青山学院大学
Aoyama Gakuin University

字幕付き動画の作成とウェブページへの埋め込みに関するテクニカルノート

～英語教員のための Do-It-Yourself マニュアル～

Adding subtitles to video clips and embedding them in your Web pages

A do-it-yourself manual for teachers of English wishing to introduce subtitle translation as part of their classroom activities but have never got round to it due to technical difficulties.

染谷 泰正
青山学院大学

SOMEYA YASUMASA

Aoyama Gakuin University

謝　辞

本冊子に含まれている以下の 4 つの論文は、青山学院大学の在外研究員制度の適用を受けて、筆者が 2008 年 4 月 1 日から 1 年間、オークランド大学（ニュージランド）に滞在中に執筆したものである。ここに記して関係者への感謝の言葉としたい。

染谷 (2008a) 「動画への字幕の挿入について」 (40 pages)
染谷 (2008b) 「字幕付き動画のオンライン配信について」 (69 pages)
染谷 (2008c) 「「ハードサブ」動画の作成について」 (40 pages)
染谷 (2008d) 「字幕付き動画作成のための SAMI ファイルの作成・編集について」 (35 pages)

なお、上記論文中にも明記してあるとおり、これらの論文はもともとひとつのものとして執筆されたものであるが、大学紀要や学会誌に投稿する必要上、分冊にしたものである。ただし、分冊にしてもなお大学紀要や学会誌が求める制限ページ内に収まらないこと、およびその性質上、一括して読んでいただくことが好ましいことから、本冊子のような形をとって出版することにしたものである。

2009 年 4 月 1 日
青山学院大学 文学部英米文学科
染谷 泰正

【著者紹介】 染谷 泰正 (SOMEYA Yasumasa)　青山学院大学教授。日本通訳学会理事･会誌編集長 (2000-2007)。日本メディア英語学会理事 (2005-)。専門は言語情報科学、コーパス言語学、ESP（とくにビジネス英語）、通訳理論・通訳教育方法論。近年の主な論文として、「英語教育における母語の扱いについて——メタ言語能力を育てるための "CA＋1" の英語教育の勧め」 *Interactive* 23 号, 2007 年 (pp. 7-10) 旺文社、「ビジネス英語のコロケーション分析： 相互情報量 (MI) と仕事量基準 (CC) による分析とその比較」 青山学院大学紀要第 47 号, 2005 年 (pp. 117-131)、「通訳ノートテイキングの理論のための試論——認知言語学的考察」『通訳研究』 第 5 号, 2005 年 (pp. 1-29) 、「わが国の大学・大学院における通訳教育の実態調査」（共著）『通訳研究』 第 5 号, 2005 年 (pp. 285-310) 等がある。

総目次 ※

謝辞 . . . i
目次 . . . ii
Part 1　**動画への字幕の挿入について**　(40 pages) . . . 1
Part 2　**字幕付き動画のオンライン配信について**　(69 pages) . . . 41
Part 3　**「ハードサブ」動画の作成について**　(40 pages) . . . 111
Part 4　**字幕付き動画作成のための** SAMI **ファイルの作成・編集について** (35 pages) . . . 151

※ 総目次のページ番号は通し番号です。以下、各パートごとの目次はそれぞれのパート内でのページ番号です。なお、各ページ下部中央に通し番号のページ数を、ページ上部外側に各パート内でのページ番号を表示してあります。

