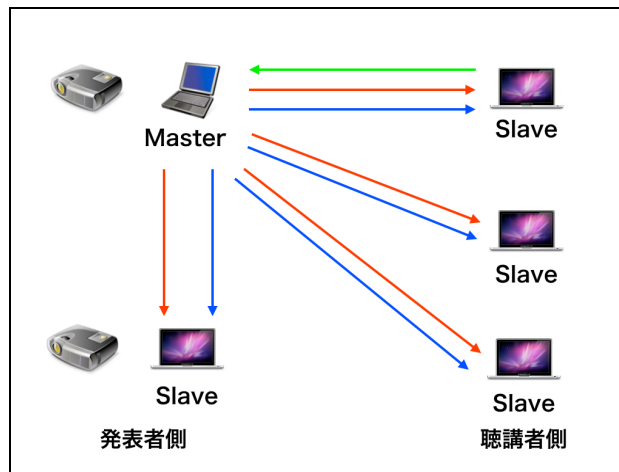


図 8: システムフロー

表 2: 各矢印の説明

矢印の色	処理
赤	発表資料 (Master) のスライドが切り替わった旨をブロードキャスト
緑	ある聴講者 (Slave) が分からなかったキーワードをブロードキャスト
青	緑の処理を受け発表者側の Slave と緑の聴講者以外の聴講者 (Slave) へキーワードをブロードキャスト

本システムでは、事前に発表資料からキーワードを抽出する。キーワードが持つ意味を調査しデータベースへと登録する。発表中は発表資料 (Master) のスライドが切り替わる度にスライドに含まれるキーワードと意味を発表者側の Slave と聴講者のコンピュータ (Slave) へと送信する。

また、聴講者のコンピュータ (Slave) から Web ブラウザを介して理解できなかったキーワードをクリックすると発表者側の Slave と他の聴講者のコンピュータ (Slave) で映しだされている当該キーワードがハイライトされ、誰が理解していないかを示すマークが表示される。

これらの処理を実現するために、本システムでは Node.js と socket.io を合わせたサーバと Web ブラウザ間の双方向通信を用いている。

Socket.io では、WebSocket と呼ばれる双方向通信技術の仕組みを利用している。これは、従来の HTTP Polling といった仕組みと比べて効率的に双方向通信を実現することができる。

また、今回は Web ブラウザを利用したシステム実装を行っていることからプレゼンテーションツールとして Reveal.js[4]を使用した。Reveal.js は HTML など Web に関するマークアップ言語やプログラミング言語を用いて簡単に高機能なプレゼンテーション資料を作成できるライブラリである。また、vll.io[5]と呼ばれるプログラミングが苦手なユーザでも Web ブラウザ上で PowerPoint を使用する感覚で同等のプレゼンテーション資料を作成できるツールも用意されている。特に、エンジニアに人気があり、企業の説明会などで使用されることも少なくない。

本システムでは、Node.js と socket.io、Reveal.js がイベントハンドラを用いて簡単にイベントに対する処理を行うことができるため組み合わせて使用した。

発表者 (Master) が発表資料の切り替えを行うと、スライドが切り替わった後、スライド番号に一致するキーワードを検索するクエリがデータベースへと発行される。次に、

全て Slave へキーワードとそれが持つ意味を JSON 形式でブロードキャストする。受け取った Slave は jQuery を用いた DOM 操作を行い、画面内容を新しいキーワードと意味で再描画する。図 8 においてこれらの処理は赤の矢印で示している。

次に、図 8 において右側の最も上に配置されている聴講者 (Slave) が理解できなかったキーワードがあったことを想定する。この時、聴講者 (Slave) は理解できなかったキーワードをクリックする。キーワードをクリックされるとクリックイベントが発行され、当該キーワードがサーバ (Master) へと送信される。図 8 においてこれらの処理は緑の矢印で示している。

受け取ったサーバ (Master) はクリックイベントを発行した聴講者 (Slave) を含め、全ての聴講者 (Slave) と発表者側の Slave へとその旨をブロードキャストする。これを受け取った全ての Slave は jQuery を用いた DOM 操作によりキーワードをハイライトし、誰が理解できなかったかを示すクエスチョンマークを描画する。図 8 において、これらの処理は青の矢印で示している。

