

## Web アプリケーション開発技術者のための教育体系

Curriculums for Web Application Developers

新 谷 ひ ろ み , 芹 生 朋 子 , 西 文 朗

**要 約** 近年インターネット, イン트라ネット, エクストラネット上で稼働するアプリケーションの開発技術 (Web アプリケーション開発技術) が急速かつ劇的に進歩した。多くの技術が氾濫したため, 学習者の中には何を学習すればよいのか迷う人も多くなっている。教育コースを提供する際, 単に教育コースを提示するだけでなく, 受講者が学習すべき一連の技術を整理し見極めやすくする必要がある。従来からも教育コースの体系化はおこなわれていたが, この分野では一層その必要性がでてきていると考える。そこで, 筆者等は Web アプリケーション開発技術を六つの基本型に分類し, 型毎に要素技術を体系的に学習できるカリキュラムを用意した。これにより, 受講者は受講する教育コースの選択が容易になる。

さらに, その中の一つの基本型をパッケージコース化し, コースの最後に, それまでに学習した技術のすべてを駆使する必要がある総合的な演習コースを配置した。これにより, 受講者は Web アプリケーション開発に必要な技術を体系的・総合的に習得できるようになり, 学習効果は大きく向上する。

**Abstract** Recently the "technology for Web applications development" has made rapid and remarkable progress. It facilitates users to create the Web pages linked to databases build client/server systems on Web pages, or develop Web applications running in cooperation with the mission critical systems. Consequently those who create Web pages now require numerous technologies, including script languages and databases linkage, to "dynamically" create the Web pages, as well as the simple HTML technology to display text and images on the client screen.

This situation has brought that many developers have been perplexed to appreciate which technologies to study among various Web technologies. So we have classified these technologies into six categories, and prepared a curriculum for each category to facilitate to choose the set of technologies to learn.

Further we have offered one of these curriculums as a packaged course. At the end of the course a comprehensive seminar, where students practice the Web pages implementation, is arranged.

This has made it possible that the students systematically acquire the knowledge required for developing Web applications through the practice. And consequently the rapid improvement in the efficiency of learning has been achieved. We do not know any curriculum elsewhere based on such a classification as described in this paper.

### 1. はじめに

日本のインターネットは, 黎明期から普及の時代に入り, 全ての人の為のインフラストラクチャを目指して発展を続けている<sup>[1]</sup>。ある調査では, 日本のインターネットの利用者数は 1998 年初頭に 1000 万人を超えたとの報告もある (図 1)<sup>[2]</sup>。中小企業などでは, まだ経営上の必要性を感じていない経営者が多いこと等もあり導入が進んでいないが, 普及に向けて数々の施策が試みられており, 早い時期に普及が進むものと思われる<sup>[3]</sup>。

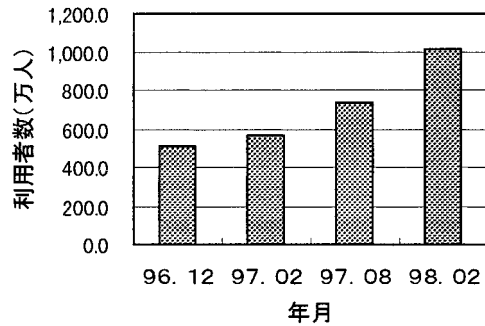


図1 インターネット利用者数の推移  
© Access Media International, 1998

利用目的は、仕事の上では情報収集、商取引、連絡相談に使用している人が多く、期待通りの効果があったとしている。プライベートな面では、趣味、ニュース、企業、観光、行政、学術等の情報収集やコンピュータ関連商品、食料品、本・雑誌・文具、衣料品等のショッピングに主に利用されている。今後、仕事面においてもプライベート面においてもさらに利用されるものと思われる<sup>[4]</sup>。

これに伴い、情報を提供するサーバの数も世界的に激増している(図2)<sup>[4]</sup>。図では、全世界のサーバの指数関数的な著しい増加率に比べ、1998年までの日本のサーバの増加率は直線的にみえる。このことは、日本に特に特殊な事情が無く世界と同じ発展の道を辿るとすれば、今後日本のサーバが劇的に増加することを示唆している。

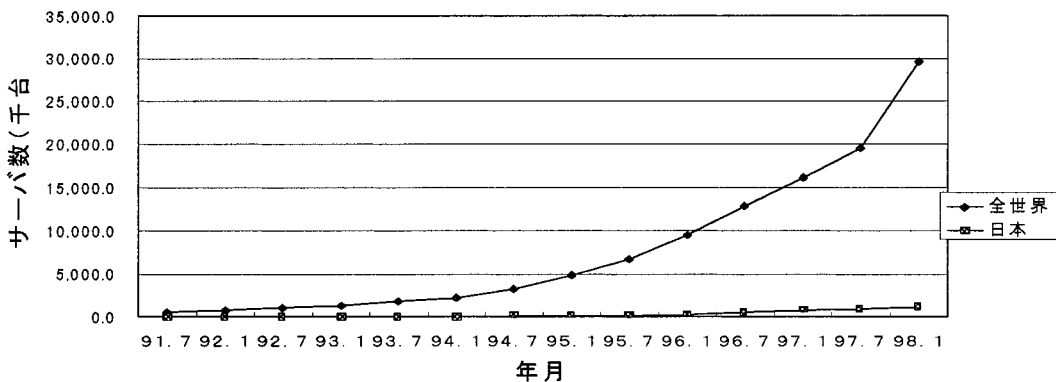


図2 インターネットに接続されているサーバの数の推移  
Network Wizards (http://www.nw.com/) により作成

さらに、インターネット技術の利用分野は、単なる情報発信やショッピングだけでなく、イントラネット・システムやエクストラネット・システムに広がっており、既に数多くのシステムが稼働している<sup>[5][6][7]</sup>。電子商取引( EC: Electronic Commerce )等も具体化しつつある<sup>[8][9]</sup>。電子商取引市場は、米国のみならず日本においても大きなマーケットになると予想されている。通産省の調査では、現在の市場規模に比べ2003年には、一般消費者向け電子商取引で約50倍(1998年650億円 2003年3.16

兆円), 企業間電子商取引で約 8 倍 (1998 年 8.62 兆円 2003 年 68.4 兆円) の予測がされている。このことは, サーバ技術も, 単に HTML を知っていればよかった時代から, より高度な技術を必要とする時代にはいったことを意味する。サーバでアプリケーションを構築するためには, データベース連携技術, 従来の基幹業務との連携技術, セキュリティ技術, 暗号化技術等より高度な技術を総合的に習得する必要が生じてきた。Web による 3 階層クライアントサーバ・システム構築技術や Cookie<sup>\*1</sup> によるセッション管理技術, Java<sup>\*2</sup>, ASP<sup>\*3</sup> 等の技術も登場している。従来の HTML や C 言語, Perl<sup>\*4</sup> 以外に学ぶべき技術が増えた。これらの各技術に対応したコースを個別に提供するのには, 複雑化するアプリケーションを効果的, 体系的に構築するためには不十分である。本稿では, ユーザの目指すシステム形態に対応した技術を身に付けるための, 体系的なコースの構成について論ずる。