Part 1

動画への字幕の挿入について

1. 字幕付き動画の出力に最低限必要なもの ... 1
2. 字幕ファイルの作成 ... 2
   2.1 SAMI ファイルの書式と文法 ... 4
      2.1.1 <SAMI> SAMI タグ ... 4
      2.1.2 <HEAD>〜</HEAD> ヘッダ部 (Lines 2-14) ... 4
         1) <STYLE TYPE="text/css">〜</STYLE> スタイル情報指定部 (Lines 4-13) ... 4
         2) サンプル（図 2）で使用されているスタイルタグ（CSS タグ）の解説 ... 4
            a) 基本スタイルの設定 ... 5
            b) 言語クラス定義 ... 7
            c) 多言語字幕ファイルの作成 ... 8
            d) 言語クラス内での書式スタイルの指定 ... 10
      2.1.3 <BODY>〜</BODY> 本文（字幕情報指定部）(Lines 15-27) ... 11
         1) 空白行の挿入 ... 11
         2) SAMI ファイルで使用可能な HTML スタイルタグ ... 12
   2.2 テンプレートを使った SAMI ファイルの作成 ... 13
   2.3 字幕ファイルのファイル名と保存先 ... 14
3. 字幕付き動画の出力 — 字幕ファイルと動画ファイルの同期表示 ... 15
   3.1 Windows Media Player (WMP) を使って字幕を表示させる ... 15
   3.2 独自の字幕エンジンを搭載したメディアプレーヤ（GOM Player や ViPlay）を使う ... 17
      3.2.1 GOM Player について ... 17
      3.3.2 ViPlay について ... 20
      3.2.3 「ソフトサブ」と「ハードサブ」について ... 20
   3.3 DirectVobSub（字幕表示用プラグイン）をインストールする ... 21
      3.3.1 インストールの手順 ... 21
      3.3.2 インストールの確認および初期設定 ... 23
      3.3.3 DirectVobSub の起動 ... 25
      3.3.4 字幕の 2 重表示防止について ... 25
4. まとめ ... 28

資料 1 SAMI 字幕ファイル テンプレート 1（日本語字幕用）<SAMI_Template1.smi> ... 30
資料 2 SAMI 字幕ファイル テンプレート 2（日英 2 ヵ国語用）<SAMI_Template2.smi> ... 31
補足資料： ハードウェア・アクセラレーションと画像表示 ... 32
巻末注 ... 33
参考文献･資料 ... 39

Part 2

字幕付き動画のオンライン配信について

1. インターネット上で字幕付きの動画を配信するための前提条件 ... 1
2. HTML ファイルへの WMP の埋め込み ... 2
   2.1 サンプルソースコード 1：単一言語クラス用ソースコード ... 2
      2.1.1 WMP の埋め込み (Lines 20-30) ... 2
      2.1.2 動画と字幕ファイルの読み込み (Lines 23-24) ... 2
      2.1.3 プレーヤを制御するためのパラメタの指定 (Lines 28-30) ... 4
      2.1.4 字幕表示部（字幕フレーム）の設定 (Lines 33-35) ... 5
   2.2 サンプルソースコード 2：複数言語クラス用ソースコード ... 7
      2.2.1 字幕ファイルと動画ファイルの読み込みと言語クラス設定情報の取得 (Lines 15-20) ... 7
      Player オブジェクトモデルでのローカルパス指定について ... 8
      2.2.2 WMP の埋め込み とパラメタの指定(Lines 58-64) ... 11
      2.2.3 字幕表示エリアの設定と言語クラス選択ボックスの作成 (Lines 41-43) ... 11
      2.2.4 字幕の文字サイズ選択ボックスの追加（サンプルソースコード 2 の改良版）... 14
         1) SAMI ファイルへの文字クラス定義の追加 ... 14
         2) HTML ファイルへの SAMIStyle プロパティおよび文字サイズ選択ボックスの追加 .. 16
            a) SAMIStyle プロパティの追加 ... 16
            b) 文字クラス選択ボックスの追加 ... 17
   2.3 サンプルソースコード 3：プレーヤの横に大きく字幕フレームを開く ... 20
      2.3.1 ページ設計の基本的な考え方とタグ構造 ... 20
      2.3.2 幕の表示方法に関する工夫 ... 23
         1) 色と太字（強調文字）で現在チャンクを動的に示す ... 23
         2) 下線で現在チャンクを動的に示す ... 24
         3) 文字背景色を使って現在チャンクを動的に示す ... 25
      2.3.3 パワーポイントファイルの動画化について ... 26
   2.4 サンプルソースコード 4：字幕を縦書き表示させる ... 26
      2.4.1 ルビの表示 ... 30
   2.5 サンプルソースコード 5： DirectVobSub を使って字幕をオーバーレイ表示させる ... 31
      2.5.1 <embed> タグによる WMP の埋め込み ... 32
      2.5.2 「ハードサブ」の作成について ... 34
3. まとめ ... 36

