

大学デスクトップ: 大学における情報基盤サービスのデスクトップ環境 への拡張

梶田 将司[†] 間瀬 健二[†]

[†]名古屋大学情報連携基盤センター
〒464-8601 名古屋市千種区不老町1
E-mail: kajita@nagoya-u.jp

あらまし 本論文では、サーバサイド・デスクトップサイドが強力に連携して大学における各種情報サービスを提供するための「大学デスクトップ (Institutional Desktop)」について述べる。まず、これまで我々が取り組んできた「名古屋大学ポータル MyNU.jp」および ULAN プロジェクトの「コンテキストウェアコース管理システム uClassroom」をサービスアーキテクチャの観点から整理する。そして、eclipse Rich Client Platform による大学デスクトップの実装を前提とした場合の大学デスクトップが有するべきサービスコンポーネントを、大学ポータルと比較しながら述べる。

キーワード コース管理システム, 教育学習支援システム, 大学情報基盤, リッチクライアント, オープンソース, eclipse リッチクライアントプラットフォーム

Institutional Desktop: Expanding Institutional Information Infrastructure Services onto Desktop Environment

Shoji KAJITA[†] and Kenji MASE[†]

[†] Information Technology Center, Nagoya University
Furo-cho 1, Chikusa-ku, Nagoya, 464-8601 Japan
E-mail: kajita@nagoya-u.jp

Abstract This paper describes Institutional Desktop to provide information services on campus using highly-coupled server-side and desktop-side information infrastructure services. Firstly, we summarize our activities related to Nagoya University Portal and Ubiquitous Classroom, along with the service architecture. Then, we describe service components on eclipse Rich Client Platform based Institutional Desktop to realize the paradigm shift from Web browser based to Rich Client based paradigm, comparing with service components of Nagoya University Portal.

Key words Course Management System, Teaching and Learning Support System, Institutional Information Infrastructure, Rich Client, Open Source, eclipse Rich Client Platform

1. はじめに

近年の情報通信技術の発達により、ギガビットネットワークによるキャンパスネットワークや、UNIX, Windows, Macintoshなどで構成される大規模分散教育支援システム, 全学的な無線LAN など, 大学における教育研究活動において情報通信技術の活用は必要不可欠なものになっている。これらを活用したアプリケーションとして, 大学・学部・学科などのウェブページや, 履修登録・シラバスなどの教務システム, コース管理システムに代表される教育学習支援システムなど, 様々なサービス・コンテンツがウェブベースのアプリケーションとして構築され

ている。しかしながら, これらのシステムは, サービスまたはコンテンツを提供する学内の組織がそれぞれの方針の下で構築しているため, 全体として統一感がなく, 最悪の場合, それぞれのシステムごとにアカウントが異なり, ユーザはシステムごとに使い分けなければならないという状況さえ生まれ始めている。これは, それぞれの情報システムが, 情報発信者の視点から構築されているためであり, 情報発信者の権益を守りつつ, ユーザの視点に立ったシステムの統合が必要になってきている。

このような背景の下, 我々は, JA-SIG (Java Architecture Special Interest Group) が開発した uPortal [1] を用いた大学ポータルの構築を 2002 年度から進めており, 正式稼働した

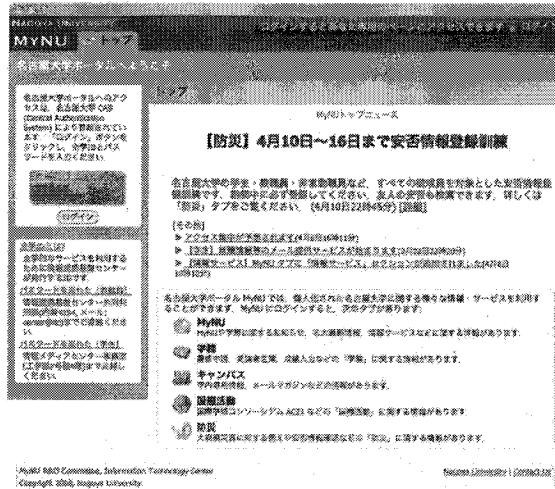


図1 第3世代の名古屋大学ポータルトップ画面。

2004年度からは、JA-SIGのCAS (Central Authentication Service) をベースにシングルサインオンを実現し、教務システムによるウェブスペース履修登録や成績入力、教員プロフィールデータベースなどとの連携を実現している[2], [3].

