

招待論文

全学的学習支援システムの連携と それに基づく安否確認システムの開発

中野 裕司^{1,2,3,4,5,a)}

受付日 2019年3月11日, 採録日 2019年3月11日

概要: 2000 年前後から大学の各種手続きや学習環境のオンライン化が始まり, その数が増えるにつれて, 各々のシステムでのログイン, 同じような情報の多重入力, システムごとの情報の差異等が問題になってきた。そのため, シングルサインオン, 学務情報データの共有, 大学ポータルへの導入, 標準化等により, 学務情報システム, 学習管理システム, シラバス, 時間割, 学修成果可視化等様々な学習支援システムの全学的な連携を行ってきた。また, その環境に適合する Web アプリケーションの大まかな仕様の設定と雛形の提供を行い, 新たなシステム導入時の連携を推進した。本稿では, これらの取り組みを紹介するとともに, 2016 年 4 月におきた熊本地方を震源とする熊本地震に際し, この共通基盤上で緊急対応した安否確認について報告する。

キーワード: 学習支援システム, シングルサインオン, データ連携, IMS, 安否確認

University-wide System Collaboration for Learning Support and Urgent Development of Safety Confirmation System based on the Environment

NAKANO HIROSHI^{1,2,3,4,5,a)}

Received: March 11, 2019, Accepted: March 11, 2019

Abstract: As the university's administrative procedures and learning environments became online from about 2000, there had been problems with troublesome login to each system, cumbersome input of similar information, differences in recorded information of each system, etc. For this reason, collaboration of various learning support systems such as student information system, learning management system, syllabus, timetable, visualization of academic outcome, etc. have been proceeded by utilizing single sign-on, sharing of student information data, university portal, standardization, etc. In order to prompt collaboration when introducing new systems, rough specifications and templates for web applications that conform to the environment have been also set up. In this paper, we introduce these efforts and report on the safety confirmation system that was urgently handled on this common platform in the Kumamoto earthquake that occurred at the Kumamoto district in April 2016.

Keywords: learning support system, single sign-on, data synchronization, IMS, safety confirmation

¹ 熊本大学総合情報統括センター
Center for Management of Information Technologies,
Kumamoto University, Kumamoto 860-8555, Japan
² 熊本大学大学院社会文化科学教育部教授システム学専攻
Graduate School of Instructional Systems, Kumamoto Uni-
versity, Kumamoto 860-8555, Japan
³ 熊本大学教授システム学研究センター
Research Center for Instructional Systems, Kumamoto Uni-
versity, Kumamoto 860-8555, Japan
⁴ 熊本大学大学院自然科学教育部情報電気工学専攻
Department of Computer Science and Electrical Engineer-
ing, Graduate School of Science and Technology, Kumamoto
University, Kumamoto 860-8555, Japan
⁵ 熊本大学大学情報分析室
Institutional Research section, Kumamoto University,
Kumamoto 860-8555, Japan
a) nakano@cc.kumamoto-u.ac.jp

1. はじめに

大学における Web ベースの各種システムは 2000 年前後から導入がはじまった。たとえば, 熊本大学の場合, 学務情報システム SOSEKI を他大学に先駆けて 1999 年から運用を開始し [1], 梶田氏によって早期に日本語化された学習管理システム (Learning Management System, LMS) である WebCT [2] を 2004 年に全学導入した [3]。この時点ではまだシステム数も少なく, 各々にユーザ ID とパスワードを入力してログインしても, それほど煩雑ではなかったであろうが, 同様のシステムが増加することで, いちいちログインすることの煩雑さ, 同じような情報の多重入力,

2.2.1 LMS (WebCT, Moodle)

まず、このようなデータ連携を始めたきっかけであるが、2004年にWebCTを導入した際に、普及促進を図るために全科目、全受講者、全担当教員を学務情報システムと同期させてWebCTに登録する仕組みを構築したことである[13]。リアルタイムの同期はできなかったが、毎日深夜にバッチ処理で同期をさせることができた。WebCTが標準規格であるIMS Enterpriseに対応しており、規格に沿ったXMLファイルに科目リストや受講名簿を登録してWebCTにわたせば、登録科目や受講者のリストが更新される機能を有しており、それを利用することにした。残念ながら、SOSEKIはIMS Enterpriseには対応していなかったが、CSV出力が可能であったため、中間サーバにIMS Enterprise形式のデータへの変換プログラムを置き、同期させることに成功した[13]。

このIMS Enterpriseによるデータ連携は、その後も役に立ち、2013年度から始まったWebCTからMoodleへのLMSの移行(2014年度までは並列運用)においても、MoodleもIMS Enterpriseに対応していたため、登録科目や登録ユーザに関しては大きな混乱もなく、並列運用、移行が行えた。