「アクセシビリティ」に関する補足 ... 37

付録：パワーポイントスライドの動画化について ... 39

資料 1（図 2a） サンプルソースコード１（単一言語クラス用）
<HTML_Template1.txt> ... 48
資料 2（図 3a） サンプルソースコード 2（複数言語クラス用）
<HTML_Template2.txt> ... 50
資料 3（図 8a） サンプルソースコード 3（プレーヤの横に字幕フレームを開く）
<HTML_Template3.txt> ... 52
資料 4（図 12a） サンプルソースコード 4（字幕の縦書き表示）
<HTML_Template4.txt> ... 54
資料 5（図 14a） サンプルソースコード 5（DirectVobSub による字幕表示）
<HTML_Template5.txt> ... 56
資料 6（図 15a）サンプルソースコード 6（フォントサイズを相対指定する）
<HTML_Template6.txt> ... 58
巻末注 ... 61
参考文献･資料 ... 68

Part 3
「ハードサブ」動画の作成について

はじめに ... 1
1. VirtualDub を使った「ハードサブ」動画の作成 ... 3
   1.1 使用するアプリケーション ... 3
      1) VirtualDub 1.8.3 (32-bit version) ... 3
      2) VobSub 2.23 ... 3
   1.2 VirtualDub のダウンロードとインストール ... 3
   1.3 Vobsub のダウンロードとインストール ... 4
   1.4 VirtualDub の起動およびハードサブ作成の手順 ... 5
   1.5 映像サイズと字幕の品質について ... 11
2. DivXLand Media Subtitler を使った「ハードサブ」動画の作成 ... 14
   2.1 使用するアプリケーション ... 14
      1) DivXLand Media Subtitler 2.0.5 ... 14
      2) VirtualDub 1.8.3 (32-bit version) ... 14
      3) Subtitler.vdf (ハードサブ用のプラグイン) ... 14
      4) Xvid codec 1.1.3 Stable ... 14
      5) BeSweet (Audio Encoder) v.1.5b31 ... 15
   2.2 Media Subtitler のダウンロードとインストール ... 15
   2.3 Subtitler.vdf のダウンロードとインストール ... 15
   2.4 Xvid codec のダウンロードとインストール ... 16
   2.5 BeSweet のダウンロードとインストール ... 16
   2.6 Media Subtitler の起動およびハードサブ作成のための準備作業 ... 16
      プレビュー時の字幕の位置設定 ... 17
      プレビュー用のフォント設定 ... 17
      外部フィルタの設定と音声データの抽出 ... 18
      ハードサブの作成 ... 19
3. TMPG Encoder を使った「ハードサブ」動画の作成 ... 22
   3.1 使用するアプリケーション ... 22
      1) TMPGEnc version 2.525（無償版） ... 22
      2) DirectVobSub ... 22
      3) 対応する動画および字幕ファイルの形式 ... 22
   3.2 TMPG Encoder のインストールとプログラムの起動 ... 22
   3.3 「プロジェクトウィザード」を使ったエンコーディング ― ハードサブ動画の出力 ... 23
   3.4 MPEG-1 と MPEG-2 の違い、および映像のサイズ設定について ... 26
4. GOM Encoder を使った「ハードサブ」動画の作成 ... 28

4.1　使用するアプリケーション ... 28
1) GOM Encoder（試用版） ... 28
2) 対応する動画および字幕ファイルの形式 ... 28
4.2　GOM Encoder のインストールとプログラムの起動 ... 28
4.3　入出力映像の設定、および字幕の書式設定 ... 28
5. まとめ ... 33

補足資料：Windows のデータ実行防止 (DEP) の設定の変更 ... 34
巻末注 ... 36
参考文献･資料 ... 39

Part 4

字幕付き動画作成のための SAMI ファイルの作成・編集について

1. 一般的な文書作成用エディタと WMP を使って字幕ファイルを編集する ... 1
2. 「Windows Media ファイルエディタ」を使って正確なタイミングを計測する ... 3
   2.1 「Windows Media エンコーダ 9」(WME9) のインストール ... 3
   2.2 「Windows Media ファイルエディタ」の起動と字幕タイミングの計測 ... 4
3. 専用の「字幕エディタ」を使って字幕ファイルを編集する ... 5
   3.1 SubCreator (Ver.1.2) ... 6
   3.2 Subbu (0.3.0) ... 7
   3.3 ChronoSub (7.7.1) ... 8
   3.4 DivXLand Media Subtitlerb (ver 2.0.5) ... 10
   3.5 SubStation Alpha (v4.08) ... 12
   3.6 Subtitle Processor (7.7.1) ... 13
   3.7 Subtitle Workshop (Ver. 2.51) ... 16
4. まとめ ... 21

