

高等教育機関における 次世代コース管理システムの構築に向けて†

梶田将司*・角所考**・中澤篤志***・竹村治雄***・美濃導彦**・間瀬 健二*

名古屋大学情報連携基盤センター*・京都大学学術情報メディアセンター**

大阪大学サイバーメディアセンター***

コース管理システムは、教育機関である大学が教育の情報化を進める上で必要となる中核的なソフトウェアであり、講義・セミナー・実験などの教育・学習現場を総合的に支援する。また、コース管理システムは、履修登録や成績管理などを行う教務システムやシラバスシステム、電子図書館など、他の学内システムと連携させるため、大学の情報基盤システムとして戦略的な導入・活用および継続的な開発が必要となる。本論文では、コース管理システムに関する要素技術を、オープンソースやオープンスタンダードの観点からソフトウェアアーキテクチャを軸としながら俯瞰的に解説する。そして、その先端事例として2004年度から我々が研究開発を行っている次世代コース管理システムについて解説する。

キーワード：コース管理システム、オープンソース、オープンスタンダード、コンテキストウェアネス、大学における情報基盤システム

1. は じ め に

近年の情報技術 (Information Technology, IT) 分野における技術革新の進展は、高等教育機関における教育・学習活動に関わるすべてに大きな影響を与えている。その結果、教育・学習活動での効果的な IT 活用、認証評価・職業人教育・生涯教育・予算削減・国際化など多くの課題を抱える現在の高等教育機関にとって極めて重要な課題となっている。特に、教育現場における多様なニーズに効率的に対応するため、米国やカナダ、オーストラリア等の高等教育機関においては、コース管理システム (Course Management System,

CMS) が普及している。CMS は、高等教育機関における教育・学習活動を、講義時間外も含め、IT により総合的に支援するためのシステムである。そのため、CMS は、高等教育機関の教育・学習活動に不可欠な情報基盤システムとして位置づけられている。例えば、米国における CMS の導入率はすでに90%を越えるとともに、46.8%を越える講義等で活用されている (GREEN 2006)。我が国においては、ラーニングマネジメントシステム (Learning Management System, LMS) の利用率が、大学の学部・研究科では約46.9% (調査対象の学部・研究科数は3,570, 回答率は52.7%) となっている (NIME 2007)。米国との厳密な比較はできないが、我が国においても、CMS あるいは LMS の利用は着実に進み始めていると言える。

このような流れの中、北米の高等教育機関では、Andrew W. Mellon 財団や NSF (National Science Foundation) による支援の下、オープンソースソフトウェアやオープンスタンダードを基軸としたアプリケーション開発やミドルウェア開発の流れが加速している。具体的には、大学ポータル構築のための開発フレームワークである JA-SIG (Java Architecture Special Interest Group) の uPortal を発端に、ユーザ認証・権限管理を目的とした JA-SIG の CAS (Central Authentication

2007年8月13日受理

† Shoji KAJITA*, Koh KAKUSHO**, Atsushi NAKAZAWA***, Haruo TAKEMURA***, Michihiko MINOH** and Kenji MASE*: Next-generation Teaching and Learning Platform for Higher Educational Institutions

* Information Technology Center, Nagoya University, Furo-cho, 1, Chikusa-ku, Nagoya, 464-8601 Japan

** Academic Center for Computing and Media Studies, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto, 606-8501 Japan

*** Cyber Media Center, Osaka University, Machikaneyama-cho, 1-32, Toyonaka 560-0043 Japan

Service) や Internet2 の Shibboleth, アプリケーションのモジュール化を推進することを目的とした MIT OKI (Open Knowledge Initiative) の OSID (Open Service Interface Definition) などが挙げられる。現在では, 100 を越える主要な大学が参加する Sakai Foundation が設立され, 「コミュニティソース」という言葉で具現化されたアプリケーション開発フレームワークとその開発者・運用者・利用者のコミュニティが形成されはじめている。

このような背景の下, 我々は, 我が国がリードするユビキタス情報環境下における次世代コース管理システムの開発を「ULAN (Ubiquitous Learning Architecture for Next-generation) Project」として行ってきた。また, その過程において, 主に北米のプロジェクトとの連携を目指して, オープンソース・オープンスタンダードの動向を調査した。

そこで, 本論文では, コース管理システムに関する要素技術を, オープンソースやオープンスタンダードの観点からソフトウェアアーキテクチャを軸に据え俯瞰的に解説する。そして, その先端事例として我々が2004年度から研究開発を行っている次世代コース管理システムについて解説する。

本論文の構成は次の通りである。まず2章では, コース管理システムを, システムアーキテクチャおよびソフトウェアアーキテクチャの観点から整理する。そして, 3章では, 北米の大学における情報基盤システムに関わる動向を中心に, 4層ソフトウェアアーキテクチャを軸としたオープンソース・オープンスタンダードの動向を述べる。そして, 4章では, 我々が開発している次世代コース管理システムについて述べる。なお, オープンソースの実装言語としてはJavaを対象とし, それ以外の言語に関する動向については対象外とする。ただし, PHP ベースの Moodle に関しては, 3.3.節にて特長を簡単にまとめる。さらに, IEEE による LOM (Learning Object Metadata) や ADL による SCORM (Sharable Content Object Reference Model) など, コンテンツに関わるオープンスタンダードについても言及しない。