その一方で、パーソナルコンピュータ、携帯電話、ブロードバンドインターネットなど、エンドユーザ側の利用環境は劇的に改善され、「いつでも、どこでも、だれでも、どんな手段でも」情報サービスを受容可能なユビキタス環境の実現に向けて、時代は確実に進んでいる。このようなユビキタス環境の利便性を活用した新たな大学の教育・学習活動を実際の大学教育現場において推進するためには、エンドユーザ側の利用環境を適切に把握し適応した教育学習支援システムの構築が求められる。

このような情報技術の深化に対応するため、文部科学省研究委託事業「知的資産のための基盤技術」として、我々は、ユビキタスコンピューティング環境下において、教員・学生が利用する端末・環境・様態などの利用者状況(コンテキスト)を獲得・統合・解析し、また、コンテキストに適応して教材等を処理・提示することが可能な「コンテキストウェアコース管理システム uClassroom」の研究開発を行ってきた[4]. uClassroomは、コンテキストへの適応を可能とするための「コンテキストインフラ」をベースに、サーバ側では「日本語版 Sakai」[5], クライアント側では eclipse Rich Client Platform (RCP) [6] を用いた「uDesktop」で構成される。特に、uDesktopはユーザの各種デバイスの操作履歴であるインタラクションコンテキストを取得するためには必須である。また、eclipse RCPによる実装により、uDesktopは、サーバサイドで動作する Sakai との強力な連携が可能になるデスクトップアプリケーションに変貌しつつある。教育学習支援に限らず、教員の業績データのような個人のデータを大学のシステムとの間で効率よくやりとりしたり、成績情報や各種個人情報など高いセキュリティが求め

られる情報を大学の管理下にあるデスクトップ領域で扱えるようにするなど、サーバサイド・デスクトップサイドの連携は、大学が提供する情報サービスの様々なサービスにおいて将来必要になる可能性がある。このため、大学情報基盤整備の観点から、ユーザの管理下にあるデスクトップ環境においても、大学が提供する各種情報サービスにおいて共通に利用可能な情報基盤サービスの実現が不可欠である。

そこで、本論文では、サーバサイドとデスクトップサイドが強力に連携して大学における各種情報サービスを提供するための情報基盤システムとして「大学デスクトップ (Institutional Desktop)」を提案する。まず、これまで我々が取り組んできた「名古屋大学ポータル MyNU.jp」および ULAN プロジェクトの「コンテキストウェアコース管理システム uClassroom」をサービスアーキテクチャの観点から整理する。そして、ウェブブラウザをベースとしたパラダイムからリッチクライアントをベースとするパラダイムへ転換を図る際に求められる大学デスクトップのサービスコンポーネントについて述べる。

2. 大学ポータルにおけるサービススタック

大学ポータルは、学内に散在する情報システム・情報資源を集約し、ユーザごとに適切な情報やサービスを提供することを可能にする。しかしながら、「どう集約・統合するか」は、既存システムの機能や、システム運用ポリシー、費用対効果、各組織の業務内容など、様々な要因が複雑に絡むため、容易ではなく、様々な統合レベルが考えられる[2].

