

文系の学生向けコンピュータネットワーク学習用コンテンツ

岩田員典

要旨

コンピュータネットワークのみならず、コンピュータの仕組みそのものに不慣れな文科系の学生が学習しやすいように、どのようにコンテンツを作成するかについて述べる。

本論文では次の3種類に分類されるコンテンツを作成する。

1. 配布用資料 (PDF)
2. 提示用資料 (Microsoft PowerPoint)
3. 確認用資料 (HTML)

1の配布用資料は、講義で学生が使用するように作成したものである。図や文章中に含まれているキーワードは空欄となっており、講義を聞きながらそのキーワードを記入する形式の資料となっている。

2の提示用資料はその空欄などを説明するためのもので、講義で使用する。またこの資料は1、3とは異なりアニメーションを取り入れている。

3の確認用資料は、1の配布資料におけるキーワードを強調した形でHTMLに変換したものである。提示用資料で聞き漏らしたキーワード等は本資料を利用することで確認することができる。また、HTMLで作成しているためPowerPointが見られないような環境においても、利用することができる。本論文では特に、1と3の資料を同時に作成するSmartDocについて解説する。最後にこれらの資料をどのようにして公開しているかを述べる。

キーワード：文系向け教材，コンピュータネットワーク，SmartDoc

1. はじめに

Windows95 の発売を契機にコンピュータがそれまでよりも身近なものになり 10 年以上経過している。さらに、コンピュータが身近になっただけでなく、インターネットを含めたネットワークを利用するのも当たり前の時代となった。特にインターネットはコンピュータからだけではなく、携帯電話からもアクセス可能になったことで、コンピュータをほとんどさわったことがない人でも利用するに至っている。このような背景の元で、コンピュータネットワークに関する知識がないまま、ネットワークを利用している人が多くなっているのが現状である。特に文化系の大学では、コンピュータの基本的な利用方法は教えても、コンピュータの仕組みやネットワークの仕組みを教えているところは少ないといえる。これは大学に在籍している間にとどまらず、社会に出てからでも必要とされる知識がコンピュータの使い方に限られているからだとと言っても過言ではない。しかしながら、コンピュータやネットワークの仕組みを知らないままこれらを利用していたのでは、システムなどにトラブルが生じた際に対処することが出来ない。このような状態は会社などで専門のスタッフがいるところでは問題とならないが、中小企業や自宅などでは大きな問題となる。そこで、コンピュータの使い方のみならずコンピュータやネットワークの仕組みを理解し、トラブルに対処する方法を身につけることは重要である。

その一方でこれらの知識を身につけるのは容易ではない。それはコンピュータを専攻とした情報系の学科が多数あるにもかかわらず、コンピュータやネットワークの仕組みを、きちんと身につけられている学生が少ないことから明らかである。さらには、初心者向きと明示してあるコンピュータネットワーク学習用の書籍でさえ、情報技術を学習したことのない学生にとっては、内容を理解するのが困難である。これはコンピュータや情報技術に関する知識はあるものとして書かれているためである。そこで本論文では、情報技術に関する学習をほとんどしたことのない文化系の学生に、コンピュータネットワークに関する基本的な知識を身に付けてもらうのを目的としたコンテンツの作成について述べる。本論文の構成は以下の通りである。第 2 章では作成するコンテンツについて述べる。次に第 3 章では SmartDoc について説明し、2 章で述べた資料をどのように作成するかも説明する。第 4 章で作成した資料の公開方法について説明し、最後に本論文のまとめと今後の課題について述べる。

2. 作成するコンテンツ

文科系の学生がコンピュータネットワークを学習しやすいように、次の 3 種類のコンテンツを作成した。

- PDF 形式の配付資料
- Power Point 形式の提示資料
- HTML 形式の確認資料

講義は配付資料を学生に配り、提示資料を示しながら説明をしていく方式を採用している。この方式において PDF 形式の配付資料は、重要なキーワードなどが空欄になっており、講義を聴きながらそれらを埋めていく形をとっている。これは、配付資料にすべてが記載されていると、どこが重要かわかりづらいうえ、学生の集中力が散漫になるからである。空欄があることで講義を聞き漏らすまいと真剣に話をきき、それがキーワードを覚える手助けとなる（図 1）。