2. コース管理システムのアーキテクチャ

2.1. 情報基盤としてのコース管理システム

コース管理システムを語る上で忘れてはならない点だが, 「コース管理システムは各大学が教育・学習活動を行うための情報基盤システムである」という点である。しかしながら, 研究ベースで開発が行われる場合, 高

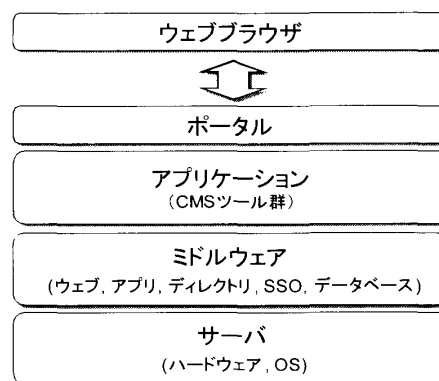


図1 コース管理システム等の情報基盤システムにおけるシステムアーキテクチャ

可用性, 高負荷対策, システム統合のための API (Application Programming Interface) 等の大学の情報基盤システムとしての機能は特に必要がないため, 軽視されがちである。実際, 複数のコースで共通に利用可能な機能を「ツール」としてまとめ, 共通化を図るというアイデアをもとに開発が始まった WebCT (Web Course Tools) (GOLDBERG *et al.* 1996) でさえ, 「バージョン 1.X では, コースは島だった」と言われるように, コースごとにユーザ ID 自体異なっていた。ところが, ひとつの WebCT サーバでの利用者が増加するにつれて, ユーザ ID の共通化がなされ, 教務システムのような「学生情報システム」(Student Information System, SIS) との統合や「大学ポータル」との統合を通じて, 大学の情報基盤としての機能が強化され, いわゆる「エンタープライズ」向けアプリケーションとしての高可用性, 高負荷対策, システム統合のための API 等が整備された。そして, 2001年頃からは「コース管理システム」と呼ばれるようになっていく (梶田 2004)。

この「エンタープライズ」化の過程において, 北米では, 雨後の筈のごとく乱立していた e-Learning システム市場は, 「エンタープライズ」化対応を完了した WebCT と Blackboard による占有化が進み, 最終的には, Blackboard 社と WebCT 社の企業合併が行われ, Blackboard による寡占化へと至っている。

しかしながら, "mature market with immature applications" (GREEN 2006) と言われるように, 現在のコース管理システムが有する機能は十分なものとは言えず, 第1ラウンドが終わったに過ぎない。コース管理システムを使う教員・学生が増え, 利用度が深化するとともに, Ajax (Asynchronous Java Script + XML) や SOA (Service Oriented Architecture) など, 新しい技術を取り入れながら次の第2ラウンドがどのように展開

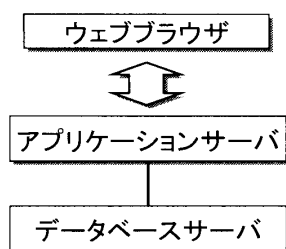


図2 最も簡単なシステム構成

するかを予想することは難しい。しかしながら、「大学の情報基盤システム」としてのコース管理システムの位置づけは不変である。しかも、情報基盤システムとして各大学がコース管理システムを整備する上で極めて重要な観点となるのが「オープンスタンダード」であり「オープンソース」である。

2.2. コース管理システムのシステムアーキテクチャ

エンタープライズアプリケーションとして構築される大学の情報基盤システムでは、「ウェブ」「アプリケーション」「ディレクトリ」「シングルサインオン」「データベース」等のサーバが提供するミドルウェアを利用しながら、「アプリケーション（CMS ツール群）」を提供するためのソフトウェアが開発・実行される。そして、一元的にユーザに提供する「ポータル」を通じて、「ウェブブラウザ」によりユーザは閲覧することになる（図1参照）。これらすべての機能を同一サーバ上で実行することも可能であるが、通常、データベースサーバを他と分離し、別サーバとして機能させることが多い（図2参照）。高可用性や高負荷対策が必要な場合は、アプリケーションサーバをクラスタ化し、負荷分散装置により負荷を分散させるとともに、データベースサーバやアイデンティティサーバ（ディレクトリ・SSOサーバ）もクラスタ化することになる（図3参照）。情報基盤システムとしてコース管理システムを運用することを前提とするためには、図3のようなフレキシブルなシステムアーキテクチャで構成できるように、ソフトウェアアーキテクチャを設計する必要がある。

なお、各種サーバのハードウェアやOSに関しては、管理コストを下げるための共通化が行われる。最近では、サーバ仮想化技術の利用によりサーバ資源の有効活用も始まっている。