ここでは、大学ポータルの構築のために必要なサービスコンポーネントを「内部ネットワーク」「サーバ」「ディレクトリ」「シングルサインオン」「データベース」「ソフトウェア開発支援」「ポータルユーザインタフェース」に分けて整理し(図2参照)、名古屋大学ポータルでの利用例を述べる。

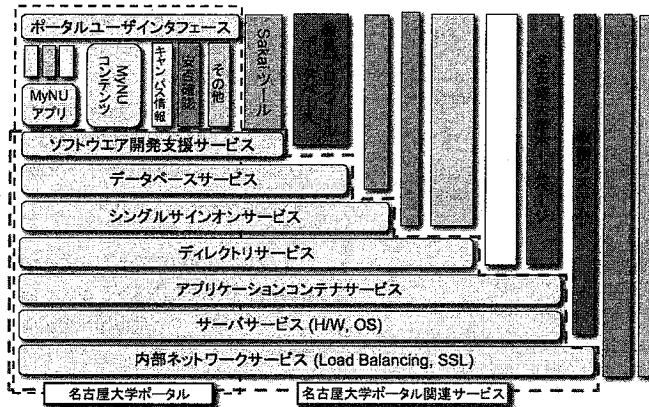


図 2 名古屋大学ポータルに関連するサービスアーキテクチャ^(注1)

2.1 内部ネットワークサービス

大学ポータルでは、数千から数万のユーザを対象としたサービスを行うため、ポータルを通じたサービスを行うシステムの前段には、負荷分散 (Load Balancing) 装置が必要となる。必要に応じてパケットフィルタリング機能やルーティング機能、IP アドレスの仮想化機能を用いて大学ポータルの内部ネットワークを構築することにより、レイヤー 3 レベルでのセキュリティを統合化することも可能である。また、SSL の暗号化・復号化も有する負荷分散装置を使用することにより、ポータルを通じたサービスを行う各システムでの SSL 暗号化・復号化処理を軽減することができる。名古屋大学ポータルでは、Nortel Networks 社製 Alteon 2424-SSL を用いている。

2.2 サーバサービス

OS レベルでのユーザ管理・セキュリティ対策・アップグレードなどのサーバ管理は、ポータルを通じたサービスを行うシステム間で共通化することにより、サーバ管理業務を軽減することが可能になる。名古屋大学ポータルでは、Sun Microsystems 社製 SunFire V210 や V240 上で、Solaris 9 または Solaris 10 を利用して共通化を図っている。

2.3 アプリケーションコンテナサービス

大学ポータルを通じたサービスは通常ウェブベースとなるため、まず、ウェブサーバ機能が必要である。また、Java Servlet を用いる場合は、Servlet コンテナとしてのアプリケーションサーバ機能が、PHP や Perl, Ruby などのソフトウェア開発言語として使用する場合は、それらのサーバサイドでの実行機能が必要となる。名古屋大学ポータルでは、Apache httpd および Tomcat, mod_php を用いている。

2.4 ディレクトリ・シングルサインオンサービス

ユーザ認証やユーザの属性情報を提供する「ディレクトリサーバ」、シングルサインオン (Single Sign On, SSO) を実現するための「SSO サーバ」など、いわゆる「アイデンティティマネジメント」に関するミドルウェア機能も、ポータルを通じたサービスを行うシステム間で共通化することにより、ユーザ管理コストを削減することができる。名古屋大学ポータル

では、ディレクトリとして Sun Microsystems 社製 Sun Java Directory Server を、SSO として JA-SIG の GAS を独自に拡張した CAS² を用いている [3]。

2.5 データベースサービス

ウェブベースのアプリケーションではデータ永続化が必須であり、各種データの格納場所としてのデータベースサーバ機能も必要となる。これらのミドルウェア機能は、ポータルを通じたサービスを行うシステム間で共通することにより、管理コストを削減することができる。名古屋大学ポータルでは、Oracle 社製 Oracle 10g Real Application Cluster を 2 ノード構成で用いている。

2.6 ソフトウェア開発支援サービス

ソフトウェア開発言語としては、Java Servlet, PHP, Perl, Ruby などさまざまなものが考えられるとともに、Java での Spring MVC, Struts, Hibernate, JSF, Sakai Framework, uPortal Framework, また、Ruby での Ruby on Rails など、それぞれの言語においても様々なソフトウェア開発フレームワークがある。名古屋大学ポータルでは、Java ベースの開発環境の整備を行っている。