付録：「Windows Media ファイルエディタ」を使った動画ファイルの編集について ... 22
1. 「Windows Media ファイルエディタ」を使ってできること ... 22
2. 動画ファイルの切り詰め編集 ... 22
   2.1 分割したファイル（ビデオクリップ）を連続して再生するための方法 ... 23
      1) 「ASX ファイル」の作成 ... 23
      2) HTML ページの作成 ... 25
3. 「再生位置マーカー」の挿入 ... 27
4. WMP で「再生位置マーカー」付きの動画ファイルを再生する ... 30
5. ウェブページ上に埋め込まれた WMP での表示と操作 ... 31

巻末注 ... 33
参考文献･資料 ... 35

動画への字幕の挿入について

～ 英語教員のための Do-It-Yourself マニュアル (1) ～

染谷 泰正
（青山学院大学）

英語教育の現場で、映画やドキュメンタリー、ニュース番組などの映像素材を使うことはもはや珍しくなくなったが、こうした素材を「第 2 言語習得」の促進、あるいは広く言語運用能力の養成という観点から活用するためのひとつの有効な方法として「字幕翻訳 (subtitle translation)」がある。英語の素材に日本語の字幕を入れたり、日本語の素材に英語の字幕を入れたりする作業である。こうした字幕翻訳作業の教育的効果についてはすでに同僚の稲生氏が詳しく報告しており (稲生 2004, 2007)、ここでは繰り返さないが、一般の英語教員が字幕翻訳を授業の中に取り入れるに当って最も問題となるのが、作成した字幕を映像に同期させて表示する際の技術的な障壁である。実は、ごく標準的なパソコンさえあれば、誰にでも比較的簡単な操作で動画に字幕を載せることのできる技術はすでに存在しているのだが、残念ながら、こうした技術の解説はもっぱら専門家＝開発者向けに書かれており、一般ユーザにとってはきわめてアクセシビリティの低いものになっている。解説書の解説書が必要たる所以である。

本稿は、「字幕翻訳」を授業の中に取り入れたいと考えているが、主として技術的な問題のために実行に踏み切れないでいる英語教員をおもな対象として、各種の動画にパソコン上でリアルタイムに字幕を表示させるための方法を、できるだけ具体的に、わかりやすく解説することを目的とする（ただし、パソコンを扱う上で最低限必要な基本的知識はあるものとして議論を進める）。なお、本稿での解説はすべて Windows XP (SP2) 上での作業を前提とする。また、授業で使用することを前提としているため、本稿で紹介するツールは原則としてすべてフリーウェアとして無償で公開され、インターネット上で入手可能なものに限定する。

1. 字幕付き動画の出力に最低限必要なもの

字幕付きの映像を出力するためには、まず対象となる映像のデジタルファイルを用意し、これに対応する字幕ファイルを作成する必要がある。この 2 つが揃えば、あとはこれを適当なメ

SOMEYA, Yasumasa (2008a), “Adding subtitles to video clips” (Part 1 of the monograph, *Adding subtitles to video clips and embedding them in your Web pages: A Do-It-Yourself manual for teachers of English wishing to introduce subtitle translation as part of their classroom activities*, 2008.)

(c) 2008 Yasumasa Someya

ディアプレーヤを使って出力する（**図 1**）。なお、本稿では動画を対象とするが、字幕付与の仕組みは音声ファイルやスライドショー用の静止画像[1]でも基本的には同じである。

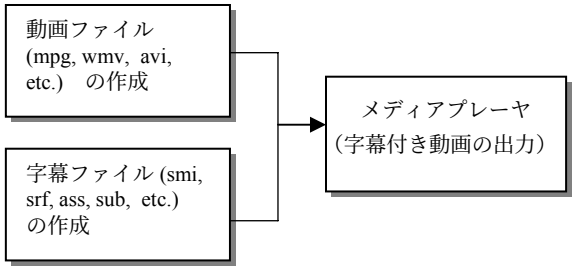


図 1　字幕付き動画の出力

字幕ファイルと動画ファイルを同期させてパソコン上のメディアプレーヤに表示させるためには、主として 次の 3 つの方法がある。

1) Windows Media Player (WMP) の字幕表示機能を使う。
2) 独自の字幕エンジンを搭載した GOM Player や ViPlay のようなメディアプレーヤを使う。
3) DirectVobSub などの字幕表示用プラグイン[2]をインストールする。