このように、標準規格に対応した場合の利点も大きいですが、実際には他のシステムの多くは、独自のCSVやSQL等で連携を行っている。これは、実装の難易度やコストパフォーマンスを考えた結果であり、現実には難しいところもあると考えている。

2.2.2 時間割システム

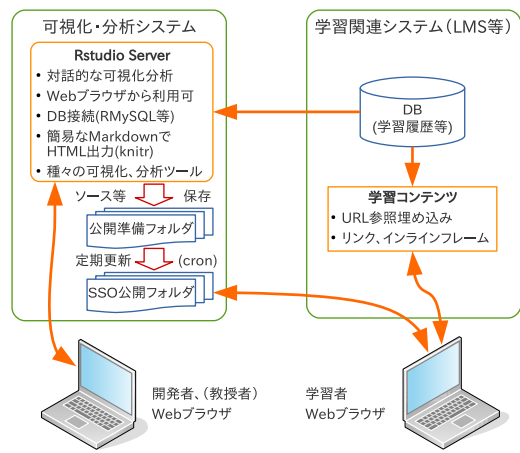
時間割システムは、大学ポータルタブとして表示され、そのユーザに自分の受講(担当)している科目の時間割が表示され、各々の科目名からLMSの該当科目ページや学務情報システム、シラバスへのリンクを設置したもので、授業ページへの入口的存在である。本Webアプリケーションは、2015年より、3章のWebアプリケーションの雛形に沿ったもの書き直し、基本的にWebAPI構成のシングルページアプリケーション(SPA)である[14]。

2.2.3 学修成果可視化システム(ASO)

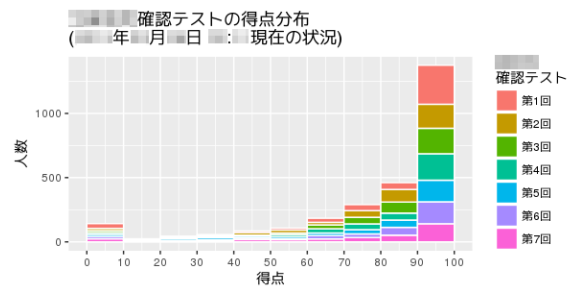
学修成果可視化システム(ASOは学内で定めた愛称)に関しては、元々全学eポートフォリオとして開発を進めてきたもので、最初はオープンソースLMS Sakai上に構築していたが、Sakaiの更新や多くのフレームワークに縛られ、その設計方針を見直したもので[15]、3章の雛形の発想の元になったものである。LMSに依存せず学修成果を蓄積するため、今の所2、3カ月に1度の割合であるが、Moodle中に蓄積された課題の説明と提出課題およびクイズの問題と提出された解答をリンクではなくその実体として複写、同期している。

2.2.4 シラバスシステム

シラバスシステム[16]はモックアップを用いて設計し、



(a)



(b)

図 3 (a) 即時的分析・共有の仕組み [18] と (b) 出力例
Fig. 3 (a) Realtime analysis and sharing method [18], and (b) an output example.

雛形とともに外部に開発を依頼した例であり、大学の定める7つの学修成果の割合等も含めて入力するため、ASOとの連携も必要になる。Moodleとの連携は、シラバス上で授業各回の資料等の設定が可能で、Moodleの機能拡張でinline frameによって対応している。

2.2.5 即時的分析・共有システム

即時的分析・共有システム[18]は、開発者レベルを対象にしているが、Webブラウザ上の統合開発環境(IDE, Rstudio Server)を利用してデータ処理のプログラムを作成することで、熊本大学IDで認証されたユーザのみに、一定時間ごとに更新された統計情報や分析結果を提示することが可能である。図3(a)にその仕組みの概略を示すが、Rの機能が使えるため、LMSでは難しいような可視化も可能で、例として、図3(b)に、複数回のテストの得点分布を積み上げグラフとして10分ごとに表示したものを示す。

この仕組みは、4章に示す熊本地震への対応にも活用したので、もう少し説明を加える。図3(a)において、開発者または教授者がWebブラウザ上のIDEで、まず、データベース等を参照しながら、対話的に試行錯誤しながら可視化、解析のプログラミングを作成する。基本的にR言語でできることは可能で、たとえば、RMySQLでMySQLデータベースに接続したり、ggplot2で可視化したり、様々な処理が可能である。また、knitrを用いれば、Rの結果を

html ドキュメントとして出力することもできる。次に、R のソースファイルを決められたフォルダ「公開準備フォルダ」に保存すると、システム側 (cron) で一定時間ごとにそのコードが実行され、SSO で認証されたユーザのみアクセス可能なフォルダ「SSO 公開フォルダ」に実行結果が保存される。たとえば、SSO 公開フォルダ内の html ファイルへリンクを貼ったり、グラフをその URL を用いて LMS のコンテンツ内に img タグで配置することも可能である。