図 1 PDF 形式の配付資料

一方、提示資料の内容は配付資料にリンクしているが、全く同じものというわけではない。キーワードの部分は色を変えアニメーションを利用することで、学生に重要だとわかりやすくしている。また、提示資料中の図は、配付資料にある図にアニメーションを付加しわかりやすさを高めている（図 2、図 3）。さらには、配付資料にある文章をそのまま提示資料にしたのでは、

スライドに記載される文字情報が多すぎるため、要点だけを絞って記載している。そして、講義においてはこの提示資料を見せながら、口頭でさらなる説明を行う。



図 2 Power Point 形式の提示資料 1

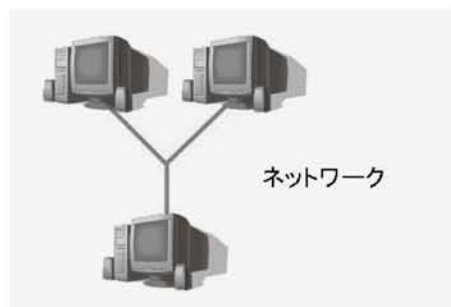


図 3 Power Point 形式の提示資料 2

確認用の資料の使い方は、上記 2 種類の資料とやや異なる。本資料の内容は配付用の PDF の資料とほぼ同じであるが、HTML 形式になっており、ブラウザから確認することが出来る。また、この資料では PDF では空欄になっていたキーワードなど埋められた状態となっている。ただし、他の文字と同じ形式ではキーワードだとわからないため、文字の色を変えた上で下線を引いている（図 4）。そのため、講義でキーワードを聞き漏らした場合でも本資料を見ることで、空欄になったままで放置されること

を防ぐことが出来る。本資料を PDF 形式で作成しなかった理由は、PDF で公開すると空欄のある PDF を利用しない学生が出てくるかもしれないと考えたからである。また、Power Point 形式の提示資料をそのまま見られるようにしなかったのは、Power Point 形式のファイルを開くには専用のソフトウェアが必要だからである。HTML 形式にしておけばよほどのことがない限り Browser がインストールされているだけで見ることが出来るため、視認性が高いといえる¹。

コンピュータネットワークとは

コンピュータの世界では、ネットワークとはコンピュータ同士を相互に接続することを意味します。これは、ケーブルで接続されていようと、電波によって無線で接続されていようとかわりません。複数のコンピュータがお互いに連絡の取れる状態で接続していれば、それはネットワークといえます。なので、世界中のコンピュータがつながっていても、隣り合った2台のコンピュータが接続されていてもどちらも同じネットワークということが出来ます。

図 4 HTML 形式の確認用資料

3. SmartDoc

本論文で作成しているコンテンツは3種類の形式に渡っている。そのためこれらをすべて作成するにはかなりの労力が必要となる。そこで、少しでもその労力を減らすために、本論文では SmartDoc というツールを利用している。本章では SmartDoc とその使い方について説明する。さらには、本論文で作成したコンテンツ向けの設定についても述べる。

¹ PDF も OS によってはインストール直後には読めないのが、HTML 形式が使いやすいといえる。

3.1 SmartDoc とは

SmartDoc は XML をベースにしたドキュメント生成ツールである²。SmartDoc 形式のドキュメントから HTML4.0, HTML3.2, LaTeX2e, JavaHelp および planetext を作成することが出来る。そうすることで、単一のドキュメントから、WWW による公開と、LaTeX による高品質印刷、JavaHelp によるオンラインドキュメントの生成、メール送信用のテキストの生成を一度に行える。

3.2 SmartDoc のインストール

SmartDoc のインストールにはまず SmartDoc の最新版を

<http://www.smartdoc.jp/download/> からダウンロードしてくる³。インストールと実行には Java が必要なのであらかじめインストールしておく。Java は Java2

(JDK1.2) 以降が必要である。ダウンロードしてきたファイルは「smartdoc-x.y.zip」(x, y はバージョン情報)と名付けられている。拡張子が ZIP になってはいるが、unzip して使うわけではないので、その点には注意が必要である。

インストールには、インストールプログラムを次のように起動する。

```
% java -jar smartdoc-1.1.zip
```

² 詳細は <http://www.smartdoc.jp/> を参照のこと。

³ 現在は 1.1 が最新であるが、ベータ版もある。

インストールディレクトリを聞かれるので入力する。デフォルト値で良い場合にはそのままリターンを入力する。

Install directory [default: /usr/local/lib/sdoc]:

コマンドスクリプトのインストールディレクトリも聞かれるので入力する。同じくデフォルト値で良い場合にはそのままリターンを入力する。ただし、このディレクトリには実行パスが通っている必要がある⁴。

Command directory [default: /usr/local/bin]:

以下のように設定の確認をしてくるので良ければ「yes」を入力する。

[Configuration]

Install directory = /usr/local/lib/sdoc

Command directory = /usr/local/bin

Type yes to install, no to re-enter, exit to exit
> yes

以下のようにインストールが実行される。

Extract archives...