α 版では、これらの双方向通信と処理を発表中に継続して行っており、発表資料のスライドが進むだけでなく戻るといった際にも動作するようになっている。

現在は、発表者と聴講者が同じ LAN に所属していることを前提としている。

以上が本稿執筆時点で開発が完了しているシステムとなる。次章では本研究の新規性などについて関連研究とともに述べる。

5. 関連研究

近年、ゼミナールに限らず広くは研究学会などにおいてもプレゼンテーションは行われており、そのあり方や相互理解を促進するための研究も行われている。

本研究のように、これまでのプレゼンテーションの形式をシステム導入によって変更することによって新しい知見や効果が現れている事例が報告されている。

前田らの MAEDE[6]では、発表資料が中心であった従来のプレゼンテーションを発表者が発表資料の前に立ち、身振り手振りを交えて行う発表者中心のプレゼンテーションスタイルへと変更した。MAEDE では、従来は発表者に重なってしまい見えなくなる発表資料中の文章を発表者の輪郭を深度カメラによって取得し、重ならないよう退避して表示する手法を取っている。MAEDE を使用することで従来のプレゼンテーションよりも聴講者を引き込む効果があることが報告されており、システム導入によって良い効果をもたらす場合は、スタイルの変更を視野に入れて検討していくべきであると考えた。

また、相互理解のための研究としては黒河らの研究[7]

や山野らの MultiCollab[8]が報告されている。

黒河らが提案したシステムは、発表資料に対し聴講者が Web ブラウザを介してコメントを行い、その内容がコメントログや発表資料に重畳表示されることで発表者に説明を促すことや相互理解を促す事を期待するものである。本研究では視認性が悪くなることから発表資料とキーワードにスクリーンを独立する方針を取っている点やシステムによる相互理解のための自動的な補助を行っている点などで異なる。

山野らの MultiCollab では、タブレットデバイスを中心とした対話を想定し、音声認識や手書き文字認識を用いて会話中のキーワードを抽出し、画像などを自動的に検索し提示することでお互いのイメージを補完することによる相互理解を促している。本研究とは想定している利用シーンが異なるほか発表資料といった事前にプレゼンテーションに関わる情報を活用する点で異なる。今後、表 1 で示した発散や収束などのステップで新たに議論内容という情報群を活用することが考えられる。これらのステップでは積極的に音声認識などの技術を活用して組み合わせていくことを検討している。

6. 今後の予定と考察

今後の予定について、まず、本稿の執筆と並行して開発している β 版のシステムの構成と実装について述べる。

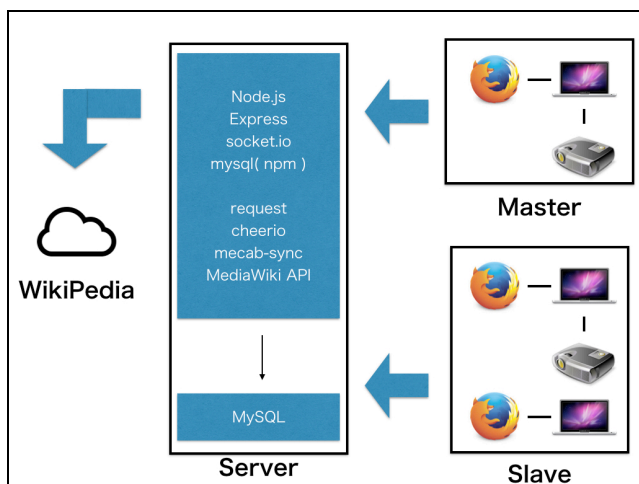


図 9: β 版システム構成図

図 9 に β 版のシステム構成を示す。

β 版では、α 版で発表資料から手動でキーワードを抽出し、検索した意味をデータベースに登録していた部分の自動化を図っている。また、α 版では発表者のコンピュータがサーバと Master の役割を兼任していたが、それぞれ独立させた。これに伴い、発表者は行う操作はサーバへと発表資料をアップロードするだけとなった。

β 版では、発表資料に対して Node.js のパッケージであ

る request と cheerio を用いたスクレイピングを行う。

Request とは、スクレイピング対象の Web ページを URL 指定して HTML 形式で取得するためのパッケージである。また、cheerio とは jQuery ライクな HTML スクレイピングを行うためのパッケージである。これらを用いて Reveal.js によって作成されている発表資料をスクレイピングし、発表資料中の文章を全て抽出する。

次に、抽出した文章中からキーワードとなりうるものを Node.js のパッケージである mecab-async を用いて抽出する。mecab-async とは形態素解析エンジンとして広く知られている MeCab[9]を Node.js から呼び出せるようにしたパッケージである。これを用いて一般名詞のみを抽出する。