2.2.6 研究者総覧, 評価システム

研究者総覧 [17] と評価システムは、まだ実運用には達していないが、外部システム (researchmap) の WebAPI を利用した例で、今後 WebAPI によるシステム間連携が進むものと考えている。

2.2.7 LRS (Learning Record Store)

LRS (Learning Record Store) は、Learning Analytics で学習履歴を蓄積する場所のことを示し、この場合は、Moodle の Caliper log store プラグインとオープンソースの LRS である Open LRW を WebAPI で接続し、Moodle 上の学習履歴を Caliper 標準形式で集積している [19]。

2.2.8 安否確認システム

熊本地震への緊急対応で開発を開始し、後に整備したもので、4 章でその詳細を示す。

3. Web アプリケーションの雛形

2.2 節で示したように、認証やデータをシステム間で連携する仕組みがある程度固まってきたため、新たに Web ベースのシステムを導入する際に役立つよう、Web アプリケーションの雛形を作成した [20]。

元々のきっかけは、2.2.3 項で示した ASO の開発で、それまでのように多くのフレームワーク等に縛られない最低限の仕様を決めたことであり、それはおよそ以下のようなものである。

- 基本的に、Web ブラウザ上で動作するシングルページアプリケーション (SPA) とし、クライアントプログラムは、HTML, CSS, JavaScript で構成すること。
- サーバとのデータのやりとりは、REST 形式 (データ形式は JSON/JSONP) の Web API で行うこと。
- 認証が必要な場合は、サーバが CAS に対応すること。
- ライブラリ等は、オープンソースのものを使うこと。
- クライアントプログラムは、以下の JavaScript ライブラリの利用を推奨する。
 - jQuery : WebAPI 等
 - jQuery Mobile : モバイル対応
 - jQuery i18n : 国際化

といった、かなり緩やかな仕様であり、すべての可視化はクライアントである Web ブラウザで行い、必要なデータは Web API でサーバからそのつどとってくる方法である。メリットは、WebAPI の仕様さえ決まっていれば、サーバの

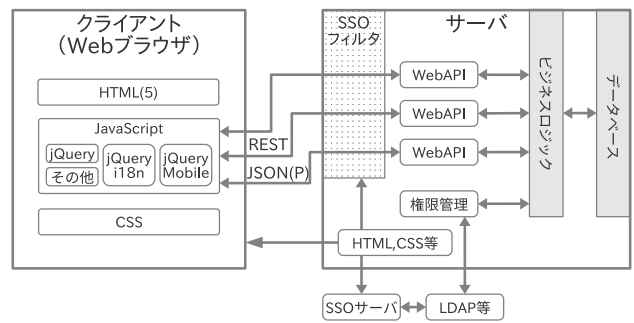


図 4 Web アプリケーションの雛形の構造
Fig. 4 Model structure of Web application.

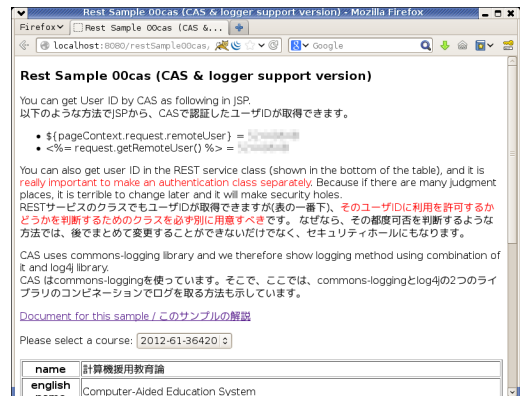


図 5 Web アプリケーションの雛形 (サンプルプログラム)
Fig. 5 Web application template (sample program).