2.3. コース管理システムのソフトウェアアーキテクチャ

言うまでもなく、コース管理システム開発は「ソフトウェア開発」そのものである。「プログラミング言語」「ソフトウェア開発フレームワーク」「統合開発環

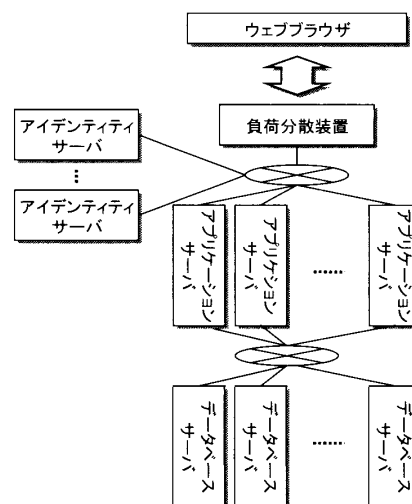


図3 高可用で高負荷対策を行った場合のシステム構成

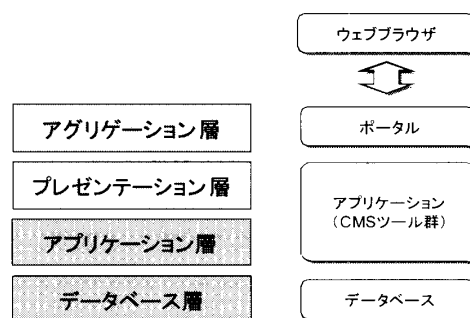


図4 4層によるソフトウェアアーキテクチャ

境」「開発方法論」など、ソフトウェア開発を進める上で必要となる事項は幅広い。また、開発チームに関係する人々も幅広い人材が必要であり、プロジェクトマネージャ、プログラマ、デザイナーだけでなく、実際にコース管理システムを使用する教員・学生などのユーザや、実行環境へのソフトウェアの配備や日々の運用管理を行うシステム管理者などで構成される。このように、様々な技術や関係者が関わり、ソフトウェア開発の規模が大きくなるにつれて、ソフトウェアアーキテクチャが重要になる。

本論文では、ソフトウェアアーキテクチャとして、「アグリゲーション層」「プレゼンテーション層」「アプリケーション層」「データベース層」の4層アーキテクチャを用いる（図4参照）。これは、モデル・ビュー・コントロールを分離する MVC 型のアーキテクチャにポータルとの対応を考慮した「アグリゲーション層」を加えたものである。システムアーキテクチャではそれぞれ「ポータル」「アプリケーション」「ミドルウェア（データベース）」に対応する。

3. コース管理システムに関するオープンソースおよびオープンスタンダードの動向

3.1. なぜオープンソース・オープンスタンダードか？

2章でまとめたように、コース管理システムは「大学の情報基盤システム」として用いられ、大学の最も重要な活動である教育・学習活動の中核的なシステムである。このため、大学としては長期的な戦略の下で、コース管理システムの開発・導入・運用・更新を考える必要がある。しかしながら、実際は、独自開発の場合、ソフトウェア開発を行う業者や個人に強く依存しすぎ、独自開発であるにも関わらず、著作権上独自にカスタマイズができなかったり、担当者の退職により維持管理ができなくなったりする。商用プラットフォームを用いる場合でも、ベンダーへの依存が大きすぎ、ベンダーの製品戦略の影響を受けたり、カスタマイズに多額の費用が発生する場合もある。

このような状況から大学を守るための方策が、「オープンソース」であり「オープンスタンダード」である。「オープンソース」であれば、開発コストや保守コストを抑えることができるし、業者や個人に依存していた状態をオープンソースのコミュニティベースとすることで長期的な維持管理を実現できる。もちろん、これらは、オープンソースコミュニティの活発度・広がりや著作権、配布ライセンス形態にも依存する。このため、大学自身がオープンソースコミュニティに積極的に寄与したり、開発した大学あるいは組織が著作権を有したり、「改変」「再配布」「事業化」の観点においてできるだけ緩い配布ライセンスを認めることが望ましい。

一方、「オープンスタンダード」に準拠したものであ

れば、特定のベンダーやオープンソース実装に依存しなくすることができる。特に、上手く定められた「オープンスタンダード」準拠のモジュール群で全体が構成できるようになれば、各高等教育機関が独自の戦略に基づいた最もよいモジュール（ベンダー製、オープンソース、独自開発モジュールなど）の組み合わせによりコース管理システムを構築できるとともに、特定のベンダーによる独占を排除することで、競争原理を働かせ、不必要なコスト高騰を防ぐことができる。これらのメリットは、オープンソースをベースに製品開発を行ったり、オープンスタンダードに製品を準拠させたりすることにより、ベンダー自身も享受することができる。このため、オープンソース・オープンスタンダードの方向を目指すことは、大学・ベンダー双方にとって望ましいものである。