## 2. Web アプリケーション開発に必要な技術とカリキュラム

### 2.1 Web アプリケーション開発技術の分類の背景と目的

1 章で述べたような状況が, HTML を使って “静的に” 情報発信をするホームページを作成する比較的初歩的な技術者から, 高度なインターネット・アプリケーションを構築する技術を有する高度な技術者まで, 早急にかつ大量に Web アプリケーション開発技術者を養成する必要を生じさせつつある。弊社のマーケット環境も例外ではなく, この様な状況にあるか, または今後直面することになると思われる。技術者の養成にあたっては要請される技術が多岐にわたるため, 独学・自習よりは, 系統立てた教育を受ける方がより効果的である。

数多くの技術の中から学習すべき技術を容易に見極められるように, コースが整理・体系化されている必要がある。多岐にわたる数多くの技術の中で, 何を学習すべきか悩んでいるユーザが多く, 何を勉強したら良いのかの相談を受けることがある。どのような Web アプリケーションを開発したいかはわかっているが, 実現の手段としてどのようなツール/言語を使用すればよいかを決めかねているユーザも少なくない。特に, 自力で Web サイトを立ち上げようとしているエンド・ユーザの方に多い。

教育機関としては, 単に必要とするプロダクト毎の知識を習得するためのメニューを示すのではなく, ユーザが目的としているシステム形態に要求される知識の獲得に焦点を絞るべきであると考え。本稿では, Web アプリケーション構築に焦点を当て, ユーザが目指すシステム形態の分類から, 必要な技術や受講すべき技術を明らかにすることを試みる。筆者等は, システム形態と文献<sup>10)</sup>を参考にして, 6 種類の基本型を定義した。6 種類の基本型には, 幾つかの技術を関連付けることができ, 基本型を定義することで, 受講者が抱えている開発対象のシステム形態との対応付けが可能になる。表 1 に, 基本型とシステム形態との対応を示す。この形態と 6 種類の基本型とを照らし合わせることにより, 受講者が学習すべき技術を見極めることが可能になる。

### 2.2 Web アプリケーションの分類と必要な技術

各基本型で必要となる技術項目を整理すると, 以下のようになる。実際のアプリケーションでは, これらの基本型の幾つかが混合使用される。(なお, 各技術について

表 1 Web アプリケーションの基本型と形態の対応

基本型	システム形態
HTML 型	①静的情報発信形態 予め用意した文書やイメージをそのままクライアントに表示する。
簡易データベース連携型	①動的情報発信形態 データベースに保存されたデータを基に動的に表示内容を変える。 ②データエントリ形態 クライアントから入力されたデータを単純にサーバのデータベースに保存する。
ソフトウェア部品型	①クライアント・プログラム形態 クライアントのみで完結する業務。 ②2 階層クライアントサーバ形態 クライアントからサーバのデータベースをアクセスする。
サーバ・プログラム型	③3 階層クライアントサーバ形態 アプリケーション・プログラム、データベース共にサーバ側にある。
基幹業務連携型	①基幹システム開放形態 基幹業務を組織内外に Web を通して開放する。または、Web 上のアプリケーションと基幹業務の間でデータの交換をする。
クライアント・スクリプト型	①入力チェック等 HTML に若干の機能を追加する補助的な型。

は略語等を記述するに止めている。巻末を参照願いたい。)

#### 1) HTML 型

① 概要：サーバから送られてくる HTML ファイルの記述に従ってブラウザが画面に文字や画像を表示するシステムである。あらかじめ HTML ファイルとしてサーバに用意している内容のみを表示でき、クライアントからの入力内容を基に表示内容を動的に変えることはできない。ブラウザで HTML ファイルを解釈し表示するだけなので、実行時間は短い。任意のブラウザが使用でき、使用するブラウザを特定する必要はない。最も基本となる型である（図 3）。

② 習得項目：HTML

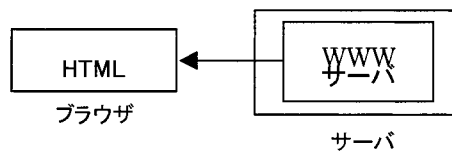


図 3 HTML 型

#### 2) ソフトウェア部品型

① 概要：HTML ファイルの他に、クライアントで実行されるソフトウェア部品（プログラム）を WWW サーバからダウンロードして実行する。クライアントのみで可能な処理であれば、サーバに負荷をかけることなく実行できる。ソフトウェア部品がクライアントにロード済みであればネットワークに負荷をかけずに済むので処理は速いが、そうでなければダウンロードのための時間がかかる（図 4）。

② 習得項目：HTML

Java (Java アプレット<sup>\*5</sup> 作成時) または VB/C++ (ActiveX コントロール<sup>\*6</sup> 作成時)

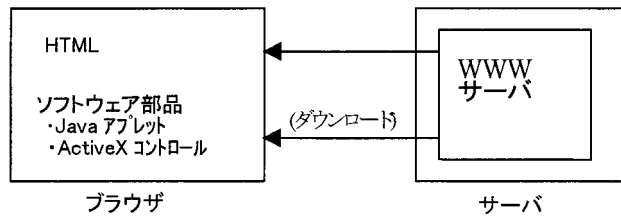


図 4 ソフトウェア部品型

### 3) クライアント・スクリプト型

① 概要：ソフトウェア部品型ではコンパイルされたプログラムが必要に応じてサーバからダウンロードされるのに対し、クライアント・スクリプト型ではスクリプト言語で記述されたプログラムが HTML 中に組み込まれて送られる。プログラムはブラウザのインタプリタにより解釈され実行される。データベース・インタフェース等はないが、ブラウザ側で入力データのチェックや計算などが行えるため、サーバやネットワークの負荷の軽減が図られる。スクリプトがクライアントに送られるため、処理内容がクライアントから見えてしまい機密性を保ちにくい(図5)。

② 習得項目：HTML

JavaScript<sup>\*7</sup> または VBScript<sup>\*8</sup>

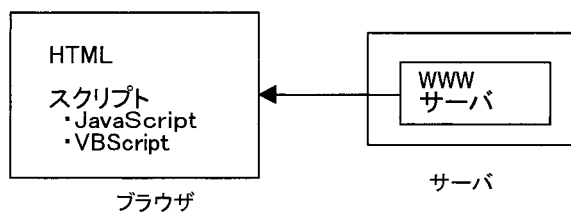


図 5 クライアント・スクリプト型

### 4) 簡易データベース連携型

① 概要：入力データを基にどのような SQL<sup>\*9</sup> を実行するのか記述したファイルと、SQL の実行結果をどのように HTML ファイルに編集するのかを記述したテンプレートを作成する。プログラミング機能はないが、極めて簡単にデータベースをアクセスできる。そのため、単純にデータベースの内容を検索表示する様なシステムでは開発工数も少なく済み、重宝がられている。IIS<sup>\*10</sup> で提供される(図6)。