実装方法はあまり問わないことと、ブラウザ側の HTML, CSS, JavaScript によるプログラム開発だけでほとんどの実装が行え修正が簡単にできることと、JSONP 形式を用いるとクロスサイトスクリプティング等のセキュリティ制限を越えて他の Web サイトからも利用できること等があると考える。また、jQuery Mobile によるある程度のモバイルデバイスへの対応と jQuery i18n による国際化も、クライアントの範囲内で対応できる。

このような仕様を考えたとうえで、2.2.4 項に示したシラバスシステムのようにモックアップで設計後、外部に開発依頼を行ったり、2.2.3 項の ASO のように、途中まで開発し、部分的に外部に開発依頼を行ったりした。そのような際の利便性も考え、仕様を具体化した雛形の作成を行った。図 4 にその構造を、図 5 に雛形の実例を示す。実際には、図 5 を含め、WebAPI (REST) サンプル、国際化サンプル、WebAPI+SSO (CAS) サンプル、WebAPI+MySQL (jdbc) サンプルの 4 つを作成した。実際に図 2 における、researchmap と Moodle 以外の WebAPI と記されたシステムは、ほぼこの仕様と雛形に沿っている。

4. 熊本地震への緊急対応

以下の報告では、熊本地震への緊急対応の中で、認証やデータのシステム間連携、Web アプリケーションの仕様や雛形に関する話題を中心に取り上げるが、タイムラインや

学生教職員のコメント等の詳細は [21] を参照されたい。

4.1 準備状況

まず、安否確認システムの準備状況に関して、地震発生時までの中長期的な経緯であるが、大学として、大学 ICT 推進協議会 (AXIES) のクラウド部会の安否確認システム共同開発プロジェクト [22] に参加したり、商用の各種システムの導入可能性を検討していた。地震発生のほんの数カ月前にも商用システムを具体的に検討していた。しかし、時期的なこと、予算的なこと、それになによりも、正直なところ地震が起きるといふ切迫感が関係者の間に足りなかったこともあり、実際の導入には至っていなかった。

4.2 地震発生時の緊急対応

2016 年 4 月 14 日夜 21:26 熊本地震の前震 (最大震度 7) が起き、大学内でも、建物にヒビが入ったり、物が倒れたり、棚からものが落ちて散乱したりといった被害が発生すると同時に、学生、教職員も、自宅等でも同様の被害を受けており、中には負傷者もあり、自宅が危険なため避難している者も多数あった。翌朝から、大学では、部局ごとに、学生、教職員の安否確認の作業が開始され、主に電話連絡により作業が実施されていたが、大変な作業であり、なかなか進まなかった。幸いにして、電源と大学のネットワークが無事であったことから、工学部長から、総合情報統括センターに、簡単な CGI レベルのもので構わないので、安否を申告できるものが 2、3 時間で実現可能なら、電話等による現在の安否確認作業が楽になるのではという問合せがあった。すなわち、現在の安否確認作業を補助する手段として、できる限り早急に対応サービスが開始できないものかという相談であった。

そこで総合情報統括センターでは、直後から対応を検討し、開発に取り組むことで、3 時間弱後に安否確認サービスを開始した。ここで役に立ったのが、前述の認証やデータのシステム間連携、Web アプリケーションの仕様や雛形であり、詳細は、4.3 節の緊急開発で実際に述べる。工学部長から、全学で利用を開始するよう提案があり、その後、事務組織も含めて全学的な対応をとることとなった。すなわち、本震の前に一応、補助的ではあるが安否確認体制を整えた。

その夜、前震を上回る本震が熊本地方を襲った。本震の揺れは激しく、その被害はニュース等で知られるとおりである。大学は夜中であったため、幸い構内での人的被害はほとんどなかったが、停電、水漏れ、設備の破損等きわめて大きな被害があった。ネットワーク機器も L3 スイッチを含む多数の被害があったが、幸いにして、全学サービスの中心を担っているサーバ室の被害がほとんどなく、ネットワークの外部接続も短時間で復旧したため、構築したシステムはすぐに動作を継続することができた。

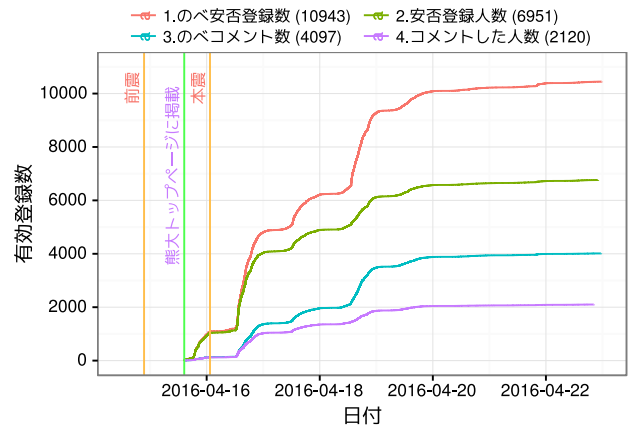


図 6 安否確認の登録回数とコメントの書き込み (全学生：10,471 名)
 Fig. 6 Number of registrations and comments of safety confirmation (all students: 10,471).