最後に、抽出した一般名詞を検索ワードとして Wikipedia の概要にあたる文章を取得し、MySQL へとキーワードと組んで登録する。Wikipedia は MediaWiki[10]というソフトウェアをベースに作られているため、MediaWiki API が使用可能である。MediaWiki API を用いて取得した内容と同様のスクレイピングを行いデータベースへと登録する処理を行っている。

次に、本システムについての実験について述べる。今後の実験スケジュールとしては、本稿で述べた β 版が完成次第、岩手県立大学基盤ソフトウェア学講座[11]で行われているゼミナールに本システムを適用していくことを考えている。

現在、岩手県立大学基盤ソフトウェア学講座では、すでに研究テーマが決定しておりテーマに対する専門性の高い活動報告が行われる大学院生ゼミナールが開かれている。また、今後具体的な研究テーマを決定すべく興味のある分野を広くサーベイし、その成果を報告する学部生ゼミナールが開かれている。同じゼミナールではあるが、研究活動としてのフェーズが異なり、異なる性質を持った 2 つのゼミナールにシステムを適用し実験を行う。

実験では、通常通りゼミナールを行った場合とシステムを適用した場合とで発表者及び聴講者の理解が互いにどれほど深まったかをユーザー評価型のアンケートによって調査することを検討している。

なお、実際にゼミナールに適用していくにあたり、現在のシステムでは、従来の発表資料作成手順とは異なる手順を取らなければならない。そのため、従来のプレゼンテーションツールで作った資料をエクスポートする方法や従来のプレゼンテーションツールから大きな不便のない作成方法を提示することを検討している。

最後に、本研究が目指す最終的なミーティングシステムについて述べる。

本研究での大きな前提として、従来の PowerPoint を用いたプレゼンテーションスタイルにより行われるゼミナールから、より相互理解が深まり、有意義な議論が交わされるゼミナールへと進化することを目指している。このため、

本稿で提案したシステムが単独で使われるのではなく、表1に示した7つのステップをトータルに支援するミーティングシステムの開発を検討している。このミーティングシステムそのものがフレームワークとなり、利用をしていく中で表1の7つのステップに沿うような形で議論が進められていくことを想定している。

また、これらを実現するために単に表1の7つのステップをなぞるのではなく、システムを導入したことによる明確なメリットが存在しなければ普及には至らないと考えている。従来のゼミナールやプレゼンテーションスタイルでは実現が困難な要素としてその場に存在しない専門家の知識に頼ることや集合したメンバーの知識領域や発想の外にある事柄について気づきを得るといった点があげられる。

この点に関して、今回提案したシステムでは、キーワードに対してその場に存在する発表者や聴講者の知識だけに頼るのではなくインターネット上の集合知による補完を行った点が足がかりとしている。

今後、ゼミナールを支援するトータルシステムを開発していくにあたって、本システムの実験等で得たノウハウやインターネット上の集合知、研究活動における論文サーベイなどを支援する研究など様々な情報や知識を組み合わせた知識マップをどう構築し、システムへと反映していくかが現在の大きな課題となっている。

参考文献

- 1) 山田豊, 横館暁郎: 会議の上手なやり方が面白いほどわかる本, 中京出版(2007)
- 2) Node.js: <http://nodejs.jp/>
- 3) 松井暢之, リアルタイム Web を構築しやすくする Socket.io とは, CodeZine(2013) <http://codezine.jp/article/detail/7075>
- 4) reveal.js : <http://lab.hakim.se/reveal-js/>
- 5) rvl.io : <http://slides.com/>
- 6) 前田晴己, 黒澤裕也, 栗原一貴, 宮下芳明: MAEDE スクリーン前でのプレゼンテーションスタイル, WISS2011 (2011)
- 7) 黒河優介, 藤枝崇史, 福本佳史, 関戸亮介, 服部隆史, 荻野達也: Web を用いた参加型プレゼンテーション, WISS2008(2008)
- 8) 山野真吾, 佐藤俊樹, 小池英樹: MultiCollab 相互理解のための議論補助システム, WISS2012 (2012)
- 9) MeCab : <http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html>
- 10) MediaWiki : <http://www.mediawiki.org/wiki/MediaWiki/ja>
- 11) 岩手県立大学 ソフトウェア情報学部 基盤ソフトウェア学講座: <http://www.dais.soft.iwate-pu.ac.jp/>