② 習得項目：HTML

IDC<sup>\*11</sup>

## SQL

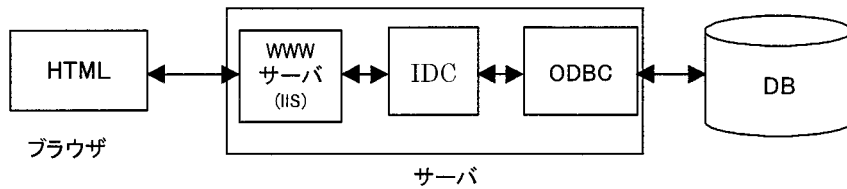


図 6 簡易データベース連携型

## 5) サーバ・プログラム型

- ① 概要：サーバ上でデータベース・アクセスを含む本格的なプログラムを記述できるため、従来のクライアントサーバ・システムに相当するシステムを構築することができる（図7）。

大きく

- ・汎用的な CGI<sup>\*12</sup> を使用する C/C++/Perl
- ・特殊なインタフェースを使用する ASP, PL/SQL<sup>\*13</sup>
- ・プラットフォームを選ばない Java

に分けられる。ただし、Perl がアクセスできるデータベースは特殊なデータベースに限定される。

この中で、CGI を使用するものは多くの OS や WWW サーバで採用されているが、サーバ・プログラム（CGI プログラム）を起動するごとにプロセスが生成されるので、ユーザの数が増えるとパフォーマンスが低下する。特殊なインタフェースを使用するものはサーバ・プログラムを起動するごとにプロセスを生成しないのでパフォーマンスに優れている。

ASP はスクリプト言語（VBScript, JavaScript）を使用するため習得・開発が比較的容易である。他のサーバ・プログラム型と同様にブラウザを選ばないが、提供される WWW サーバが IIS のみである。

Java は「Write Once, Run Anywhere」と言われ、移植性が高く、メモリ管理機構が充実していてバグを発生させにくい。

PL/SQL はデータベースとの親和性は良いが、Oracle<sup>\*14</sup> データベースに特化した言語である。OAS<sup>\*15</sup> のみで使用できる。

- ② 習得項目：HTML

SQL

C/C++/Perl(CGI を使用する場合)または VBScript/JavaScript ADO<sup>\*16</sup> (ASP を使用する場合)または Java/JDBC<sup>\*17</sup> (Java を使用する場合)または PL/SQL と関数 (PL/SQL を使用する場合)

## 6) 基幹業務連携型

- ① 概要：汎用機や UNIX<sup>\*18</sup>, 及び Windows NT<sup>\*19</sup> 等のトランザクション・システムに代表される Web アプリケーション以外のシステム（基幹業務）にイ

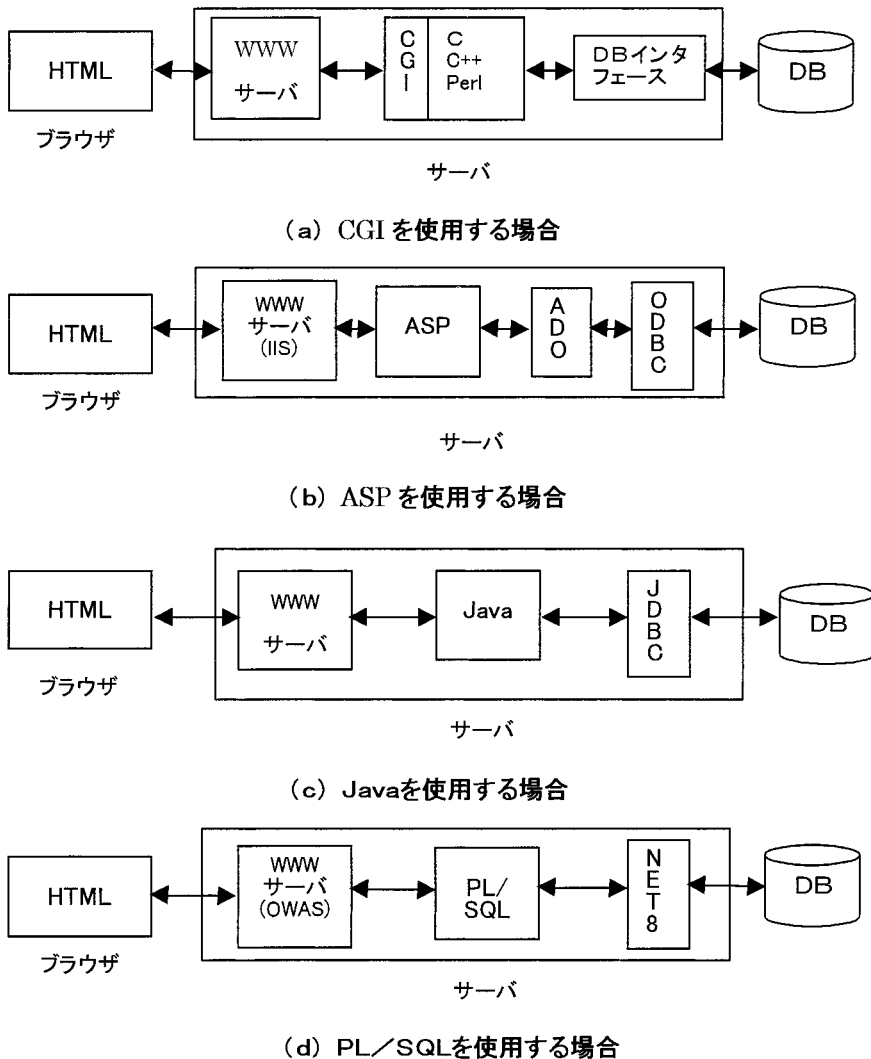


図 7 サーバ・プログラム型

インターネットを通じて外部からアクセスできる仕組みを提供する(図8)。例えば、ASPはWindows NTのMTS<sup>\*20</sup>トランザクション・システムにアクセスできる。弊社の技術のいくつかを以下に挙げる。