2.7 ポータルユーザインタフェースサービス

大学ポータルとして、最終的にエンドユーザの目に触れる部分となるポータルユーザインタフェースについて、「見た目のデザイン」「ナビゲーション方法」「カスタマイゼーションによる個人化」「マルチデバイス対応」などを提供する。ポータルユーザインタフェースレベルでの統合を支援する「ポータルフレームワーク」としては、名古屋大学ポータルでは JA-SIG の uPortal を用いている。

3. コンテキストウェアコース管理システムでのデスクトップサイドへの拡張

「コンテキストウェアコース管理システム uClassroom」は、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、ユーザが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、ユーザがシステムにアクセス

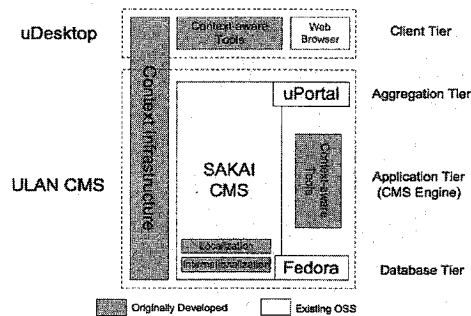


図3 uClassroom のアーキテクチャ。

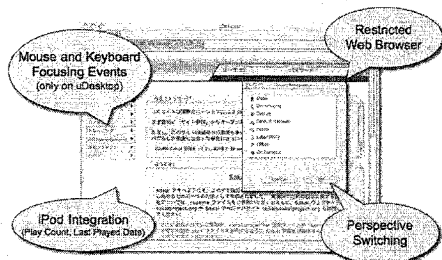


図4 eclipse RCP ベースの uDesktop。

する場所や時間に関する「時空間」コンテキスト、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、教育学習コンテキストの5つのコンテキストへの適応を可能とするための「コンテキストインフラ」をベースに、サーバ側では「日本語版 Sakai」、クライアント側では eclipse RCP を用いた「uDesktop」で構成される。そして、サーバ側あるいはクライアント側において、コンテキストに適用したサービスを提供するためのツールが構築される(図3参照)

uDesktop は、eclipse RCP を用いたクライアントサイドアプリケーションで、クライアント側でコンテキスト情報を取得したり、サーバ側に送付したり、適応的な処理を行う(図4参照)。uDesktop はウェブブラウザ機能も有し、既存のウェブベースの CMS の機能に加えてコンテキストウェアな機能が追加されている。

一方、ULAN CMS は、サーバサイドのアプリケーションで、Sakai CMS とコンテキストインフラとが連携してコンテキストウェアな機能を実現する(図3参照)。サーバ側のコンテキストインフラの主機能は「コンテキストリポジトリ」であり、サーバ側あるいはクライアント側のツールに対してコンテキスト情報の取得や検索、保存機能を提供する。

4. 大学デスクトップによる次世代大学情報基盤の構築

ULAN プロジェクトで開発している uDesktop は、「次世代

コース管理システムのためのコンテキストの取得」および「コンテキストウェアなデスクトップツールの開発」がその主要な開発動機であるが、大学における情報基盤整備という観点で考えた場合、教育学習支援に限らず、大学が提供する情報サービスのすべてにおいて将来必要になる可能性がある。そこで、uDesktop のようなデスクトップ環境における大学情報基盤としての情報基盤システムを「大学デスクトップ」と呼ぶこととする。

ここでは、大学デスクトップがもたらすであろう特長とそれに付随する課題を述べるとともに、現在利用可能なリッチクライアントソフトウェアの開発手法について整理する^(注3)。そして、リッチクライアントソフトウェアとしての大学デスクトップが有すべきサービスコンポーネントを、大学ポータルと比較しながら述べる。