このような「オープンソース」「オープンスタンダード」重視の方向性は、Andrew W. Mellon 財団の支援による uPortal や MIT OKI の達成した成果をベースに始まった Sakai Project の一連の流れとして顕在化している。特に、Sakai に関しては、独自開発によるコース管理システムの開発・運用を進めてきた研究大学が、多様なニーズに限られた資源で対応するために、共通部分をオープンソースとして共同開発することになった点に注目すべきである（梶田 2004）。

3.2. 4層アーキテクチャで見る全体像

「アグリゲーション層」「プレゼンテーション層」「アプリケーション層」「データベース層」の4層アーキテクチャに沿ってまとめたオープンソースおよびオープンスタンダードの概要を図5に示す。図では、JCP (Java Community Process) や OASIS (Organization for

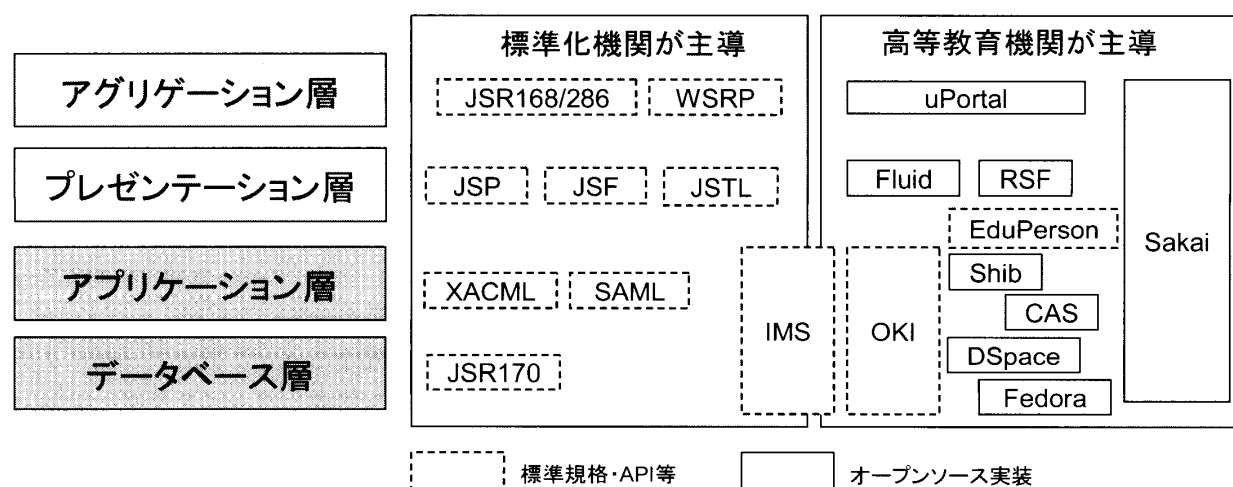


図5 大学の情報基盤システムとしてのコース管理システムに関するオープンソース・オープンスタンダードの動向

the Advancement of Structured Information Standards), IMS Global Learning Consortium, MIT OKI 等が策定している標準規格・API および JA-SIG, Sakai 等が開発しているオープンソース実装を、それぞれ、標準化機関が主導するものと高等教育機関が主導するものに分けて示してある。

まず、アプリケーション層では、JA-SIG の uPortal が最も有名かつ広く使用されている。uPortal は、高等教育機関用のポータルを作成するためのフレームワークで、Java クラスのセットおよび XML/XSL ドキュメントで構成される。uPortal は、JA-SIG のメンバである大学・企業の開発者がコミュニティ・デベロプメントの考え方で開発を行っており、無償のリファレンスインプリメンテーションとして uPortal コードが利用可能になっている。uPortal は米国のソフトウェア開発業界において高く評価されており、2002年には InfoWorld Top 4プロジェクトに、2007年6月には EDUCAUSE Catalyst Award を受賞している。Sakai でも、Charon Portal などいくつかのポータル実装を提供しているが、機能的には uPortal の方が勝っている。uPortal と Sakai は、Andrew W. Mellon 財団の支援の下で行われた Sakai Project を通じて、JCP が定める JSR 168 Portlet 標準や、OASIS が定める WSRP (Web Services for Remote Portlets) 標準による統合の努力がなされたが、JSR 168や WSRP の未成熟さにより十分な統合には至っていない。しかしながら、JSR 168準拠の Portlet 実行環境として Apache Project の Pluto をベースに uPortal および Sakai の Portlet 対応は着実に進んでいる。また、JSR 168の次バージョンである JSR 268の策定への Sakai Foundation (次節参照) としての参画も行われており、将来的には JSR 268により Sakai と uPortal の統合が完了すると思われる。