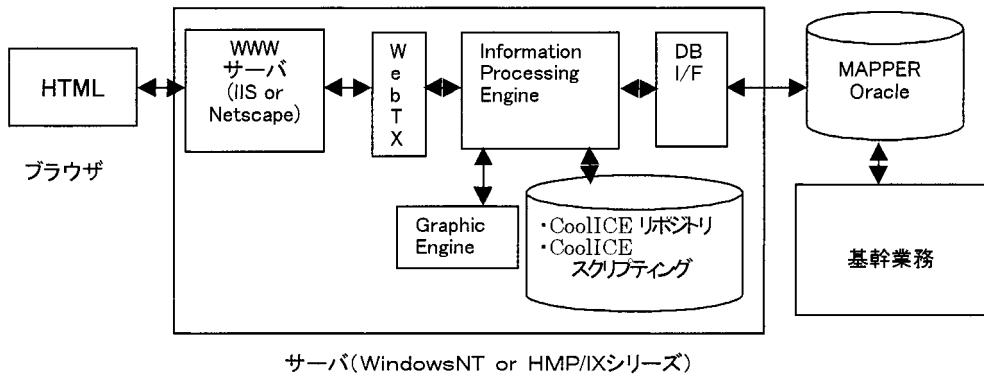
CoolICE<sup>\*21</sup>はWindows NTまたはHMP/IXシリーズ<sup>\*22</sup>上で稼働し、OracleデータベースまたはMAPPER<sup>\*23</sup>データベースを介してHMP/IXシリーズ上の基幹業務と連携する。

NX/WebCOMS<sup>\*24</sup>はHMP/NXシリーズ<sup>\*25</sup>上で稼働し、HMP/NXシリーズのLINC<sup>\*26</sup>や3GLの業務画面をWeb化する。

TransIT ODBC<sup>\*27</sup>を使用すると、CGIプログラムまたはASPから、HMP/NXシリーズのデータベースDMS II<sup>\*28</sup>やLINCにアクセスできる。

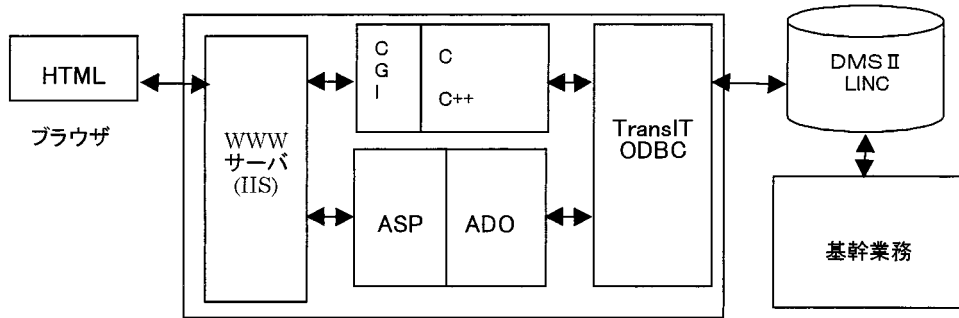
② 習得項目：CoolICEとTransIT ODBCについてのみ紹介する。

HTML, SQL, CoolICE, MAPPER, ASP (CoolICEの場合)



サーバ(WindowsNT or HMP/IXシリーズ)

(a) CoolICE の場合



サーバ(WindowsNT or HMP/NX シリーズ)

(b) TransIT ODBC の場合

図 8 基幹業務連携型

または

HTML, C/C++, ASP, ADO, ODBC<sup>\*29</sup>, DMS II/LINC(TransIT ODBC の場合)

Web アプリケーションを構築するには、上述のプログラミング技術に加え、インターネット/イントラネットの基盤技術の知識が必要である。サーバ OS, TCP/IP, SMTP<sup>\*30</sup>, DNS<sup>\*31</sup>, WWW サーバ, ファイア・ウォール<sup>\*32</sup>, DBMS 等である。分散アプリケーションを開発する場合には CORBA<sup>\*33</sup> や DCOM<sup>\*34</sup> が、セキュリティが重要なシステムの場合は暗号化・認証技術が必要になる。様々な製品が組み合わされてアプリケーションが開発されるため、Web アプリケーション開発者にはこれらの技術が横断的に必要な場合もある。この稿では、Web アプリケーション開発に必要なプログラミング技術を中心に紹介し、インフラに関連する技術については割愛した。

### 2.3 Web アプリケーション技術のためのカリキュラム構成

2.2 節で述べた各基本型に必要な技術を身に付けるためのカリキュラムを付録 1 に示した。C/C++/VB/ODBC/SQL/MAPPER/LINC 等は Web アプリケーション開発に特化された技術ではなく、その他の分野でも広く使用されている。したがっ



て、これらの技術に関する教育は従来から提供されてきた。付録1では、前述の基本型に基づき新たに用意した Web アプリケーション開発者育成のためのカリキュラムのみを掲載している。

### 3. 総合的に Web 技術を身につけるためのコース体系

第一ステップとして付録1に紹介しているカリキュラムを用意した。しかし、これらのカリキュラムの各コースを個別に学習するだけでは断片的な知識に留まる可能性が高い。カリキュラムの中の各コースで得た知識を総合的に使用して Web アプリケーションを構築する訓練が必要になる。そのためには個別の技術を習得した後に、総合的な演習コースで開発実習をするのが有効である。

そこで、基本型の中の一つである「サーバ・プログラム型」の ASP の技術取得を目的として、コースをパッケージ化した。コースの最後に総合演習コースが配されている。ASP を取り上げた理由は、付録1で紹介しているカリキュラムの中で、ASP の教育需要が我々の教育センタにおいて一番高いからである。パッケージ化した理由は、学習すべき技術をより明示的に示すと共に、多くの個別のコースを受講することにより大きくなる経済的負担の軽減を図るためである。各基本型ごとに総合的なパッケージコースを設定すべきであるが、本章では、実現した ASP に関するパッケージ・コースの試みと、その評価について述べる。

#### 3.1 ASP 関連パッケージ・コースの特徴とカリキュラム

ASP のパッケージ・コースでは、ASP だけでなく、HTML からデータベース連携技術までを一括して習得できるようにしている。ASP パッケージ・コースの特徴は次の三点である。

- ① ASP で Web アプリケーションを開発するのに必要な知識が集められており、一括して習得できる。さらに受講順序を考慮した日程および料金の面で便宜が図られている。
- ② サーバ・プログラム型は複雑なアプリケーションが多い。そのため、要素技術を個別に学習するだけでは不十分である。それらを総合的に組み合わせてアプリケーションを開発する総合演習が必要と考え、仕上げとして総合演習をパッケージ・コースの最後に配置する。
- ③ 受講者の知識に応じてパッケージコースのどのコースからでも受講してよい。

パッケージ・コースの詳細を図9と表2に示す。

このパッケージコースの一番最後で、総合演習を実施する。

#### 3.2 Web アプリケーション開発総合演習

演習コースは、ASP だけではなく、関連する技術のすべてを駆使する必要があるように設定した。結果として、パッケージの各コースで習得した技術を確実に身につけることができるようになる。