安否登録に関しては、4.3 節に示すように、当初からスマートフォンを含む Web ブラウザからアクセスでき、必須ではないがコメント入力もできるようになっていた。図 6 に、実際の安否登録状況を示すが、60%以上にあたる 6,951 名の学生が安否登録を行い、2,120 名が必須ではないコメントを残していたことが分かる。

その後、このシステムを改良しつつ、毎年、安否確認訓練を行っているが、残念ながら、これほどの高率になったことは今までない。

4.3 安否確認サイトの緊急開発

2、3 時間という短期間に開発する必要があったため、とりえず機能を最低限に絞った。SSO によるログインが必要なページであれば、そのページを見たときにはすでに CAS によるユーザ認証が完了しているはずで、ユーザ ID が取得可能である。つまり、最低限、そのユーザの情報を示し、安否確認登録するかの「確認」ボタンがあればよいことになる。ただ、緊急時であるため、連絡したい事項があるかもしれないと考え、コメントだけは入力できるようにした。また、停電地域や避難所等にいるユーザも多いと思われるため、スマートフォンに対応していることも必須とした。

その結果作成した Web アプリケーションのクライアント画面を図 7 に示す。この画面は、ほぼ初期版のもので、現在は 4.5 節に示すようにいくつかの改良がされている。実装は、まず、3 章の雛形をそのまま利用することも考えたが、サーバ部分のデータベース等で、SSO 認証されている学生番号や職員番号から氏名への変換や、時刻、ユーザ ID、コメント等の保存や検索を考えると、それらの機能を持つ既存アプリケーションへの WebAPI とクライアントページの追加が手早いと判断した。

Web サーバ部に WebAPI を追加するとともに、雛形ページからクライアント部分を追加、変更することで実装した。



図 7 学生・教職員の安全確認登録ページ

Fig. 7 Registration page of safety confirmation for students, faculties and staffs.

ここで実装した WebAPI は、認証情報から所属と氏名を返す API, そのユーザのこれまでのアクセス履歴を返す API のみで、日英併記で国際化はせず、jQuery Mobile によるモバイル対応といったものであった。

だれが安全確認をしたかを確認するための管理画面は、当初は単純なものしか提供できなかったが、それでもなんとか使えたのは、CAS によるアクセス制限 (教職員に限った) と Datatables.js ライブラリ [23] のおかげであろう。最初は、安全確認をした人を学生、教職員の区別なく、コメントのあるなしや、多重登録も含め、単に表に表示するものであり、だれが安全確認していないとか、コメントのあるもののみ抽出するといったこともできなかった。ただ、その表示に、Datatables.js を用いていたため、Datatables.js の機能で、検索、抽出、並べ替え等ができたことと、なにより、CSV で表のデータをダウンロードできたことで、事務組織でスプレッドシートやデータベース管理ソフト等で扱うことができた。

4.4 準リアルタイム統計情報

2.2.5 項で示した即時的分析・共有手法が、統計情報の提供に役立った。たとえば、図 6 の登録情報の可視化も、ある程度実時間でできたほうがよい。ここでは、図 3(a) の Rstudio Server 上で、学部や学年ごとの登録情報、アクセスしたデバイス情報等様々な統計情報を可視化する R のプログラムを開発し knitr で html 化し、10 分ごとに cron で実行していたため、10 分ごとに情報が更新される。ただし、地震発生当初は他の開発を優先していたため、扱う統計情報は徐々に増やしていった。図 8 に、統計情報の例として、最近の訓練におけるアクセスデバイスの分布を示す。

デバイスの詳細 (全アクセス数:6012)

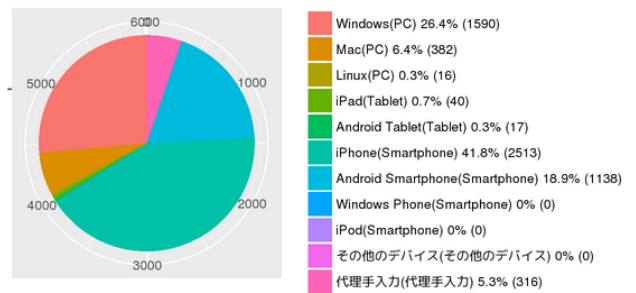


図 8 統計情報の例 (2018 年度 2 回目の訓練時のアクセスデバイス)
Fig. 8 Example of statistical information (access device at the second training in 2018).