次に、プレゼンテーション層では、動的な HTML を生成するための SSI (Server Side Include) テクノロジーの Java 版として2001年に JSP (JavaServerPages) が JSR 58 として導入された。しかし、HTML に Java 言語を直接埋め込むため、Java 言語を理解しないウェブデザイナーには取り扱いづらいものであった。このため、2002年には JSR 52として JSTL (A Standard Tag Library for JavaServer Pages) が、2004年には JSR 127として JSF (JavaServer Faces) が策定され実装が進んだ。その後、JSP は JSR 152 (JavaServer Pages 2.0), JSR 245 (JavaServer Pages 2.1), JSTL は JSR 267 (JSP Tag Library for Web Services)と進化している。このような努力にも

関わらず、User Interface (UI) の構築作業はデザイナーおよびプログラマの共同作業であることは変わりなく、ユーザにとって使いやすい UI を実現するためのさらなる改善が進められている。例えば、2007年からはトロント大学 (カナダ) 主導で Fluid Project が Andrew W. Mellon 財団の支援で開始され、ユーザビリティ、アクセシビリティ、国際化、品質管理、セキュリティ対策のための共通的なメカニズムを提供することを目指している。また、ケンブリッジ大学 (英国) は、Sakai Project においてロジック処理との結合なしに見た目をデザイン可能な RSF (Reasonable Server Faces) を提案している (RSF 2007)。

アプリケーション層では、ユーザ認証 (Authentication) や権限管理 (Authorization) に関する標準やオープンソース実装が顕在化している。まず、LDAP (Lightweight Directory Access Protocol) で用いられる大学機関向けの標準スキーマとして EduPerson を EDUCAUSE が定めている。その他、SSO (Single Sign On) 機能を提供する JA-SIG の CAS や Internet2 の Shibboleth が有名である。国内でも国立情報学研究所の CSI (Cyber Science Infrastructure) 事業の一環として進められている UPKI イニシアチブにおいて Shibboleth の検討が行われている (UPKI 2007)。また、名古屋大学や熊本大学では CAS の実運用が行われている (内藤 2006)。また、ユーザ認証・権限管理に関する標準化として、Shibboleth でも一部採用されている OASIS の SAML (Security Assertion Markup Language) や XACML (eXtensible Access Control Markup Language) がある。一方、アプリケーションとして具体的なツールを作成する際の支援となるソフトウェア開発フレームワークとしては Sakai Framework がある (次節参照)。OKI や IMS でも関連するものがあるがその利用は進んでいない。また、Java によるプログラミングパラダイムは、Spring Framework に代表される Dependency Injection (実行時における依存性の注入) が顕在化しており、簡素に作られた Java Object (Plain Old Java Object, POJO) をベースに Dependency Injection による疎結合を実現する形でソフトウェア開発が進められる。

データベース層では、「リポジトリ」と呼ばれる様々なコンテンツの集積・格納場所に関する標準化やオープンソース実装が顕在化している。例えば、機関リポジトリのためのオープンソース実装として、MIT の DSpace やバージニア大学・コーネル大学の Fedora がある。Sakai でも JSR 170 (Content Repository for Java Technology

API) に準拠したリソースツールの開発や, MIT OKI の Repository OSID (Open Service Interface Definition) による様々なリポジトリシステムとの結合が進んでいる。

3.3. Sakai

Sakai Project は, ミシガン大学のリーダーシップの下, MIT, インディアナ大学, スタンフォード大学, および, JA-SIG が参画し, Andrew W. Mellon 財団の支援により 2003年12月から開始された。そして, 2年間のプロジェクト期間中, Sakai Educational Partner Program (SEPP) を組織化, さらに2005年10月には非営利団体 Sakai Foundation を登記し, 現在では, 約100の大学や企業の会費により運営されている (SEVERANCE 2007)。

Sakai では, アプリケーション層における機能を実装するための支援として API とリファレンス実装の組み合わせで Sakai Framework を提供している (Sakai 2.2 Framework 2006)。

2007年5月にリリースされた Sakai 2.4では, Generic Tools として26種類, Teaching and Portfolio Tools として17種類, さらには, Contributed Area として10種類のツールが提供されている (SEVERANCE 2007)。

3.4. MIT OKI および IMS

MIT Open Knowledge Initiative (OKI) は, 高等教育機関における教育研究活動を支える大規模アプリケーション開発を容易にするための API を規定する活動を行っており, OKI API に準拠したモジュールであればインターオペラビリティが保障され, モジュールレベルでの独自開発やベンダーシステムの利用が可能になることを目指している。現在, 活発に連携しているメンバは, ウィスコンシン大学, ミシガン大学, ペンシルバニア大学, ダートマス大学, ケンブリッジ大学, ノースキャロライナ州立大学, MIT である。OKI では, OSID (Open Source Interface Definition, オープンソース・インタフェース規格) と呼ばれる12項目の API 規格を策定している (梶田・間瀬 2006)。

一方, IMS グローバル・ラーニング・コンソーシアムは, 相互運用可能なラーニングテクノロジーのためのオープンな仕様を策定し, 普及を行う非営利団体である。IMS は50以上の正規メンバおよび機関で構成され, ソフトウェアベンダー, 教育機関, 出版社, 政府機関, システムインテグレータ, マルチメディアコンテンツプロバイダ, 他のコンソーシアムが参画している。IMS が主にデータの標準化を中心に行っているのに対し, OKI はモジュール間の API を定めている (梶田・間瀬 2006)。