Generate script...

script = /usr/local/bin/sdoc

Done.

⁴ インストール後に追加することも出来る。

UNIX の場合には、生成されたスクリプトに実行権を付加する必要があるため、以下のコマンドを実行する。

% chmod a+x /usr/local/bin/sdoc

スクリプトのインストールディレクトリにパスが通っていればすぐにでも実行が出来る。実行結果が以下のようになれば、正常に動作しているといえる。

% sdoc -version

Copyright(c) 1998-2003 ASAMI,Tomoharu.

All rights reserved.

SmartDoc Version 1.1 (20031212) by

asami@XMLSmartDoc.org

3.3 SmartDoc の基本的な利用方法

3.3.1 ファイルの作成

基本的なファイルの書き方は図 5 の通りである。

```
<?xml version='1.0' encoding="EUC-JP" ?>
<doc xml:lang="ja">
  <head>
  </head>
  <body>
    こんにちは!
  </body>
</doc>
```

図 5 SmartDoc のサンプル

「<head>」タグ内にはタイトルなどを記述することができ⁵、本文に記述したい内容は上記のように「<body>」タグ内に記述する。本文を記述する際に、段落の作成方法は以下の2通りある。

- パラグラフタグ
- 自然入力

パラグラフタグは段落を作成する際に

「<p>」タグを使う方法である。一方、自然入力では、空白行を段落の変更と見なす方法である⁶。箇条書きや図・表などもタグを使い記述することが出来る。そのため、SmartDoc の文章作成には LaTeX の知識よりも HTML に関する知識の方が必要といえる。

3.3.2 SmartDoc ファイルの変換

作成したファイルを保存し、sdoc コマンドにより HTML や LaTeX のファイルに変換することが出来る。

3.3.1 節で説明したのと同様の内容のファイルを Hello.sdoc という名前で保存したとする。この SmartDoc ファイルから HTML を作成するには以下のコマンドを実行する。

```
% sdoc Hello.sdoc
```

この場合は Hello.html のみが生成される。LaTeX2e のファイルを生成する場合には format オプションを使う。format オプシ

ョンにパラメタとして latex2e を指定すると LaTeX2e のドキュメントが生成される。

```
% sdoc -format:latex2e Hello.sdoc
```

また、HTML と LaTeX2e のファイルを同時に生成することもできる。両方のファイルを生成する場合には以下のように format オプションに html4 と latex2e の両方を指定する。

```
% sdoc -format:html4,latex2e Hello.sdoc
```

3.3.3 プロパティファイルの利用

3.3.2 節のようにコマンド実行時にオプションを指定することで、複数の種類のファイルを作成したり、作成するファイルの状態を設定したりするが出来る。しかしながら、コマンドラインで入力する際に毎回これらのオプションを入力するのは非常に手間がかかる上に、入力間違いする可能性もある。そこで SmartDoc では SmartDoc.properties と名付けたファイルを作成し、その中に設定を記述することでコマンドラインで長いオプションを入力する手間を省くことが出来る。sdoc コマンドは、コマンドを実行したディレクトリ内に SmartDoc.properties という名前のファイルがある場合のみ、このプロパティファイルの設定を利用する。SmartDoc.propertis の文法は通常の Java のプロパティファイルと同じで“名前=値”の形式を採用してい

⁵ 何が利用できるか詳しくはマニュアル (XML SmartDoc 公式リファレンスマニュアル ピアソンエデュケーション) を参照のこと

⁶ この方式は LaTeX と同じである。

```

<define name="space1">
<native format="html4">
<![CDATA[<span style="color:#990000; text-decoration: underline">]]><refer></refer><![CDATA[</span>]]>
</native>
<native format="latex2e"> [<refer></refer>] </native>
</define>

```

図 6 キーワードのための設定

る。また，“#”でコメントも記述できる。

前述の Hello.sdoc において HTML と LaTeX2e のファイルを異なるディレクトリに作成するための SmartDoc.properties は図 7 のようになる。