このうち、1) は映像画面の下に字幕表示専用エリア（字幕ペイン）を開き、そこに字幕を表示させるスタイルになる（**図 3**, p. 9）。これに対して、映像上に直接字幕をかぶせる「オーバーレイ方式」の字幕表示をするためには 2) と 3) の方法を使う（**図 9a**, **図 9b**, p.17; **図 13**, p. 26）。それぞれの具体的な手順、対応する動画メディア形式、相対的なメリットとデメリット、および（2 と 3 の場合の）プレーヤおよびプラグインの入手方法等の詳細については後述するが、いずれの場合も、使用する動画に対応した「字幕ファイル」を用意しなければならない。そこで、まずは字幕ファイルの作成について説明する。

2. 字幕ファイルの作成

字幕ファイルにはさまざまな形式があるが、もっとも扱いやすく、かつ汎用性が高いのはマイクロソフト社が公開している字幕表示技術（キャプショニング・テクノロジー）である SAMI (Synchronized Accessible Media Interchange) 方式である。[3]

SAMI を用いて映像上に文字情報を表示させるためには、SAMI の文法に従った SAMI ファイル（. smi という拡張子を持つテキストファイル）を作成する。次ページの**図 2** に示すとおり、SAMI ファイルは基本的にはウェブページ作成用の記述言語である HTML (HyperText Markup Language) に似た構造を持ち、冒頭のヘッダ部（<HEAD>～</HEAD>）に組み込まれたスタイルパラメタ･ブロック（<STYLE>～</STYLE>）でフォント、文字サイズ、字幕表示位置、表示色等のスタイル情報を指定し、本文（<BODY>～</BODY>）中に字幕情報を記述する。

```
01  <SAMI>     ← SAMI ファイルであることの宣言(必須要素)
02  <HEAD>     ← ヘッダ部開始タグ
03  <TITLE>Sample SAMI File</TITLE>   ← タイトル表示（オプション）
--  (「パラメタ情報」記述部 ←オプション＝巻末注 4 参照)
04  <STYLE TYPE="text/css">    ← スタイルタグとタイプ指定
05  <!--       ← コメントアウト開始タグ
06    P { margin-top:10pt; text-align:center;
07        font-size:10pt; line-height:125%;
08        font-family: Arial,'ＭＳ ゴシック';
09        color:white; background-color:transparent;
10        }
11    .JACC {name:Japanese; lang:ja; SAMIType:CC;}
12  -->          ← コメントアウト終了タグ
13  </STYLE>     ← スタイルセクション終了タグ
14  </HEAD>      ← ヘッダ部終了タグ
15  <BODY>       ← 本文開始タグ
16  <SYNC START=0>                ← 映像開始時間
17     <P CLASS=JACC>        ← 空白行
18  <SYNC START=1000>             ← 最初の字幕開始時間
19     <P CLASS=JACC>最初の字幕   ← 字幕
20  <SYNC START=2000>             ← 次の字幕開始時間（以下同）
21     <P CLASS=JACC>次の字幕     ← 字幕（以下同）
22  <SYNC START=3000>
23     <P CLASS=JACC>この字幕は<br>2行に分けて表示
24    （中略）
25  <SYNC START=60000>            ← 映像終了時間
26     <P CLASS=JACC>        ← 空白行
27  </BODY>     ← 本文終了タグ
28  </SAMI>     ← SAMI ファイル終了タグ
```

スタイル指定と言語クラス指定

字幕本文

図 2　SAMI ファイルの例（単一言語クラスの場合）

以下、**図 2** の例を使って、SAMI ファイルを構成する主要なタグとそれぞれの記述上の注意点について簡単にまとめておく（各行冒頭の数字は本稿での説明のためのもので、実際のスクリプトにはこれらの数字は入らない）。

2.1 SAMI ファイルの書式と文法

2.1.1 <SAMI> SAMI タグ

ファイル冒頭の SAMI タグ（<SAMI>）は、このファイルが SAMI ファイルであることを宣言するもので、SAMI ファイルは必ずこのタグで開始され、最終行も </SAMI> で閉じなければならない。なお、通常の HTML タグと同じように、SAMI ファイルで使用するタグは原則としてすべて閉じられている必要があるが、本文中の <SYNC> と <P> タグは閉じられていなくても特に問題なく動く。また、マイクロソフト社の技術文書中に挙げられている例でもこの 2 つのタグ対応するエンドタグ（</SYNC> および </P>）は使われていないことから、本稿でも記述を簡素化するためにこれらのエンドタグは省略した。