4.5 安全確認システムとしての改良

地震発生後、1 年程度経ったときに、この緊急対応で開発した安全確認システムを大学としての安全確認システムに採用することになった。緊急で開発したもので、不備も多いが、半年ないし 1 年に 1 度の安全確認訓練に向けて、コメントの記述や学内構成員からの意見に対応し、改良を繰り返している。

まず、多かった意見として、携帯電話 (いわゆるガラケー) に対応していない点で、学生からはほとんどないが、教職員からはかなりあった。これに関しては、CAS の認証自体が携帯電話に対応していないこともあり、アクセスしたデバイスが携帯電話かどうか自動判断し、携帯電話の場合はシステムにはいらず、電話による登録用のページを表示するようにした。図 8 の代理手入力の中に、この電話による登録は含まれる。

CAS の認証画面でユーザ ID とパスワードを入れるのが大変という意見も多いが、これはブラウザに記憶させることによる対応をお願いしている。スマートフォンのアプリケーションを用意すれば、よりセキュアに解決できると思われる。

クライアントプログラムに関しては、図 7 の送信履歴をリンクではなく、ページ下部に表示することで、登録したかどうか分かりやすくした。また、送信ボタンを目立たせる等、見た目の変更も行っている。

管理画面に関しては、所属による絞り込み、代理登録、コメントの代理入力、未登録者の表示、メールによるコメントへの返信等、多くの機能を追加した。

5. 今後のシステム連携にむけて

大学における各種オンラインサービスの利便性を向上させるため、認証やデータのシステム間連携を推進してきた。しかし、その過程で見えてきた様々な問題とともに、今後の可能性について考える。

5.1 認証連携について

なぜ SSO として CAS を選択し、最初は学生番号、職員

番号を使い、その後、生涯 ID である熊本大学 ID を導入したかという点、それは 2.1 節で示したように、まさに時代の流れがそうであったからというのが本質であろう。今から、新たに、SSO 環境を導入するのであれば、Shibboleth と生涯 ID の組合せにすればよいのだが、現在のような状況においては、旧 ID から新 ID に徐々に移行しながら、認証も連携を検討するしかないと考えており、妙案はない。

5.2 データ連携について

5.2.1 標準化

2.2.1 項で示したように、データ連携の際、データ形式が標準化に従っていると、システム間の接続やシステムの移行の際、簡単に行える場合もある。ただし、対象システムがその標準化仕様を満足していない場合、対応させるのが大変で、現実的に困難な場合もあるだろう。

また、かつては、データ連携等にかかわる標準化は、静的なファイル交換が主で、ここで紹介した IMS Enterprise も巨大な XML ファイルでバッチ処理を前提としている。しかし、関連する新しい標準規格である IMS OneRoster [24] は、WebAPI によるリアルタイムのデータ交換も定義しており、他の標準規格も WebAPI の導入が進んでいる。2.2.7 項の Caliper についても WebAPI を利用している。このような動的な標準化が多く取り入れられるようになってくると、システム連携の形も変わってくるのではないかと思われる。

5.2.2 データ形式

図 2 を見ると、標準化を採用できているものは少なく、CSV、SQL 等個別フォーマットが大半であり、その同期処理は個別のシェルスクリプト等で行っている場合が多い。シェルスクリプトといっても、中は、シェルコマンドだけではなく、各種外部コマンドの組合せで、grep, sed, sort, cut, wc, perl... と様々で、また、作成者の癖のあるコードであることもあり、なかなか共有が難しい。

このあたりは、ETL (Extract Transform Load) ツール [25] の導入が有効ではないかと考える。ETL ツールとは、種々のシステムからデータを抽出 (Extract) し、必要に応じて変換 (Transform) し、利用先システムで読み込む (Load) ためのミドルウェアである [25]。その用途は、各種システム連携 [25], [26] に活用でき、属人性を減らすことができる [26]。

また、CSV や SQL の連携ならまだよいが、WebAPI で受け取ったデータを加工してデータベースへ入れるとか、別の WebAPI へわたすとか、動的な入出力がある場合シェルスクリプトだと限界があると思われるが、ETL ツールの場合、WebAPI をサポートしている場合もあり、このようなケースにも利用できる。

本報告の 3 章の雛形に関しても、認証部分さえなんとかなれば、WebAPI をサポートする ETL ツールでサーバ部

分を構築できる可能性もある。

5.2.3 データの可視化

2.2.5 項で紹介した即時的分析・共有システムは、基本的に R でできることなら何でもできるが、逆にいえば、R 言語が分からないと使えないし、どんな簡単な処理でも R 言語で記述しなければならない。この部分は、Business Intelligence (BI) ツールが使える可能性がある。BI ツールは、企業等でよく利用されるデータの分析可視化により迅速な意思決定を助けるツールで、大学においても Institutional Research 部門で最近使われるようになってきている。BI ツールによっては、Web 上にユーザ制限付きでアクセス時の分析結果を公開したり、WebAPI を持つものもあり、このようなツールを使えば、そのつど、分析可視化した結果を得たり、定期的に更新したりできる可能性がある。