ここで、argument0 には通常は変換元のファイルを指定する⁷。

```

argument0=Hello.sdoc
format=html4,latex2e
packager=dir

```

図 7 SmartDoc.properties の例

3.4 本論文で使用している設定

本論文では第2章で説明したように PDF 形式の配付資料ではキーワードは空欄になるようにしている。その一方で確認用の HTML 形式の資料においてキーワードは色を付けて下線を引いている。そのため SmartDoc.properties を利用して、このような処理が行えるようにしている。ただし、すべての処理が行えるわけではないので、

⁷ 厳密には sdoc コマンドに渡される引数を指定することになる。

プロパティファイルに加えてスクリプトと独自の LaTeX の設定ファイルも利用している。本節ではプロパティファイル、スクリプト、LaTeX の設定ファイルについて説明する。

3.4.1 本論文でのプロパティファイル

本論文で説明している資料を作成するためのプロパティファイルの内容を図 10 に示す。尚、「networkXX.sdoc」の「XX」は資料の番号を示し 01～10 まで作成した。また、キーワードを PDF 形式では空欄にし、HTML 形式では色を変えて下線を引くために XML のヘッダ部分に図 6 のような設定を加える。ヘッダに記述した設定を利用し、キーワードの部分は図 8 のように記述する。尚、図 8 においては改行がなさ

```

<strong><space1>コンピュータ同士を
相互に接続する</space1></strong>

```

れているが、実際は 1 行で記述している。

図 8 キーワードの設定

3.4.2 スクリプトファイル

本論文でのプロパティファイル 3.4.1 節で説明したプロパティファイルと XML の

ヘッダを利用し、キーワードを指定することで、HTML 形式においてはキーワードの色が変わり下線が引かれる。しかしながら、PDF 形式においてキーワードは空欄にはならず、キーワードが残ったままになる(図 9) ⁸。そこで、このキーワードを空欄にするための処理が必要となる。本論文ではそのためのスクリプトを作成し、キーワードを空白に置き換えることで空欄を作成する。その際に文字列の長さを考慮している。そのためのスクリプトを図 11 に示す。

[コンピュータ同士を相互に接続する]
 によって無線で接続されていようとかまい

図 9 キーワードが残ったままの PDF

```
argument0=networkXX.sdoc
verbose=true
format=html4,latex2e
split=section
packager=dir
html4.css.url=./network.css
latex2e.encoding=ISO-2022-JP
latex2e.option=twoside
latex2e.class=article=jsarticle
latex2e.package=graphicx,lecture
latex2e.pagestyle=fancy
latex2e.imageLoc=htbp
latex2e.box=AscmacLaTeX2eBoxHandler
latex2e.table=ArrayLaTeX2eTableHandler
latex2e.program=AllttLaTeX2eProgramHandler
latex2e.console=AllttLaTeX2eConsoleHandler
```

図 10 プロパティファイル

⁸ 設定の段階で空欄にすることも可能ではある。

```
#!/usr/bin/perl
use File::Basename;
require "jcode.pl";
$count = 0;
$base = basename($ARGV[0], "%*.tex");
$newname = $base . "-space.tex";
open(IN, "$ARGV[0]");
open(OUT, ">$newname");
while(<IN>){
    $count++;
    chomp $_;
    $_ = &jcode'euc($_);
    if( / %*.*/ ){
        print "$count¥n";
        print "< $_¥n";
        print "---¥n";
        s/ %*.*/ /make_space(length($&))/eg;
        print "> $_¥n";
    }
    print OUT "$_¥n";
}
close(OUT);
close(IN);
sub make_space {
    $length = $_[0]/2 * 1.5;
    $space = "  ";
    for($i = 0; $i < $length; $i++){
        $space = $space . "  ";
    }
    $space = $space . "  ";
}
```

図 11 空欄作成スクリプト

文字列の長さを考慮して空欄を作成すると述べたが、単純に元々の文字列と同じ長さの全角スペースを入れたのでは空欄が狭すぎる。そこで本スクリプトでは、元々の文字列の長さの 1.5 倍となるように全角ス