受講者には、

- ① 図10に示す画面遷移図を含む仕様書
- ② 実習環境

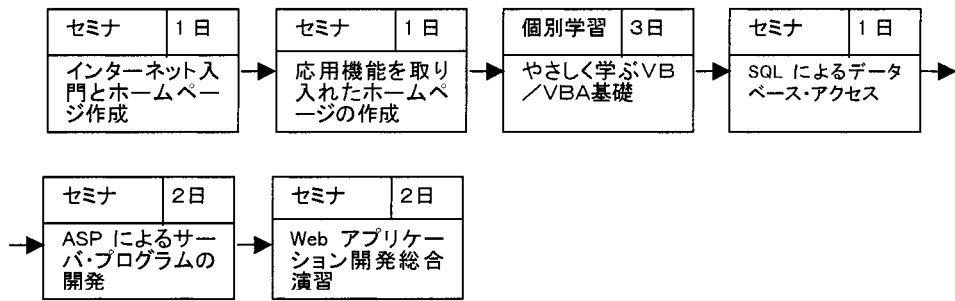


図 9 ASP 関連パッケージ・コース体系

表 2 ASP 関連パッケージ・コースの学習内容

コース	学習内容
SQLによるデータベース・アクセス	<ul style="list-style-type: none"> <li>・データベースの検索と更新</li> <li>・トランザクション</li> </ul>
ASPによるサーバ・プログラムの開発	<ul style="list-style-type: none"> <li>・VBScriptとASP</li> <li>・ユーザーから送信されたデータの取得</li> <li>・ADOを使用したデータベース・アクセス</li> <li>・電子メールの送信</li> </ul>
Webアプリケーション開発総合演習	<ul style="list-style-type: none"> <li>・図書館システムをモデルにしたWebアプリケーション開発演習</li> </ul>

(注) 「インターネット入門とホームページ作成」「応用機能を取り入れたホームページの作成」「やさしく学ぶVB/VBA基礎」の学習内容は、付録1を参照。

③ サンプルデータ

が与えられ、ASPを使用したWebアプリケーションの実装を演習する。受講者は、パッケージ内の各コースで習得した知識のすべてを駆使して実装を完成させる必要がある。この過程において受講者の知識が総合的に整理・体系化されるのがねらいである。

3.3 パッケージコースの評価

新しい試みである、基本型に従い、さらにコースの最後に総合演習を配したパッケージコースは、以下のような成果が期待できる。

- ① 個別コースで学習した内容をより確かなものにできる。
- ② 受講者自身の今後の課題（強化しなければいけない分野）が明らかになる。
- ③ コースの最後に発表の場を設けているので、様々な考え方や開発時のヒントを互いに得ることができる。

受講後のアンケートに記述される受講者の評価で、パッケージ化以前のコースの評価（表3）と、パッケージ化したコースの評価（表4）を比較した（評価は何れも4点満点）。

「ASPによるサーバ・プログラムの開発」コースの評価をパッケージ化する前とパッケージ化した後で比較すると、総合評価で3.0から4.0に向上している。また、

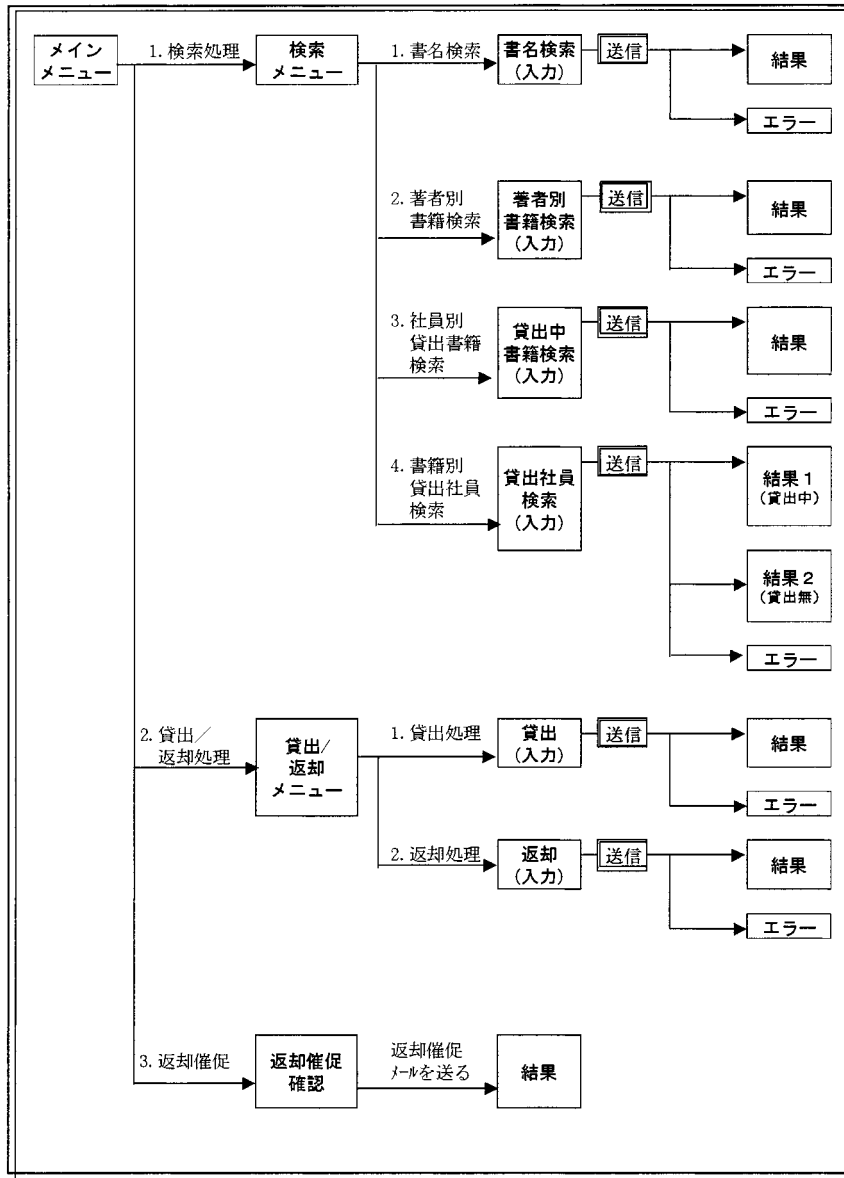


図 10 「Web アプリケーション開発総合演習」の課題 (画面遷移図)

表 3 パッケージ・コース化以前の ASP および他のセミナーの受講者の評価

コース	理解度	総合
IIS による Web ページとデータベースの連携 -IDC-	3.1	3.1
ホームページ作成のための CGI と Perl 基礎	3.0	3.0
ASP によるサーバプログラムの開発 (パッケージ・コース化以前)	3.0	3.4