4.1 リッチクライアント化による期待

NCSA Mosaic を出発点に、Netscape や Internet Explorer などの普及を通じてウェブブラウザが広く利用されるようになった結果、現在では多くの大学情報システムがウェブベースで開発されるようになってきている。しかしながら、「データ更新の度にウェブサーバとの間で通信が発生する」「OS ネイティブなデスクトップアプリケーションと比べ操作感が悪い」「ネットワーク接続が前提となっておりオフラインでの利用ができない」など、問題点が指摘され始めている。これに対応するため、4年ほど前からウェブベースのアプリケーションの機能を強化する「リッチクライアント」と呼ばれるアプローチが考えられはじめています。

大学を対象としたアプリケーション開発において、リッチクライアント化により期待できる機能についてまとめると次の通りである:

- ウェブによる技術的制約に縛られない操作感の良いアプリケーション開発が可能になる。
- オフライン時でも利用可能な状態を実現することができる。
- ITMS - iTunes - iPod に見られるような「サーバ - デスクトップ - モバイルデバイス」間の連携によるサービス提供ができる。
- クライアントサイドに設置されている実験装置などと接続することにより、より高度な実習型の教育が行える。
- シミュレーション教材のような CPU 負荷が発生する教材をクライアントサイドで実行することによりサーバサイドの負荷を削減することができる。
- デスクトップ上で起動されているローカルなアプリケーションとの連携が可能になる。
- 大学側でコントロール可能な領域を用意し、セキュリティレベルを確保しながら、教材や成績等の機密性の高いデータをクライアントサイドに保存できる。
- リッチクライアントを「デスクトップサイドでの大学情報サービスのための開発プラットフォーム」と考え、機能強化

(注3): 多くは、文献 [7], [8] での議論に基づく。

のためのフレームワークの提供を行うことにより効率の良い開発環境を整えることができる。

これらの他、実際の教育現場で広く利用されるためには、次の事項も考慮する必要がある。

- 既存の多くの大学情報システムはウェブベースであり、既存のウェブブラウザの機能が包含されている必要がある。
- 大学情報サービスが対象とする利用者は各大学の教員・学生であるため、数千～数万人が対象となる。そのため、インストールや機能更新ができるだけ簡単である必要がある。
- 利用者が使用する端末は、Windows, Macintosh, UNIX系 OS など多岐にわたる可能性があるため、できるだけ多くの OS をサポートする必要がある。

このように、リッチクライアント化による様々な期待と課題が挙げられるが、現時点での実現可能性や将来性を鑑み、リッチクライアント開発手法については慎重に選ぶ必要がある。

4.2 リッチクライアント開発手法

リッチクライアントの開発に関しては様々なアプローチが利用可能であるが、ここでは現在の動向から選んだ5つの開発手法について述べる。

まず、ウェブブラウザベースのリッチクライアント開発手法としては、JavaScript の XMLHttpRequest を駆使し、非同期にサーバとの間でデータ通信を行うことによりユーザエクスペリエンスを上げることを目指す「Ajax (Asynchronous JavaScript and XML)」ベースのものが挙げられる。また、1997年から利用され始めた、高度なアニメーションを作成可能な Macromedia Flash をベースとしたものとして「Macromedia Flex」や「OpenLaszlo」がある。JavaScript の互換性や Flash プラグインの必要性という問題は若干あるものの、Ajax ベース・Flash ベースいずれにしてもブラウザに依存しないソリューションとみなせる。さらに、ブラウザに依存するものの、クライアントサイドの機能拡張を行いやすいソリューションとして「XUL アプリケーション」がある。

一方、ウェブブラウザベースではないリッチクライアント開発手法としては、Java の開発環境として知られる eclipse Foundation の「eclipse Rich Client Platform (RCP)」ベースのものが挙げられる。

4.2.1 Ajax

通常、Web サーバ側との通信はリンクのクリック等のユーザのアクションに応じて同期的に行われるのに対して、Ajax ではユーザの要求とは非同期に Web サーバ側との通信を行うことで、必要になるであろうデータを取得したり処理を行い、ユーザの使用感を向上させようというものである。例えば、Ajax の有効性が広く認められるきっかけとなった Google Map では、要求された地点の地図だけでなく、非同期通信によりその周辺の地図をあらかじめ取得しておくことにより、滑らかな地図の切り替えを実現している。