ースを挿入している (図 12)。

[
、電波によって無線で接続されようとかま

図 12 空欄作成後の PDF

4. 資料の公開

作成した資料のうち、配付資料は
<http://egg.aichi-u.ac.jp/2007/> にて公開し
ている (図 13)。

- [第1回 イントロダクション](#)
 - [第2回 コンピュータネットワーク](#)
 - [第3回 2進数](#)
 - [第4回 イーサネット](#)
 - [第5回 アドレス](#)
 - [第6回 プロトコルとOSI参照モデル](#)
 - [第7回 TCP/IP](#)
 - [第8回 ネットワークを支える技術](#)
 - [第9回 Linuxを使った演習1](#)
 - [第10回 Linuxを使った演習2](#)
- [演習用スクリプト](#)

図 13 公開中の資料

ただし、上記資料は学外から無条件でアクセスされるのを禁止している。これには、Apache⁹のアクセス制限を利用し、学外からのアクセスに対してはユーザ名とパスワードが必要となっている¹⁰。尚、学内からのアクセスではユーザ名・パスワードともに不要である。Apache のアクセス制限のための設定ファイルを図 15 に示す。

⁹ <http://www.apache.jp/>

¹⁰ 詳しくは

<http://www.shtml.jp/htaccess/basic.html> 等を参照のこと

図 14 ユーザ名とパスワードの入力

```
AuthUserFile /home/xxx/xxx/network/.htpasswd
AuthGroupFile /dev/null
AuthName "Please enter username and password"
AuthType Basic

require valid-user
Satisfy any

order deny,allow
allow from xxx.xxx.xxx.xxx
allow from 127.0.0.1
deny from all
```

図 15 アクセス制限用設定ファイル

また、図 15 で設定されている項目のうち「AuthUserFile」において、ユーザ名とパスワードを保存しているファイルを示している。本ファイルは次のコマンドによって作成することが出来る。

```
% htpasswd -c .htpasswd username
New password:
Re-type new password:
```

「-c」オプションは新規ファイル作成時のみに必要である。「username」にはアクセスする際のユーザ名を入力し、2 度のパス

ワード入力を経てファイルに情報が付加される。その際にパスワードファイルは暗号化されるため、本ファイルが流出したとしてもパスワードが漏れることはない。

その他の資料は LMS¹¹ の CEAS¹² を利用し、<http://neldb.aichi-u.ac.jp/> にて公開している（図 16、図 17）。



図 16 CEAS のログイン画面



図 17 授業実施画面

5. まとめと今後の課題

本論文では文系学生向けのコンピュータネットワーク学習用コンテンツをどのよう

に作成するかについて述べた。コンテンツとしては 3 種類の資料を作成することで学習効果を高めた。特に配布用資料においては、空欄を穴埋めする形で学習を進めることで重要項目の学習をしやすいようにした。また、本論文では配布用資料を作成するのに使用している SmartDoc について解説した。基本的な使い方から、空欄を含む資料の作成方法まで説明した。最後に作成した資料の公開方法について説明した。

今後は LMS を利用することで学生との質疑応答等を通じて資料を改編していくとともに、より効果的な資料作成方法がなかいか検討していく予定である。

参考文献

- [1] LaTeX2 ϵ 美文書作成入門，改訂第 4 版 奥村 晴彦，技術評論社，2007.
- [2] XML SmartDoc 公式リファレンスマニュアル，浅海 智晴，ピアソンエデュケーション，2002.
- [3] 入門 XML，第 2 版 エリック・T・レイ，宮下 尚，牧野 聡，立堀 道昭，オライリージャパン，2004.
- [4] Java 言語プログラミングレッスン(上)，改訂第 2 版 結城 浩，ソフトバンククリエイティブ，2005.
- [5] Java 言語プログラミングレッスン(下)，改訂第 2 版 結城 浩，ソフトバンククリエイティブ，2005.
- [6] やさしい Java，第 3 版 高橋 麻奈，ソフトバンククリエイティブ，2005.

¹¹ Learning Management System (学習管理システム)

¹² <http://ceascom.iecs.kansai-u.ac.jp/>

- [7] やさしい Java 活用編, 第3版 高橋
麻奈, ソフトバンククリエイティブ,
2005.
- [8] Apache クックブック—Web サーバ管理
者のためのレシピ集, ケン コール, リ
ッチ ボーエン, 林 秀幸 (翻訳), オラ
イリージャパン, 2004.
- [9] Apache セキュリティ (単行本) アイヴ
ァン リスティック, クイープ (翻訳),
オライリージャパン, 2005.