電子教科書におけるサーバ連携機能の開発

A1078307　岩崎　広平

指導教員　田村　恭久

1. はじめに

電子教科書を導入することには，印刷費用の削減，動画や3Dなど書籍では表現できないコンテンツの表示，障碍のある方を含めた広い学習者への対応などの利点がある．また，生徒の情報をオンラインで管理，蓄積，整理することや，協調学習への利用も期待できる．しかし，現状では電子教科書に要求される機能と，それらを実現するための仕様が定まっていない．田村研究室では，この問題を解決するために，電子教科書に要求される機能項目を73項目に整理した[1]．この73項目は，Jung[2]やKim[3]らの先行研究をベースに，DiTT[4]が提案する標準機能案やISO SC36が策定した電子教科書の要求機能に関するアンケートの要素を盛り込み，さらにアクセシビリティを考慮して作成したものである（図1）．

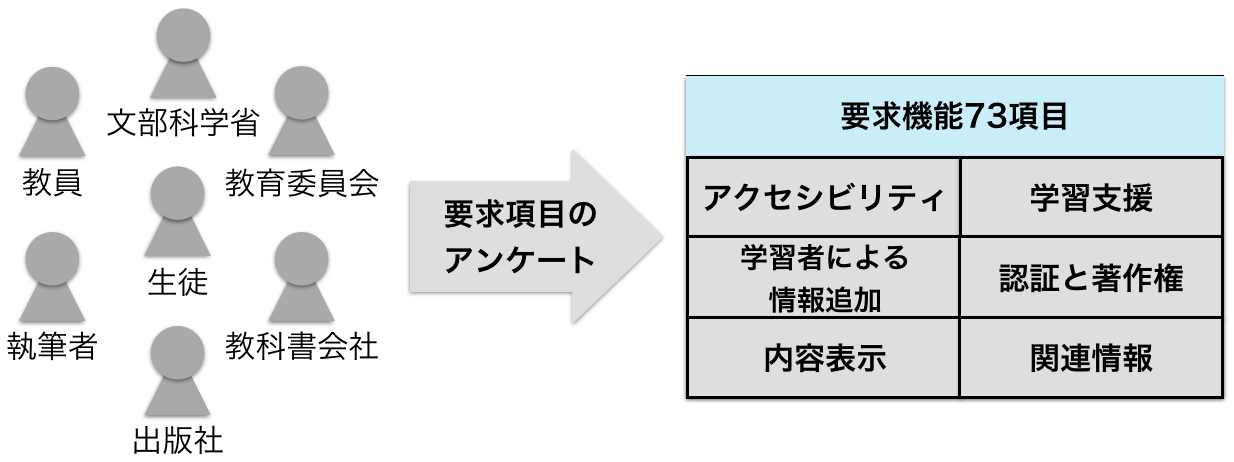
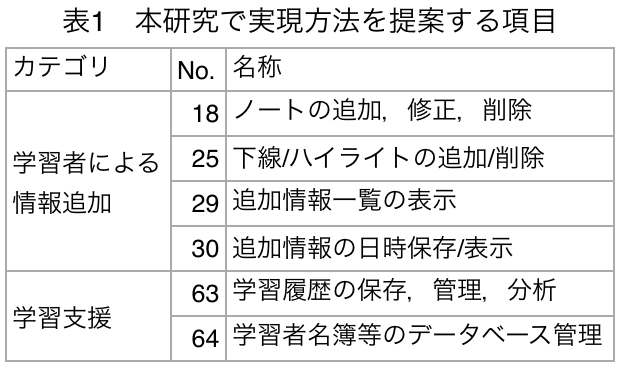


図1　要求される機能項目の整理

さらにこれら73項目の要求項目の実現方法について，既存規格の適用可能性を検討することで調査した．本研究では表1の項目を岩崎が担当し，これらの機能を開発した上で，実現方法として提案し，公開した．



1. サーバ連携機能の開発

　情報を追加するための機能をクライアント（電子教科書リーダー）に実装した．追加された情報はクライアントとサーバそれぞれで保持し，他人の追加した情報についても同期できるように機能を開発した．図2は，本研究の成果物である．左側には教科書の本文があり，情報を追加する際には本文を邪魔しないよう，右側のサイドバーから情報を入力し，サーバへ送信することができる．