4.2.2 Macromedia Flex

Flex は「Macromedia Flash の SSI (Server Side Include) 版」である。つまり、MXML (Macromedia Flex Markup Language) により記述されたコンテンツがサーバ側で解釈され、

SWF ファイルにコンパイルされた Flash コンテンツを Web ブラウザが Flash プラグインにより表示する [9]。

4.2.3 OpenLaszlo

OpenLaszlo は、オープンソース (CPL) で公開されている Flex と捉えることができる。コンテンツ開発者は LZX 言語や JavaScript を用いてコンテンツを作成する [10]。

4.2.4 XUL

XUL (XML based User Layout) は、Mozilla ブラウザを開発するために作成されたもので、XUL アプリケーションは XUL, JavaScript および XPCOM (Cross Platform Component Object Model) により構成される。Mozilla をはじめ、Firefox, Thunderbird など XUL アプリケーションである [11]。

4.2.5 eclipse RCP

eclipse は Java の統合開発環境 (Integrated Development Environment, IDE) として広く使われているが、eclipse 3.0 からはクロスプラットフォームを強く意識した SWT (Standard Widget Toolkit) ベースのアプリケーション開発環境として機能強化が行われている [6]。eclipse 自体、Rich Client Platform 上に構築されており、テンプレートに従った開発も可能となっている。

4.3 eclipse RCP による大学デスクトップの構築

上述のように、現在の動向から見てもいくつかのリッチクライアント開発手法が挙げられるが、uClassroom での uDesktop 実装において、我々は次の観点から eclipse RCP による開発手法を選択している：

- eclipse では、Java Swing と比較して OS ネイティブアプリケーションと同等の応答速度やルック & フィールが得られる SWT (Standard Widget Toolkit) が開発され利用されている。
- SWT 自体、Windows 系 OS, Unix 系 OS, Macintosh など、クロスプラットフォームでの利用が可能である。
- 既存のウェブベースアプリケーションとの連携を行うためのウェブブラウザ機能が、Browser コンポーネントとして利用可能である。
- ULAN CMS で開発するサーバサイドアプリケーションは Java ベースで開発しており、クライアントサイドの開発環境も Java に統一することで開発コストの増大を抑えられる。
- JavaWebStart や Eclipse 3.0 から導入された Update 機能により、作成した RCP アプリケーションの配布や更新が比較的簡単に可能である。

4.4 eclipse RCP による大学デスクトップの実装と求められる情報基盤サービス

uDesktop と同じ理由により、大学デスクトップの実装についても eclipse RCP の利用を前提としたとき、求められる情報基盤サービスコンポーネントとして、「アイデンティティ」「データベース」「ソフトウェア開発支援」および「デスクトップユーザインタフェース」に関するものが必要である^(注4) (図5参照)。

(注4)：「PC ハードウェア」や「OS」についても、「大学仕様」のノートパソコン

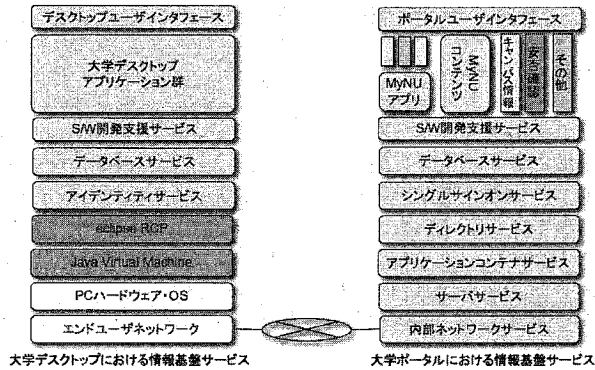


図5 大学デスクトップにおいて求められる大学情報基盤サービス。

4.4.1 アイデンティティサービス

他のサーバサイドアプリケーションの利用時に再度ユーザ認証が必要とならないよう、サーバサイド側で提供されているシングルサインオンサービスとの連携による大学デスクトップへのログインが必要である。また、アイデンティティ確立後は、プライバシーやセキュリティ上問題にならないよう、大学デスクトップが使用するリソースが、不必要にローカルマシンに漏洩しないようにしなければならない。