2.1.2 <HEAD>〜</HEAD> ヘッダ部 (Lines 2-14)

ヘッダ部は、ファイルのタイトルを記入する <TITLE> セクションと、スタイル情報を記述する <STYLE> セクションからなる [4)]。このうち <TITLE> セクションはオプション要素で、省略することができる。記述した場合でもプレーヤ上にこの情報が反映されることはない。

1) <STYLE TYPE="text/css">〜</STYLE> スタイル情報指定部 (Lines 4-13)

スタイルタグは文書のスタイル属性を指定するためのタグで、<STYLE TYPE= "…/…"> という書式で記述し、"…/…" の部分には MIME（マイム）タイプを記入する。MIME (Multipurpose Internet Mail Extension) とはインターネット上でデータをやり取りする際の標準的な規格のひとつで、ウェブ上で利用されるファイルの拡張子とファイルのデータ型（データの種類）の関連付けを行うものである。MIME タイプは、TYPE="タイプ名／サブタイプ名" の形式で構成され、たとえば HTML 文書の場合は "text/html" となる。"text" の部分は MIME タイプと呼ばれ、"html" の部分は MIME サブタイプと呼ばれる。図 2 の例のようにデータの内容がスタイルシート (CSS) の場合は TYPE="text/css" となる。タイプ指定は必須属性であり、省略することができない。具体的なスタイル設定情報は <STYLE TYPE="text/css"> から </STYLE> の間に、通常の CSS (Cascading Style Sheets) の規格に従って記述する（CSS については次項で解説）。

なお、スタイルシートに対応していないブラウザでは、STYLE 要素の内容がそのままブラウザ上に表示されてしまうことがあるため、スタイル定義部をコメントアウトし、ブラウザ上で表示されないようにしておくのが慣例である。[5)]

2) サンプル（図 2）で使用されているスタイルタグ（CSS タグ）の解説

CSS とはウェブページのレイアウトを定義する規格で、フォントや文字の大きさ、文字飾り、行間など、テキストの見栄えに関するスタイル情報を、文書の論理構造を記述した HTML から切り離す目的で策定され、WWW に関する標準化団体 World Wide Web Consortium (W3C) によって標準化されているものである 。CSS に関する詳しい解説については他に譲るとして [6)]、本

稿では**図 2** で使用されている以下のスタイル要素 (Lines 6-11) を中心に、その概要を解説する。

```
06   P { margin-top:10pt; text-align:center;
07       font-size:10pt; line-height:125%;
08       font-family: Arial,'ＭＳ ゴシック';
09       color:white; background-color:transparent;
10       }
11   .JACC {name:Japanese; lang:ja; SAMIType:CC;}
```

a) <u>基本スタイルの設定</u>

CSS の最も簡単な記述例は以下のようになる。このようなひとつのまとまりを「要素 (element) 」と呼ぶ。

```
P {color:blue}
```

CSS 要素は、**セレクタ { 宣言}** という構造を持つ。このうち「セレクタ (selector)」はスタイルを適用する対象を示し、「宣言 (declaration)」は適用するスタイルの内容を指定する。上記の例では、P が「セレクタ」で、{color:blue} が「宣言」にあたる。「宣言」はさらに「プロパティ (property)」と「値 (value)」とに分けられる。上記の例では color が「プロパティ」で、blue が「値」である。P はパラグラフを意味するが、SAMI では字幕テキストはパラグラフ要素として記述することになっているため、スタイル指定は原則として <P> タグ（段落タグ）に対して一括して行い、これ以降、本文中のパラグラフ要素内の記述（つまり、字幕本文）に対して、すべてここで指定したスタイルが適用されることになる。

ひとつのセレクタに対して複数の宣言（＝スタイル指定）を書く場合、次のようにそれぞれの宣言をセミコロン（;）で区切る。

```
P {color:white; font-size:10pt; background-color:black;}
```