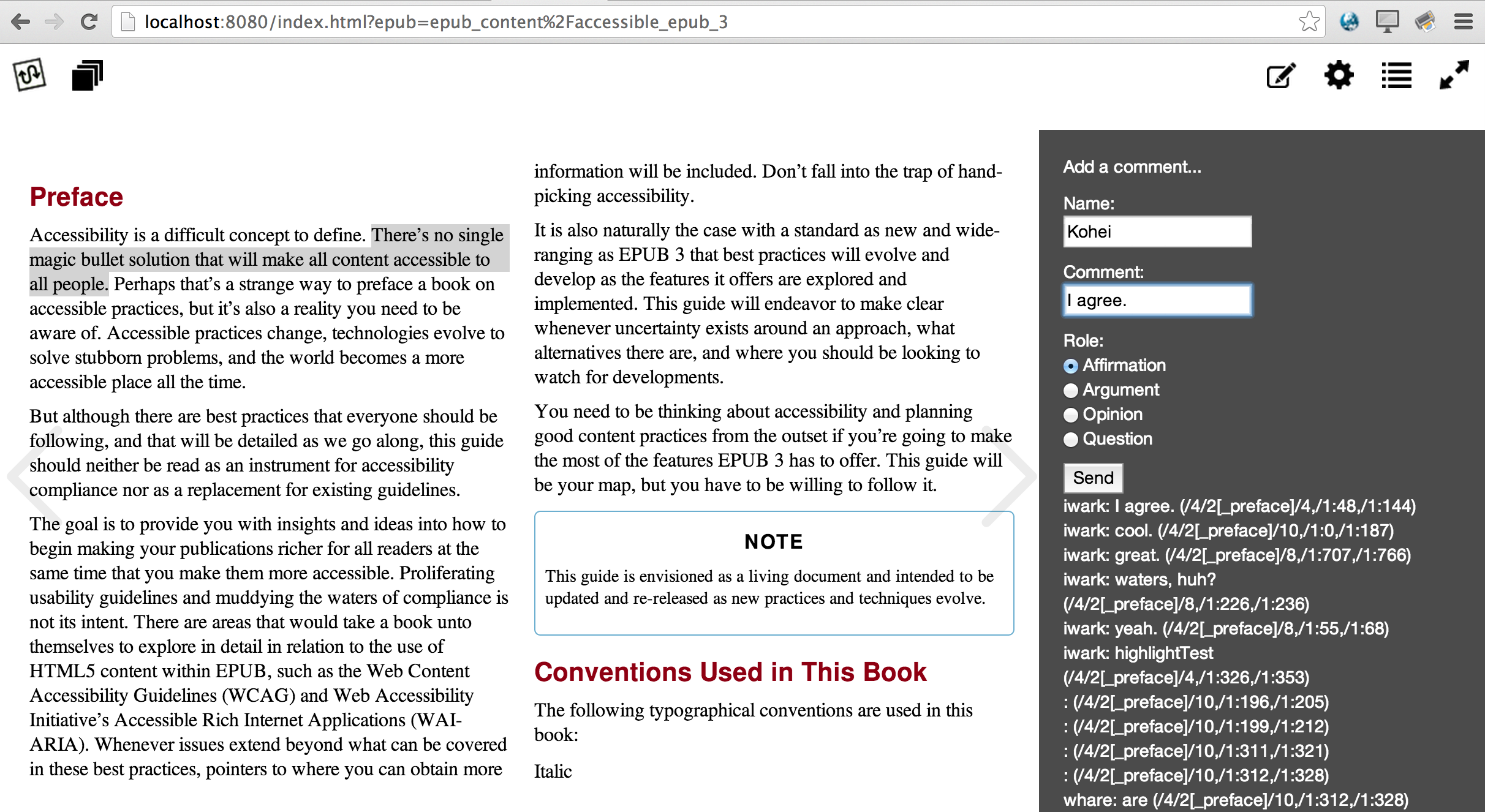


図2　情報追加機能を加えた電子教科書リーダー

次に，リーダーとサーバ，それぞれどのように実装したのかを述べる．

2.1.　クライアントの機能の開発

電子教科書のコンテンツとして表示するファイルは，米国の電子出版業界の標準化団体であるInternational Digital Publishing Forum (IDPF)が策定したファイル規格であるEPUBを用いた．EPUBはコンテンツ開発が容易であることや，インタラクティブな動作やオンラインでの通信を可能にするjavascriptを扱えるなどのメリットがある．また，EPUBはiOSやAndroidを搭載した機種や，楽天kobo Touchなど多くのデバイスで動作する．本研究では，IDPFのオープンソースEPUBビューアであり，Webブラウザ上でEPUBを閲覧することができるreadium-js-viewer[5]を電子教科書のリーダーとして用い，サーバ連携と，それによる協調学習に必要な機能を追加した．追加した機能は以下の３点である．

1. 教科書内への情報の追加
2. 追加した情報のサーバへ送信，他のユーザーへの共有
3. 追加された情報に対しての，情報の追加

３点とも，開発にはjavascriptを用いた．これらの機能を実現しているサービスにHypothes.is[6]があるが，本研究ではHypothes.isには無い，コメントの役割を分ける機能を追加した．ユーザーは，コメントをする際に「賛成」「反対」「意見」「疑問」の4つの中から役割を選択する．このように役割を分けることによって，賛成側の意見と反対側の意見を分けて整理できるなどのメリットがある．