4.4.2 データベースサービス

大学デスクトップアプリケーションに対して、ファイルを含む各種データの格納場所を提供するデータベースサーバ機能も必要となる。また、サーバサイドアプリケーションとの連携や使用する端末が使用時ごとに異なる場合に対応するため、サーバサイドデータベースとの同期機能も必要である。

4.4.3 ソフトウェア開発支援サービス

eclipse RCP を利用するため、ソフトウェア開発言語としては Java を利用することになる。サーバサイドアプリケーションを Java で開発する場合と同じ支援サービスを提供することで、サーバサイド・デスクトップサイドに関係なく、共通の開発支援を行うことができる。

4.4.4 デスクトップユーザインタフェース

ポータルユーザインタフェースと同様に、「見た目のデザイン」「ナビゲーション方法」「カスタマイゼーションによる個人化」などを提供するが、「見た目のデザイン」や「ナビゲーション方法」については、eclipse RCP の UI コンポーネントにより行われる。サーバサイドの「ポータルフレームワーク」である uPortal のような、大学デスクトップ用の UI フレームワークも必要であろう。

5. まとめと今後の課題

本論文では、情報基盤システムとしての「大学デスクトップ (Institutional Desktop)」を提案し、ウェブブラウザをベース

の販売など、大学が共通サービスとして提供する場合もあるが、ここではアプリケーションソフトウェアとしての大学デスクトップを対象とするため、考慮の対象から外した。

としたパラダイムからリッチクライアントをベースとするパラダイムへ転換を図る際に求められる大学デスクトップのサービスコンポーネントについて述べた。

今後は、すでに開発を行っている uDesktop をベースに、本論文で述べた大学デスクトップに求められる機能を実装していきたいと考えている。

謝 辞

本研究の一部は、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。

文 献

- [1] Java Architecture Special Interest Group, <http://www.java-sig.org/>
- [2] 梶田将司, 内藤久資, 平野靖, 瀬川午直, 小尻智子, 間瀬健二, “名古屋大学ポータルによる情報サービスの統合と課題”, 情報処理学会研究報告 (分散システム/インターネット運用技術), Vol. 2007, No. 72, pp.1-6 (2007.7)
- [3] 内藤久資, 梶田将司, 小尻智子, 平野靖, 間瀬健二, “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, Vol. 47, pp.1127-1135 (2006.4).
- [4] Shoji Kajita and Kenji Mase, uClassroom: Expanding Awareness in Classroom To Ubiquitous Teaching and Learning, Fourth IEEE International Workshop on Wireless, Mobile and Ubiquitous Technology in Education (WMUTE2006), pp.161-163, November 16-17, 2006, Athens, Greece (2006.11).
- [5] Sakai Foundation, <http://www.sakaiproject.org/>
- [6] Eclipse Foundation Web Page, <http://www.eclipse.org/>
- [7] 梶田将司, 間瀬健二, “Eclipse Rich Client Platform を用いたクライアントサイドでの WSRP チャネルアグリゲーション”, 情報処理学会教育学習支援情報システム研究会, pp.70-75, 2005.12.
- [8] Shoji Kajita, “Aggregation of WSRP channels on Eclipse Rich Client Platform”, JA-SIG Winter Conference 2005, Austin, U.S.A., 2005.12.
- [9] Macromedia Flex, <http://www.macromedia.com/jp/software/flex/>
- [10] OpenLaszlo, <http://www.openlaszlo.org/>
- [11] XUL Planet, <http://www.xulplanet.com/>