表 4 パッケージ・コース化したセミナーの受講者の評価

コース	理解度	総合
ASP によるサーバプログラムの開発	3.3	4.0
Web アプリケーション開発総合演習	3.3	3.8

パッケージ・コースの総仕上げである「Web アプリケーション開発総合演習」の総合評価は 3.8 と、いずれも一般的な他のコース評価に比較して高い。理解度についても、パッケージ・コース化する前よりもパッケージ化した後のほうが、3.0 から 3.3 へと向上している。この理由は、個別技術の習得では理解しにくかったことも、総合的なコース体験で理解が深まったためだと想定される。受講者の受講所感でも、「独学ではなかなか一つのシステムを開発するというところまでできないので、今回の講座はとても良い機会だったと思います。」、「ASP 基礎,SQL 基礎と続けて受講してきた、そのまとめとして Web アプリケーション開発総合演習を受講してきたわけであるが、なかなかこのような研修スタイルは無いので珍しいと思います。印象&成果として、とても良く理解も深まったと思います。」といった肯定的な意見が見受けられた。筆者等が想定していた総合演習を伴ったパッケージコースの効果が出ていると判断される。

#### 4. ま と め

Web アプリケーション開発者には多種多様な技術が求められる。新しい技術も次々に生まれている。その中で、学ぶべき技術は何かを模索されている方も少なくない。そこで、筆者等の所属する弊社大阪の教育センタでは、セミナーを提供する際次の点に留意している。

- ① 学習すべき技術をコア技術としてわかりやすく提示する
- ② 総合的な Web アプリケーション開発技術の取得を目指す
- ③ 新技術へ迅速に対応する

そのために、Web アプリケーション開発技術を六つの型に分類し、型毎にカリキュラムを提示した。さらに、ASP 関連のカリキュラムはパッケージ・コース化し、最後に総合演習コースを配した。このパッケージ・コースでは、必要な個別技術 (HTML, SQL, ASP 等) の学習に加えて、総合演習コースで個別に取得した技術をいかにして Web アプリケーション開発に活かすかを総合的に経験・体得できる。これにより教育効果を向上させることができる。

現在のところ、総合演習までパッケージとして用意しているコースは ASP だけである。このようなパッケージ・コースは、新しく学習した技術を駆使して一つのアプリケーションを作成し総合力を養う絶好の機会となるので、他の技術についても同様に内容を検討し、総合的な知識を身に付けることのできるコースとして行く予定である。今後現れるであろう新技術についても、技術個々に教育を行うのではなく、対象とするシステム形態で必要となるコース展開の中に位置づけ、提供していきたいと考えている。

- \* 1 Cookie : ブラウザに用意されているデータ領域 . WWW サーバから読み書きできる . セキュリティ上 , 書き込まれたデータはそれを書き込んだサーバからしか読めない . WWW サーバはこの機能を使用してサーバ・プログラムのセッションの管理ができる .
- \* 2 Java : Sun Microsystems 社が発表したオブジェクト指向のプログラミング言語 . 仮想マシンを実装していれば , どのようなプラットフォームでも稼働する . Web アプリケーションのサーバ・プログラムを記述する言語としても注目されている .  
Java およびその他 Java を含む商標は米国 Sun Microsystems, Inc. の商標であり , 同社の Java ブランドの技術を使用した商品を指す .
- \* 3 ASP : Active Server Pages. IIS の元で , VBScript 等のスクリプト言語を実行する機能 . データベースへのアクセスを含む強力な言語機能を有する . また , MTS によるトランザクション・システムとの連携も ASP によって行う .
- \* 4 Perl : Practical Extraction and Report Language . もともと UNIX 上で開発され , 現在は Windows 上でも稼働するインタプリタ言語 . 強力なテキスト処理機能を有するため , HTML ファイルの編集等が容易にでき CGI プログラムを記述する言語として広く用いられている .
- \* 5 Java アプレット : Java で記述された小さなプログラム . WWW サーバからダウンロードされブラウザ上で実行される .
- \* 6 ActiveX コントロール : WWW サーバからダウンロードされブラウザ上で実行されるコンポーネント・プログラム . ブラウザに機能を追加できる . 各種 ActiveX コントロールがフリーウェアとして流通している . 主に Internet Explorer で使用できる .  
ActiveX は , 米国 Microsoft Corporation の米国およびその他の国における商標である .
- \* 7 JavaScript : Java を基にして作られたスクリプト言語 . Java よりは習得は容易 . HTML ファイルに埋め込まれ HTML ファイルとともにクライアントに送られる . ブラウザのインタプリタにより , 解釈・実行される .
- \* 8 VBScript : Visual Basic Script Edition. Visual Basic 互換のスクリプト言語 . HTML ファイルに埋め込まれ HTML ファイルとともにクライアントに送られる . ブラウザのインタプリタにより , 解釈・実行される . Internet Explorer でのみ実行できる .
- \* 9 SQL : Structured Query Language. リレーショナル・データベースをアクセスするための言語 . ISO , ANSI , JIS 等で標準化されており , 全てのリレーショナル・データベース製品で共通に使用できる .
- \* 10 IIS : Internet Information Server. Windows NT 上で稼働する WWW サーバ . ASP , IDC , CoolICE , Perl 等が IIS の元で稼働する . Microsoft 社製 .
- \* 11 IDC : Internet Database Connector. IIS の元で稼働 . ブラウザから入力されたデータに基づき SQL 文を発行してデータベースをアクセスし , 結果を HTML ファイルに展開しブラウザに送り返す . プログラミング機能はない . ブラウザからデータベースをアクセスする簡易ツール .
- \* 12 CGI : Common Gateway Interface . ブラウザ画面上に表示された入力領域 ( Form ) から入力されたデータを WWW サーバ上のプログラム ( CGI プログラム ) に渡す . また , CGI プログラムが動的に作成した HTML ファイルをブラウザに送り返し表示する . 即ち , ブラウザと WWW サーバの間にインタラクティブ機能を与える .
- \* 13 PL/SQL : Procedural Language/SQL. Oracle の拡張 SQL 言語 . SQL 文だけでなく変数や制御機能等言語機能を有する . データベースの中に保存されプログラムから呼び出して実行するストアド・プロシジャやストアド・ファンクションを作成できる . Web アプリケーション・プログラムを記述する機能も有する .
- \* 14 Oracle は ORACLE Corporation の登録商標である .
- \* 15 OWS : Oracle Application Server. Oracle 社製の WWW サーバ .
- \* 16 ADO : ActiveX Data Object . データベースをアクセスするための COM コンポーネント . COM コンポーネントとは幾つかの関数や手続きをあらかじめモジュール化したもの . Windows 上で稼働する .
- \* 17 JDBC : Java Database Connectivity. Java のデータベースインタフェース .
- \* 18 UNIX は X/Open Company Limited がライセンスしている米国ならびに他の国における登録商標である .
- \* 19 Windows NT は米国 Microsoft Corporation の米国およびその他の国における登録商標である .
- \* 20 MTS : Microsoft Transaction Server. Windows NT 上のトランザクション管理システム . Microsoft 社製 .
- \* 21 CoolICE : IIS の元で稼働する AP サーバを実現する仕組みの一つ . ASP 及び MAPPER 言語を使用してプログラミングでき , Oracle 及び MAPPER データベースをアクセスできる . UNISYS 社製 .
- \* 22 HMP/IX シリーズ : UNISYS 製汎用機の一つ .
- \* 23 MAPPER : HMP/IX シリーズ , HMP/NX シリーズ , UNIX , Windows NT 上で稼働する 4 GL システム . 独自のリレーショナル・データベースを有する .
- \* 24 NX/WebCOMS : HMP/IX シリーズ上で稼働する Web アプリケーション・サーバ . 同シリ