3.5. オープンソース CMS とそのコミュニティ

オープンソースベースでのコース管理システムを考える上で, 開発者・運用者・利用者で形成されるコミュニティは極めて重要である。Java ベースのものでは, 3.3. 節で述べた Sakai が国際的に最も大きなコミュニティを形成している。一方, 国際的なコミュニティを形成している PHP ベースのコース管理システムとして Moodle がある。Moodle との違いに関しては Sakai Foundation では次のように整理している (SEVERANCE 2007):

3.5.1. Moodle

- 成熟した製品, 豊富な機能 (1.0は2002年8月にリリース)
- 操作が簡単, インストールが簡単
- 中心的なリーダーシップにより洗練されたルック & フィールが保たれている
- PHP で開発されており, 開発者コミュニティの規模は大きい
- IT スタッフが小規模な組織に対して極めてよいソリューション

3.5.2. Sakai

- 若い製品でまだ成長中 (1.0は2004年6月にリリース)
- Java 言語で開発されており, 開発者にとっては多くのことを学ぶ必要がある
- サービス指向・コンポーネントベースのアーキテクチャ
- エンタープライズ製品としての機能を有しており, 20万人ユーザ以上でも対応可能

最終的には, Moodle と Sakai の双方のよい面が上手く組み合わせられることがコース管理システムユーザにとっては最もよいと思われる。

一方, Sakai や Moodle ほどは国際的に広がっていないものの, 教材ベースのコミュニティを形成しているコース管理システムとしては, ミシガン州立大学が Perl ベースで開発している LON-CAPA がある (LON-CAPA 2007)。また, 我が国でも Java ベースの CMS/LMS として, 東京大学の CFIVE (CFIVE 2007), 関西大学の CEAS (CEAS 2007), NTT レゾナントの Sui² (Sui² 2007) がある。これらの今後の展開に期待したい。

3.6. 動向のまとめ

本章では, 北米の大学において情報基盤システムとして開発・運用されているコース管理システムに関する要素技術を Java ベースのオープンソース・オープンスタンダードの観点から外観を述べた。運用が前提となっているシステム開発の場合, オープンソース実装

はオープンスタンダードよりも優先度は高い。しかしながら、大学として戦略的にコース管理システムを整備するためにはオープンスタンダードも合わせて追求する必要がある。

また、次世代のプラットフォーム構築を行う場合も、学術研究に重きを置くよりも、ニーズやシーズに基づいてソフトウェアアーキテクチャを設計し、これまでの成果の上に実装していくことが重要になる。

4. ユビキタス環境下における 次世代コース管理システムの開発

4.1. コンテキストウェアネス：次世代コース 管理システムに求められる機能

最近のFTTHによる広帯域化や無線LANによるローカル環境でのネットワーク接続の簡便化、携帯電話網による広域でのネットワーク接続の簡便化・広帯域化により、教育・学習の場は従来の教室から自宅・職場などに広がってきており、通学時や飲食店などでの自学自習も可能になってきている。しかも、コース管理システムにウェブブラウザを使ってアクセスできる端末環境も、従来のデスクトップ型PCに限らず、モバイルPC、携帯電話やPDA等といった携帯型端末、iPodのようなデスクトップソフトウェアと連携する携帯型デバイスなど、多様化している。このように、ネットワークや端末、あるいは、利用場所など、多様化する利用環境の特性をうまく制御することにより、教育・学習環境の高度化を図る必要性が生じてきている。

このような利用環境、すなわち、コンテキストに応じた適応的なサービス機能（コンテキストウェアネス）が、大学における教育・学習活動に及ぼす重要性は、すでに「コース管理システム先進国」の米国においても語られている。例えば、New Media Consortium および EDUCAUSE Learning Initiative が高等教育に重要なインパクトを与える可能性が高い新しい技術について毎年まとめている“The Horizon Report”の2006年度版では、今後注目すべきテクノロジーとして「コンテキストウェア環境およびデバイス（Context-Aware Environments and Devices）」が挙げられ、4年から5年後の教育現場での利用が期待されている（New Media Consortium 2006）。