これは「P に対して、文字色 (color) =白、フォントサイズ (font-size) =10 ポイント、文字列背景色 (background-color) =黒」 を適用せよという意味である。なお、この例のように複数の宣言が連続する場合、見やすくするために次のようにセミコロンごとに行を変えてもよい（HTML と同じく、CSS では改行や空白スペースは無視されるので、プログラム上は “{ ” と “}” （それぞれ open brace (left curly bracket) および close brace (right curly bracket) と呼ぶ）の間はひとつながりのコマンドとして解釈される）。宣言は必要に応じていくつ続けてもかまわない。

```
P { margin-top:10pt;          ← 上マージン幅を指定
    text-align:center;        ← テキストの表示位置を指定
    font-size:10pt;           ← フォントサイズを指定
```

```
line-height:125%;                    ← 行間の幅を指定
font-family: Arial,'ＭＳ ゴシック';   ← フォントの種類を指定
color:white;                         ← 文字の色を指定
background-color:transparent;        ← 文字の背景色を指定
…:…;                                 ← その他のスタイル指定
}                                    ← 宣言の最後は“}”で閉じる
```

字幕表示のためのスタイルプロパティとしては、基本的には上記の例に挙げたものでほぼ十分だと思われる。以下、それぞれのプロパティがどのような値をとることができるかについて、簡単に補足しておく。

- **margin-top (-bottom, -left, -right):**
  テキストの上下左右マージン幅をそれぞれ個別に指定。指定単位は通常、px (=pixel), pt (point), mm, cm のいずれかを使う。本稿ではポイント(pt) 指定を採用。なお、次項の text-align を center に指定している場合は左右マージンの指定は必要なく、また通常の横書き字幕の場合は下マージンの設定も特に必要がないので、上マージン幅のみを指定しておけばよい。
- **text-align:**
  テキストの表示位置を指定。可能な値は left (左寄せ)、right (右寄せ)、center (センタリング)、justify (両端揃え) のいずれか。ブロック要素に対して有効。
- **font-size:**
  フォントの大きさを指定。指定単位には、pt , pc (=pica), in (=inch), mm, cm などの絶対単位と、em, ex, px, % などの相対単位がある [7]。絶対単位による指定はユーザ側で表示サイズを変更することができない固定値で、一方、相対単位指定の場合はブラウザのメニューから「文字のサイズ」を選択することで、ユーザがその表示サイズを変更することができる。一般に、ウェブページをデザインするに当たっては、アクセシビリティという観点から、相対指定（とくに em による指定）が好ましいとされているが、本稿では、画面サイズとともに字幕テクストのサイズについても作成者側がコントロールできるようにという立場から、ポイント(pt) 指定を採用した。アクセシビリティの問題については別途、改めて議論する。
- **line-height:**
  テキストの高さ（行間の幅）を指定。字幕が複数行になる場合、行間をあけて読みやすくしたい場合にこのプロパティを使う。標準は 100%、ダブルスペースだと 200% になる。和文の場合は 125% または 150% あたりにすると読みやすくなる。
- **font-family:**
  フォントの種類を指定。Windows の場合、日本語フォントは「ＭＳ ゴシック」か「ＭＳ

Ｐゴシック」（P はプロポーショナルフォントを指す。それぞれ ‘MS Gothic’ および ‘MS PGothic’ と英文表記にしてもよい）、英文フォントは Arial または Verdana が標準。カンマ（,）で区切って複数記述すると、指定したフォントが無ければ次のフォントが採用される。なお、スペースを含むフォント名は ‘ＭＳ ゴシック’ や ‘Times New Roman’ のようにシングル･クォテーションマークで囲む。これは、HTML 本文（<body>～</body>）内でインライン要素としてスタイル指定をする場合も同じである。[8)]

- **color:**
  文字列の色を指定。色の指定は white や black などの色名称（カラーネーム）による指定の他に、標準のカラーコードを使うことができる。参考までに主なものを挙げておく。black (#000000), white (#FFFFFF), red (#FF0000), blue (#0000FF), yellow (#FFFF00), green (#008000), lightgray (#D3D3D3)。[9)]
- **background-color:**
  文字列の背景色を指定。可能な値は色名とカラーコードのほか、既定値として transparent（透明色）を指定することができる。透明色を指定すると文字列の背景色は字幕表示画面（＝字幕ペイン）の背景色と同じになる。