2.2. サーバの開発

膨大な追加情報の保存をし，リアルタイムに生徒へ配信するために，サーバは，高速であり，スケーラブルであることが求められる．従って本研究ではサーバの構築にNodeを用いた．また，同様の理由によりデータベースにはMongoDBを用いた．APIは，Advanced Distributed Learning (ADL)社が主導して制定を進めている規格であるTinCanAPI[7]を用いて設計した．ただし，本研究に必要の無い仕様や機能については，実装していない．また，一部本研究に合わせて改変している箇所がある．TinCanAPIでは基本的にユーザの学習情報1つ1つをStatementとして, 1つのStatementをActor, Verb, Object, Resultに分けて保存する．学習内容だけでなく，学習完了などのステートメントを様々なプラットフォームから送信することができるため，Learning Record Store(LRS)として用いることができる．本研究では学習履歴の蓄積だけでなく，生徒同士での協調学習(CSCL: Computer Supported Collaborative Learning)としての役割も果たせるサーバの開発が目的であるが，これもTinCanAPIの仕様でカバーできる．例えば，生徒が賛成の立場で教科書にコメントをした場合，図3のように情報がサーバに送信される．この例では対象が教科書であるが，対象を別のStatementにすることによって他のコメントに対してのコメントも可能になる．

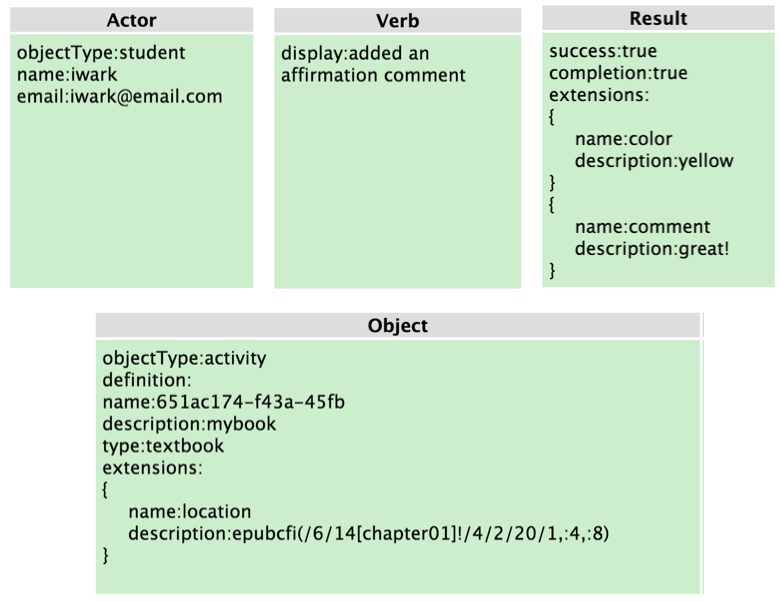


図3 　サーバに送信される情報の例

1. 結果と考察

開発した成果物は，2014年1月にニューヨークで行われたOpen Book Hack Weekend[8]というイベントで発表した．Readium Foundation, O’Reilly Media, Perseus Books, Hypothes.is, Googlといった電子教科書に関する影響力の強い数多くの関係者の前で発表することができた．また，成果物はインターネット上でオープンソースで公開しており[9]，本研究での成果の再現や，誰でも改良することが可能である．

1. 考察

電子教科書の標準化を巡る動きは，ここ数か月の間で急速に進んでいる．IDPFが2013年10月より電子教科書の標準化に向けてEDUPUBを策定している他，国内でも文部科学省が電子教科書のサンプル作成に向けて2013年12月から意見交換が開始された．こうした電子教科書の標準化に向けた活動において，電子教科書に要求される機能や議論されるべき課題の提案が求められている．国内外のこうした活動に参加して，未だ明らかになっていない問題について積極的に意見交換していくことが，今後の研究課題となる．

**参考文献**

1. 田村研究室，73\_required\_functions.xlsx，https://github.com/Iwark/thesis/blob/master/73\_required\_functions.xlsx (2014/1/16)
2. Jung, S-M. et al., “Leading Future Education: Development of Digital Textbooks in Korea”, Proc. 12th UNESCO-APEID International Conference Quality Innovations for Teaching and Learning. (2009)
3. Kim, M., Yoo, K-H, Park, C. ve Yoo, J-S., “Development of a digital textbook standard format based on XML”, Computer Science and Information Technology AST/UCMA/ISA/ACN 2010 Conferences, pp. 363–377. (2010)
4. デジタル教科書教材協議会，DiTT第一次提言書改訂版 (2011)

http://ditt.jp/office/ditt\_teigen\_1kai.pdf

1. Readium.org, readium-js-viewer, https://github.com/readium/readium-js-viewer (2014/1/16)
2. Hypothes.is, http://hypothes.is/ (2014/1/18)
3. TinCanAPI, EXPERIENCE API VERSION 1.0.0, http://www.adlnet.gov/wp-content/uploads/2013/05/20130521\_xAPI\_v1.0.0-FINAL-correx.pdf (2014/1/18)
4. Open Book Hack Weekend at The New York Public Library, https://www.eventbrite.com/e/open-book-hack-weekend-at-the-new-york-public-library-registration-9532284319 (2014/1/18)
5. 岩崎 広平, thesis, https://github.com/iwark/thesis (2014/1/18)