4.2. ユビキタスクラスルーム uClassroom

我々は、上述のコンテキストウェアネスを実装可能な次世代コース管理システムとして「ユビキタスクラスルーム uClassroom」を開発している（図6参照）。

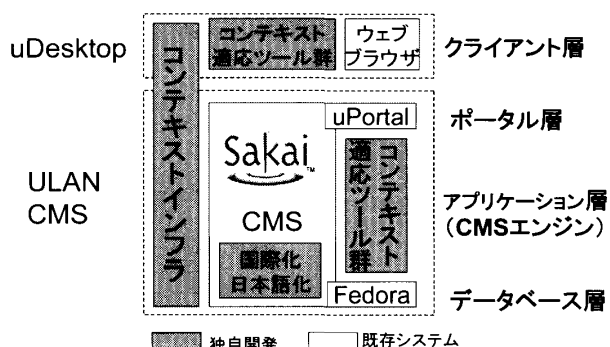


図6 uClassroom の4層ソフトウェアアーキテクチャ

uClassroom は、(1) コース管理システムとユーザをつなぐための「ネットワーク」コンテキスト、(2) ユーザがコース管理システムにアクセスするために使用する機器に関する「端末」コンテキスト、(3) ユーザがコース管理システムにアクセスする場所や時間に関する「時空間」コンテキスト、(4) 教員と学生がやりとりをするために必要な機器とそのやりとりの内容に関する「インタラクション」コンテキスト、(5)「教育学習」コンテキストの5つのコンテキストに適応可能であることを特徴とする。

uClassroom は、現行のクラスルーム空間において自然になされている教員と学生間のインタラクションやウェアネス(気づき)を、教員あるいは学生がuDesktopを開く場所でいつでもどこでも同様のインタラクションやウェアネスを提供することを目指すものである。uDesktop上の活動作業はお互いにプライバシーを尊重しつつuClassroomのユーザに知れるところとなり、互いの気づきを提供できるクラスルームを、ユビキタス化することが可能となる。気づきは互いの気配りを起こし、プライバシーの尊重は安心をもたらす（KAJITA and MASE 2006）。

4.3. uClassroom の研究開発

研究開発は、基盤ソフトウェア開発サブグループ、3大学合同実証実験サブグループ、応用ソフトウェア開発サブグループにおいて、それぞれ「プラットフォーム」「教材」「ユーザインタフェース」の観点から研究開発を行っている。

プラットフォームの観点では、ネットワークコンテキスト、端末コンテキスト、時空間コンテキスト、インタラクションコンテキスト、教育学習コンテキストに対応できるコンテキスト管理機能を考案し設計開発を進めている。具体的には、まず早期の利用展開に対応できるベースシステムとしてWebCT及びSakaiを対象にしている。また、uClassroomにはセキュアで密な



図7 CGアニメーションによる映像化例

連続性を確保することが重要である。教員と学生間の学習状況モニタリングとプライバシー確保のバランスを実現できるクライアント側におけるコンテキスト管理機能を検討した。その結果、CMS 専用リッチクライアント「uDesktop」の設計と初期システムの開発を行い、ウェブブラウザ連携、コンテキストインフラ連携、iPod 連携のための機能を確認している。これにより、通学など移動中のユビキタス学習を実現するために、モバイル PC やポッドキャストなどを活用したシナリオへの対応を目指している。

教材の観点では、今回開発する uClassroom の有用性を確認するために、実際の大学の授業で使用可能な教材を開発し、uClassroom のコンテキスト適応機能の有効性を示すことを目指している。教材については、大学によらず必要性・共通性の高い教材として、CALL (Computer-Assisted Language Learning) を利用した語学教育に焦点を当て、ユビキタス情報環境での利用を想定した教材を開発している (図7参照)。その際、文化財アーカイブなどを VR 教材として取り込み、語学教育等の場で利用できる機能を研究開発している。

ユーザインタフェースの観点では、uClassroom の端末コンテキスト適応技術として、VR コンテンツ収集システムの開発、三次元静的・動的モデルの多層解像度表現、抽象化表現技術の開発、VR コンテンツ伝送・提示方法の開発、Java ベースの QSplat プラグインの開発などを行っている (図8参照)。また、VR コンテンツ収集・伝送・提示に関する研究開発およびユビキタス学習用ウェアラブルインタフェース技術として、ユビキタス学習インタフェースを検討している。

さらに、ネットワークコンテキスト適応技術として、ネットワーク環境に依らない映像コンテンツの利用を可能とするための適応的映像圧縮・配信技術の基本アルゴリズムについて開発を行うとともに、実講義データを用いた指示動作状況の分析とモデル化を行っている (図9参照)。



図8 Javaによる階層化三次元データ提示プラグイン

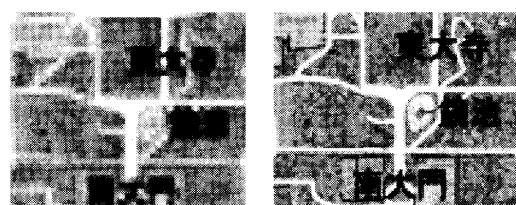


図9 画像圧縮による文字つぶれ(左)と空間解像度の維持による映像圧縮(右)

5. 結 論

本論文では、高等教育機関における教育・学習基盤ソフトウェアとしてのコース管理システムに関する要素技術を、オープンソースやオープンスタンダードの観点からソフトウェアアーキテクチャを軸に据えつつ俯瞰的に解説した。