5.3 安否確認システムからの知見について

熊本地震時の安否確認への緊急対応に関して、今回報告した認証やデータのシステム連携環境、その環境下での Web アプリケーションの雛形が大いに役立った。しかし、その前提として、システムが動作しているという大前提があり、今回はその面では非常に幸運であった。今回の経験からも、認証やデータ連携の基盤のクラウド化は必須条件である。また、そのうえで動作するシステムに関しても、緊急時に止められないシステムから優先してクラウド化を進めるべきである。

6. おわりに

大学のオンラインサービスが増えるに従い、認証やデータのシステム連携を推進してきた。様々な事例を紹介するとともに、その環境に適合した緩やかな仕様と雛形の提供を行うことで、Web アプリケーションの導入を円滑に行ってきた。その中で起きた熊本地震発生時の安否確認に対して、紹介した資源を活用して対応し、その後の発展も含めて紹介した。これらの経験から、今後のシステム連携のあり方について論じた。

謝辞 本研究は、熊本大学総合情報統括センター、大学院教授システム学専攻、情報企画課をはじめとする、大学の情報システムに関係する多くの方のご協力で成り立っており、ここに深く感謝します。また、本研究の一部は、JSPS 科研費 15H02795 の助成を受けています。

参考文献

- [1] 杉谷賢一：熊本大学学務情報システム—SOSEKI, 学術情報処理研究誌, No.3, pp.51-52 (1999).
- [2] 梶田将司, 板倉文忠：WebCT によるコースウェア作成支援環境の構築, 電子情報通信学会技術研究報告教育工学 (ET), Vol.99, No.459, pp.15-22 (1999).
- [3] 中野裕司, 喜多敏博, 杉谷賢一, 松葉龍一, 右田雅裕, 武蔵泰雄, 入口紀男, 太田泰史, 平 英雄, 辻 一隆,