ーズのトランザクション・システムとのインタフェースを有する。

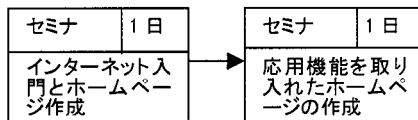
- \*25 HMP/NX シリーズ：UNISYS 製汎用機の一つ。
- \*26 LINC：HMP/NX シリーズ及び HMP/IX シリーズ上で使用できる 4 GL 開発言語。
- \*27 TransIT ODBC：HMP/NX シリーズ上のデータベースがアクセスできる Windows NT 上のデータベース・インタフェース。
- \*28 DMS II：HMP/NX シリーズで使用できるネットワーク型データベース。
- \*29 ODBC：Open Database Connectivity.Windows 上の汎用データベース・インタフェース。
- \*30 SMTP：Simple Mail Transfer Protocol。インターネットで使用されている電子メールの転送プロトコル。
- \*31 DNS：Domain Name Server。ネットワーク上のコンピュータの名前と IP アドレスの対応を保有し、名前ネットワーク上の任意のコンピュータをアクセスできる仕組みを提供する。インターネットで使用する URL にはコンピュータの名前が含まれており、URL で任意のインターネット・サイトにアクセスできるのも DNS の働きのおかげである。
- \*32 ファイア・ウォール：(企業や団体等の) 内部のネットワークに外部から侵入できないように防護をしているセキュリティ・システム。内部の LAN がインターネット等で外部と接続されている時重要な役割を持つ。
- \*33 CORBA：Common Object Request Broker Architecture。オブジェクト指向技術の標準化を推進する非営利団体 OMG (Object Management Group) によって提唱されているオブジェクト指向の分散処理環境を実現するためのアーキテクチャ。
- \*34 DCOM：Distributed Component Object Model。ネットワーク上の複数の異機種コンピュータのプロセス間で通信を可能にする仕様。Microsoft 社が提唱。

その他、本稿に記載の会社名、製品名は、各社の商標および登録商標である。なお、TM、®マークは明記していない。

### 付録1 カリキュラム一覧

#### 1) HTML 型のアプリケーションの開発者向けカリキュラム (付図1)

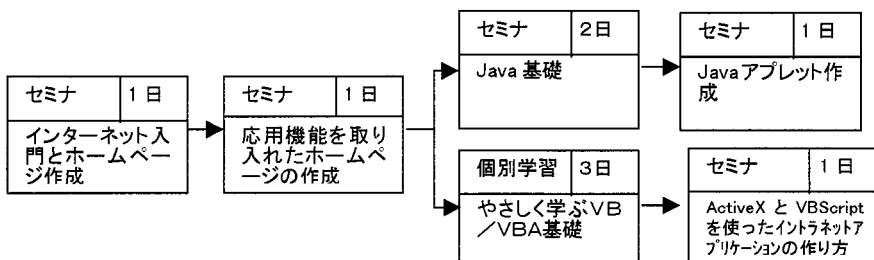
HTML ファイルを作成し文字情報や画像情報を発信するアプリケーションの開発者向けのコースであると共に、すべての型の Web アプリケーション開発に必要なである。



コース	学習内容
インターネット入門とホームページ作成	<ul style="list-style-type: none"> <li>・概要と専門用語の理解及びブラウザの操作</li> <li>・HTML エディタによるホームページの作成</li> </ul>
応用機能を取り入れたホームページの作成	<ul style="list-style-type: none"> <li>・HTML 文</li> <li>・SSI, フォーム, JavaScript</li> </ul>

付図1 HTML 型のアプリケーションの開発者向けカリキュラムと学習内容

#### 2) ソフトウェア部品型のアプリケーションの開発者向けカリキュラム (付図2)

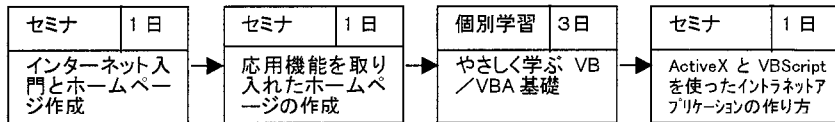


コース	学習内容
Java 基礎	<ul style="list-style-type: none"> <li>・Java 及びオブジェクト指向の概要</li> <li>・基本文法と演習</li> </ul>
Java アプレット作成	<ul style="list-style-type: none"> <li>・アプレット概要と簡単なアプレット作成</li> <li>・描画, アニメーション, GUI</li> </ul>
やさしく学ぶ VB/VBA 基礎	<ul style="list-style-type: none"> <li>・VB/VBA の基礎知識</li> </ul>
ActiveX と VBScript を使ったイントラネットアプリケーションの作り方	<ul style="list-style-type: none"> <li>・VBScript</li> <li>・ホームページへの ActiveX コントロールの組み込み</li> </ul>

(注) Java のコースについては、他に、日本サン・マイクロシステムズ(株)とのライセンス契約に基づく一連の言語/開発コースがある (東京教育センタ)。

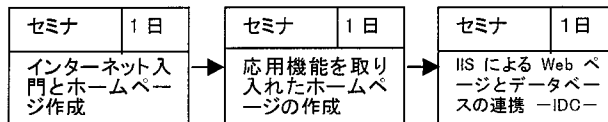
## 付図 2 ソフトウェア部品型のアプリケーションの開発者向けカリキュラムと学習内容