我が国の大学が、オープンソース・オープンスタンダードという世界の潮流の中で役割を発揮し貢献していくためには、実装能力を高めることが極めて重要である。その意味で、教育・学習現場におけるニーズ・シーズを理解し、最新のテクノロジーを駆使し実装できるソフトウェア技術者の育成は不可欠である。そして、技術者・教育者・研究者が協力し、我が国独自の次世代コース管理システムの研究開発を行う体制づくりが必要である。その中で、宮川(2005)が指摘しているように、「大きな視野を持つ、さまざまな知識に豊富な、開発チームの指導者の養成」が求められる。

謝 辞

本研究は、文部科学省平成16年度～19年度「知的資産の電子的な保存・活用を支援するソフトウェア技術基盤の構築」研究開発課題「ユビキタス環境下での高等教育機関向けコース管理システム」(研究代表者：間

瀬健二) の助成を受けて実施されている。

参 考 文 献

CEAS 2007, <http://ceascom.iecs.kansai-u.ac.jp/>

CFIVE 2007, <http://cfive.itc.u-tokyo.ac.jp/>

DSpace (2007) <http://dspace.mit.edu/>

EDUCAUSE (2007) <http://www.educause.edu/>

Fedora Project (2007) <http://www.fedora.info/>

Fluid Project (2007) <http://fluidproject.org/>

Murray GOLDBERG, Sasan SALARI and Paul SWOBODA (1996) World Wide Web - Course Tool: An Environment for Building WWW-Based Courses, Fifth International World Wide Web Conference (WWW5), http://www5conf.inria.fr/fich_html/papers/P29/Overview.html, May 6-10, 1996, Paris, France.

Kenneth C. GREEN (2006) Campus Computing 2006, The 17th National Survey of Computing and Information Technology in American Higher Education, The Campus Computing Project

IMS Global Learning Consortium, Inc. (2007), <http://www.imsproject.org/>

JA-SIG (2007), Java Architecture Special Interest Group, <http://www.ja-sig.org/>

Java Community Process (2007) <http://jcp.org/en/home/index>

梶田将司 (2004) コース管理システムの発展と我が国の高等教育機関への波及. メディア教育研究, 1(1): 85-98

梶田将司, 間瀬健二 (2006) The Sakai Foundation - 北米におけるオープンソースソフトウェアによる大学教育支援の現状と我が国の課題 - 電子情報通信学会第17回データ工学ワークショップ(DEWS2006)

Shoji KAJITA and Kenji MASE (2006) uClassroom: Expanding Awareness in Classroom To Ubiquitous Teaching and Learning, Fourth IEEE International Workshop on Wireless, Mobile and Ubiquitous Technology in Education (WMUTE2006)

LON-CAPA 2007, <http://www.lon-capa.org/>

内藤久資, 梶田将司, 小尻智子, 平野靖, 間瀬健二 (2006) 大学における統一認証基盤としての CAS とその拡張. 情報処理学会論文誌, 47(4)

New Media Consortium (2006) The Horizon Report 2006 Edition, A Collaboration between The New Media

Consortium and the EDUCAUSE Learning Initiative, http://www.nmc.org/pdf/2006_Horizon_Report.pdf

NIME (2007) メディア教育開発センター「eラーニング等の IT を活用した教育に関する調査報告書」(2006年度)

宮川繁 (2005) なぜ E-Learning プロジェクトは失敗することが多いのか. 日本教育工学会論文誌, 29(3): 181-185

OASIS (2007) <http://www.oasis-open.org/home/index.php>

Open Knowledge Initiative (2007) <http://www.okiproject.org/>

O.K.I. and Eduworks (2002) What is the Open Knowledge Initiative(TM) <http://web.mit.edu/oki/learn/papers.html>

Sakai Project (2007) <http://www.sakaiproject.org/>

Charles Severance (2007) Sakai Overview, <http://www.dr-chuck.com/talks.php?id=91>, Sakai Conference, June 12-14, 2007 Amsterdam, NL.

Sakai 2.2 Framework (2006) <http://bugs.sakaiproject.org/confluence/display/ENC/Sakai+Framework>

Shibboleth (2007) <http://shibboleth.internet2.edu/>

The RSF Framework (2007) <http://www2.caret.cam.ac.uk/rsfwiki/sakamoto>, UPKI イニシアチブ, <https://upki-portal.nii.ac.jp/>

Summary

This paper describes Course Management Systems (CMSs) from the viewpoint of Open Source and Open Standard in terms of Software Architecture. CMS is a key information infrastructure for supporting teaching and learning in higher educational institutions. The effective use of CMS requires the integration with Student Information Systems (SIS), e-Library and other information infrastructures, along with strategic planning and sustainable development. In this paper, we also introduce our latest activities to develop next-generation CMS since 2004.

KEY WORDS: COURSE MANAGEMENT SYSTEM, OPEN SOURCE, OPEN STANDARD, CONTEXT AWARENESS, INFORMATION INFRASTRUCTURE IN HIGHER EDUCATION

(Received August 13, 2007)