- 島本 勝, 木田 健, 宇佐川毅: WebCT, 学務情報システム SOSEKI, 教育用 PC システムのデータ同期, 第 2 回 WebCT 研究会予稿集, pp.3-8, 淡路島 (2004).
- [4] 梶田将司, 内藤久資, 小尻智子, 平野 靖, 間瀬健二: CAS によるセキュアな全学認証基盤の構築, 情報処理学会研究報告インターネットと運用技術 (IOT), Vol.2005, No.39, pp.35-40 (2005).
- [5] 大谷 誠, 江藤博文, 渡辺健次, 只木進一, 渡辺義明: シングルサインオンに対応したネットワーク利用者認証システムの開発, 情報処理学会論文誌, Vol.51, No.3, pp.1031-1039 (2010).
- [6] 松平拓也, 笠原禎也, 高田良宏, 東 昭孝, 二木 恵, 森祥寛: 大学における Shibboleth を利用した統合認証基盤の構築, 情報処理学会論文誌, Vol.52, No.2, pp.703-713 (2011).
- [7] 松浦健二, 上田哲史, 佐野雅彦: 複数認証基盤に対応する複合 SSO 環境でのユーザエクスペリエンス, 学術情報処理研究, Vol.16, No.1, pp.138-145 (2012).
- [8] 戸田英貴, 江木啓訓, 須田良幸, 品川徳秀: Moodle と学務情報システムのデータ連携の設計と課題, 情報処理学会研究報告コンピュータと教育 (CE), Vol.2008, No.64, pp.49-54 (2008).
- [9] 榊田秀夫, 村田和義, 渋谷 雄: 低コストな高可用性と学務システム連携を考慮した Moodle システムの構築, 情報処理学会研究報告インターネットと運用技術 (IOT), Vol.2008, No.37, pp.65-69 (2008).
- [10] 新村正明, 長岡暁子, 石田美代子, 長谷川理, 矢部正之: 履修登録情報の LMS への反映方法の検討と実装, 教育システム情報学会第 40 回全国大会講演論文集, Vol.G1-1, pp.89-90 (2015).
- [11] 永井孝幸, 杉谷賢一, 河津秀利, 中野裕司: 学認対応認証基盤とユーザ ID 体系移行用 CAS ゲートウェイの構築, 電子情報通信学会技術研究報告技術と社会・倫理 (SITE), Vol.113, No.355, pp.119-128 (2013).
- [12] 中野裕司, 喜多敏博, 杉谷賢一, 松葉龍一, 右田雅裕, 武蔵泰雄, 入口紀男, 北村士朗, 根本淳子, 辻 一隆, 島本 勝, 木田 健, 宇佐川毅: WebCT(4/6)-CAS-uPortal SSO 連携の Servlet/Portlet による実装, 第 4 回 WebCT ユーザカンファレンス予稿集, pp.1-6 (2006).
- [13] 中野裕司, 喜多敏博, 杉谷賢一, 松葉龍一, 右田雅裕, 武蔵泰雄, 入口紀男, 太田泰史, 平 英雄, 辻 一隆, 島本 勝, 木田 健, 宇佐川毅: WebCT、学務情報システム SOSEKI、教育用 PC システムのデータ同期, 第 2 回 WebCT ユーザカンファレンス予稿集, pp.3-8 (2004).
- [14] 中野裕司, 杉谷賢一, 永井孝幸, 宇佐川毅: 授業時間割 Web API とそのポータル時間割への応用, 日本教育工学会第 31 回全国大会 (電気通信大学) 発表論文集, pp.309-310 (2015).
- [15] サイエントیفイック・システム研究会 e ポートフォリオ研究 WG 編: e ポートフォリオ研究 WG 成果報告書, 4.1 熊本大学, pp.69-73 (2015).
- [16] 中野裕司, 鶴田博信, 喜多敏博, 永井孝幸, 杉谷賢一: Web API 技術を活用したシラバスシステムのモックアップを用いた設計と実装, 教育システム情報学会第 40 回全国大会発表論文集, pp.7-8 (2015).
- [17] 中野裕司, 藤山泰成, 河津秀利, 森 保夫, 久保田真一郎, 杉谷賢一: researchmap を中心に据えた Web API とマッシュアップによる大学研究者総覧構築の試み, 大学 ICT 推進協議会 2017 年度年次大会, TA2-6 (2017).
- [18] 中野裕司, 喜多敏博, 松葉龍一, Wannous, M., 宇佐川毅: Rstudio Server, knitr 等を活用したフレキシブルな Web によるデータ分析, 第 41 回教育システム情報学会全国大会, I1-7, pp.13-14 (2016).
- [19] 中野裕司, 榎原竜之輔, 喜多敏博, 戸田真志, 久保田真一郎, 右田雅裕, 杉谷賢一: Caliper log store と Open LRW を用いた Moodle 上の学習履歴のログサーバへの Caliper 標準形式による集積の取り組み, 研究報告教育学習支援情報システム (CLE), Vol.2018-CLE-25, No.11, pp.1-5 (2018).
- [20] 中野裕司, 永井孝幸, 松葉龍一, 喜多敏博, 杉谷賢一, 宇佐川毅: 共有化を狙った Web アプリケーション仕様の提案とサンプルの開発の試み, 大学 ICT 推進協議会 2013 年次大会論文集, T1A-9, pp.319-322 (2013).
- [21] 中野裕司, 新富葉子, 森 保夫, 森部英俊, 杉谷賢一, 宇佐川毅: 熊本地震発生時における安否確認システムの緊急開発と運用及び今後の課題, 大学 ICT 推進協議会 2016 年度年次大会, WF24, pp.1-8 (2016).
- [22] 梶田将司: 大学 ICT 推進協議会クラウド部会の活動紹介, p.37 に経緯, p.40 に参加大学, 入手先 (http://www.nii.ac.jp/userimg/20130208_12.pdf) (参照 2019-03).
- [23] DataTables Table plug-in for jQuery, 入手先 (<https://datatables.net/>) (参照 2019-03).
- [24] IMS OneRoster, homepage: 入手先 (<https://www.imsglobal.org/activity/onerosterlis>) (参照 2019-03).
- [25] 田中哲雄, 湯本真樹, 齋 礼: 企業情報システムにおける連携技術, 電気学会論文誌 C, Vol.124, No.5, pp.1051-1057 (2004).
- [26] 永井孝幸, 山岡裕美, 榊田秀夫: 京都工芸繊維大学における利用者原簿管理基盤の強化と連携サービスの構築, 情報処理学会研究会報告, Vol.2018-CLE-25, No.9, pp.1-8 (2018).



中野 裕司 (正会員)

1959 年生. 熊本大学総合情報統括センター・大学院社会文化科学教育部教授システム学専攻教授, 理学博士 (九州大学, 1987 年), 名古屋大学教養部, 同大学情報文化学部助教授を経て 2002 年より熊本大学教授. 電子情報通信学会, 日本教育工学会, 教育システム情報学会等各会員.