### 3) クライアント・スクリプト型のアプリケーション開発者向けカリキュラム (付図 3)



付図 3 クライアント・スクリプト型のアプリケーション開発者向けカリキュラム

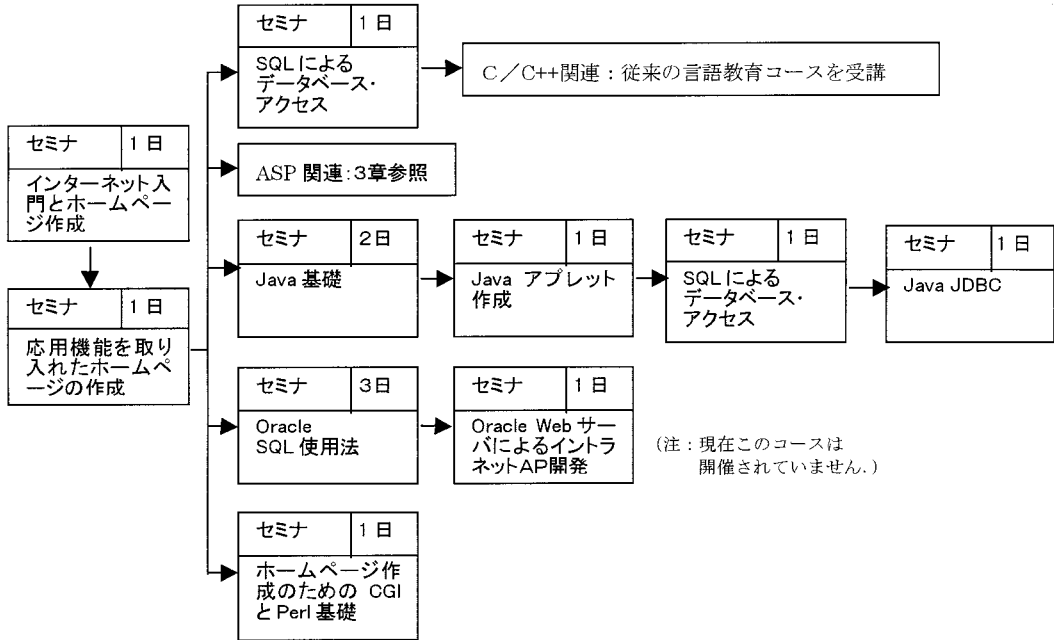
### 4) 簡易データベース連携型アプリケーション開発者向けカリキュラム (付図 4)



コース	学習内容
IIS による Web ページとデータベースの連携 -IDC-	<ul style="list-style-type: none"> <li>・IDC 概要</li> <li>・フォームと SQL</li> <li>・IDC ファイルとHTXファイルの作成</li> </ul>

付図 4 簡易データベース連携型アプリケーション開発者向けカリキュラムと学習内容

5) サーバ・プログラム型のアプリケーション開発者向けカリキュラム (付図5)

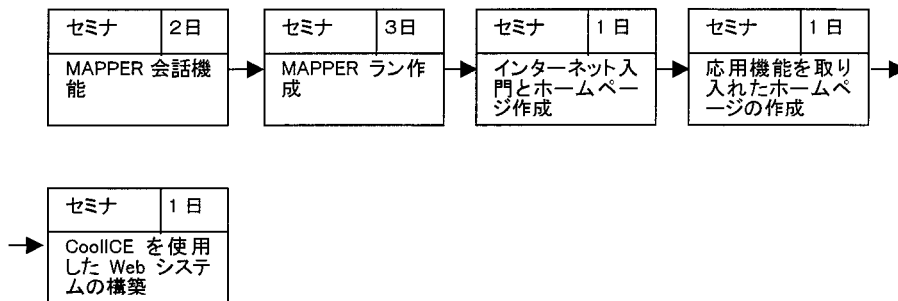


コース	学習内容
Java JDBC	<ul style="list-style-type: none"> <li>・JDBC 概要</li> <li>・JDBC を使用したアプリケーションの作成</li> <li>・JDBC を使用したアプレットの作成</li> </ul>
Oracle SQL 使用法	<ul style="list-style-type: none"> <li>・SQL</li> <li>・PL/SQL</li> </ul>
Oracle Web サーバによるイントラネットAP 開発	<ul style="list-style-type: none"> <li>・フォームと PL/SQL</li> <li>・HTTP と HTF</li> </ul>
ホームページ作成のための CGI と Perl 基礎	<ul style="list-style-type: none"> <li>・CGI とフォーム</li> <li>・簡単な文法</li> <li>・入力処理</li> </ul>
SQL によるデータベースアクセス	3.1 参照

付図 5 サーバ・プログラム型のアプリケーション開発者向けカリキュラムと学習内容



## 6) 基幹業務連携型カリキュラム (CoolICE を MAPPER で使用した場合)(付図6)



コース	学習内容
MAPPER 会話機能	・MAPPER の対話操作
MAPPER ラン作成	・MAPPER のプログラミング
CoolICE を使用した Web システムの構築	・CoolICE を使用した Web システムの構築・運用 ・既存データベースの活用

CoolICE を ORACLE で使用する場合は、ASP の学習が必要である。

付図 6 基幹業務連携型カリキュラム (CoolICE を MAPPER で使用した場合) と学習内容

- 参考文献**
- [ 1 ] 村井純, インターネット II, 岩波書店, 1998. 2
  - [ 2 ] 日本インターネット協会編, インターネット白書, インプレス, 1998. 6
  - [ 3 ] (財)日本情報処理開発協会, 情報化白書 1998, コンピュータ・エイジ社, 1998. 6
  - [ 4 ] 郵政省編, 平成 10 年版通信白書, 大蔵省印刷局, 1998. 5
  - [ 5 ] 程近智, 他, イントラネットが変える明日のオフィス, ユニシスニュース, 日本ユニシス(株) No. 435 1997. 7.
  - [ 6 ] 吉川和宏, 榊原康, インターネットでシステムを広げる, 日経オープンシステム, 日経 BP 社, No. 63 1998. 6.
  - [ 7 ] 日経 BP 社編, WWW スペースを取引先に提供し購買情報を社内に伝達 (日経データプロ・オープンプラットフォーム NT 版, 日経 BP 社, 1997. 5
  - [ 8 ] 佐々木元也著, 多様な決済手段が出そろった EC, 日経インターネットテクノロジー, 日経 BP 社, No. 004 1997. 11.
  - [ 9 ] 日本ユニシス(株)新事業企画開発部著, IT 最前線 What's EC/CALS, ユニシスニュース, 日本ユニシス(株) No. 433 456 1997. 5 1999. 4.
  - [ 10 ] NEC イントラネットプロジェクト編, イントラネット完全構築ガイド, 日経 BP 社, 1996. 10

**執筆者紹介** 新谷 ひろみ (Hiromi Shintani)

1982 年大阪工業大学工学部経営工学科卒業。同年 4 月日本ユニシス(株)入社。シリーズ 8 教育, 情報処理基礎教育, クライアントサーバーのアプリケーション開発技術教育に従事。現在, 関西支社教育サービス推進室に所属, グループウェアや Web 関連を担当。

芹 生 朋 子 (Tomoko Seriu)

1984 年関西学院大学社会学部社会学科卒業。同年 4 月日本ユニシス(株)入社。LINC を中心に SE サービスに従事。現在、関西支社教育サービス推進室に所属、PC プログラミングや Web 関連を担当。

西 文 朗 (Fumiaki Nishi)

1972 年神戸大学大学院修士課程修了。電気工学専攻。同年 4 月日本ユニシス(株)入社。製造業界を中心に SE サービスに従事。主たる担当は基本ソフトウェア。その後、リアルタイム・システムや銀行情報系システムの開発・適用に従事。現在、関西支社教育サービス推進室に所属、データベースや Web 関連を担当。