このほかにどのようなプロパティ（およびその値）があるかについて関心のある方は、巻末の**注 6** に挙げたオンライン資料を参照していただきたい。なお、太字、下線、イタリック、下付き／上付き文字などの修飾要素は、必要な文字列についてのみ HTML タグを使って部分的に追加することができる（表 1, p. 12 参照）。

b) 言語クラス定義

段落要素への基本スタイルの設定に続いて、言語クラスの定義を行う。言語クラスの定義とは要するに字幕に使う言語を指定するもので、**図 2** の例では次の行 (Line 11) がこれに相当する。

```
11    .JACC {name:Japanese; lang:ja; SAMIType:CC;}
```

SAMI での言語クラスの命名法は、ピリオドを先頭に置いて任意の名称を与えることで行う。この例の場合、.JACC は日本語キャプション (JApanese Closed Caption) であることを示す。name の値には字幕で使用する言語名（ユーザが指定する任意の名称）を、lang には言語名コード（と必要に応じて国名コード）を ISO の標準規格に従って記入する（ISO 639 ＝言語名コード、ISO 3166 ＝国名コード）。日本語の標準言語コードは ja なので、ここでは lang の値を ja とする。SAMIType の値は CC (Closed Caption) となる。英語キャプションを使う場合の言語クラス指定は、通常、次のようになる。

```
.ENCC {name:English; lang: en-US; SAMIType:CC;}
```

なお en-US は米国英語 (English-USA) の標準コードであるが、特に英語の種別（国別）を指定する必要がなければ en（英語の言語コード）のみでもかまわない。もちろん、日英以外の欧米およびアジア系言語を指定することもできる。[10)]

c) 多言語字幕ファイルの作成

言語クラスは字幕に使用する言語特有のスタイル指定用に確保されており、SAMI レンダリングエンジンは、ユーザが選択した（または字幕作成者が指定した）言語コードとクラス定義内の name 値を照合し、両者が一致する字幕行をレンダリングする。

図 2 の例に見られるように、日本語のみの単一言語字幕ファイルを作成する場合は、該当する字幕行の段落タグ内に <P Class=JACC> という形式でクラスパラメタを指定しておけば、自動的に各行が読み込まれ、<SYNC> タグ内で指定した開始時間に合わせてプレーヤに表示されることになる。

一方、同じ映像に対して日本語と英語の両方の字幕（日英に限らないが、本稿ではこの 2 つを対象にして議論を進める）を作成したい場合には、次の 2 つの方法がある。

- 日本語字幕用と英語字幕用のファイルをそれぞれ別個に作成する。
- 言語クラス定義の機能を活用して、2 ヵ国語（またはそれ以上）の字幕をひとつのファイルにまとめてしまう。

たとえば、**図 2** の例（日本語字幕ファイル）に英語の字幕を加える場合には、次のように、まず英語の言語クラス定義を追加した上で、英語の字幕行に対して <P Class=ENCC> という形式でクラスパラメタを指定しておく。

```
 .JACC {name:Japanese; lang:ja; SAMIType:CC;}
 .ENCC {name:English; lang:en-US; SAMIType:CC;}      ← 英語の言語クラス定義
-->
</STYLE>
</HEAD>
<BODY>
  <SYNC START=0>
     <P CLASS=JACC>最初の字幕
     <P CLASS=ENCC>This is the first line.        ← 英語の字幕行
  <SYNC START=2000>
     <P CLASS=JACC>次の字幕
     <P CLASS=ENCC>This is the second line.       ← 英語の字幕行
```

このように SAMI ファイル中に 2 つの言語クラスを指定し、それぞれに対応する字幕を用意しておくと、該当する映像をプレーヤで開いた際、どちらの言語の字幕を表示させるかをユー

ザが選択することができるようになる（ただし、プレーヤが SAMI ファイルに対応している必要がある）。たとえば WMP の場合、以下（**図 3**）に示すとおり、プレーヤのメニューから [Play]→[Lyrics, Captions, and Subtitles] （日本語版のプレーヤでは [再生]→[キャプションと字幕] ）を選択すると、表示可能な字幕の言語クラス名がポップアップ表示され、ユーザがこのいずれかを選択することができるようになる。なお、これに先立って、WMP のメニューから [ツール (Tools)] → [オプション (Options)] → [セキュリティ (Security)] を開き、[コンテンツ (Content)] メニューの [ローカルキャプションがある場合は表示する (Show local captions when present)] のチェック ボックスをオンにしておく必要がある。

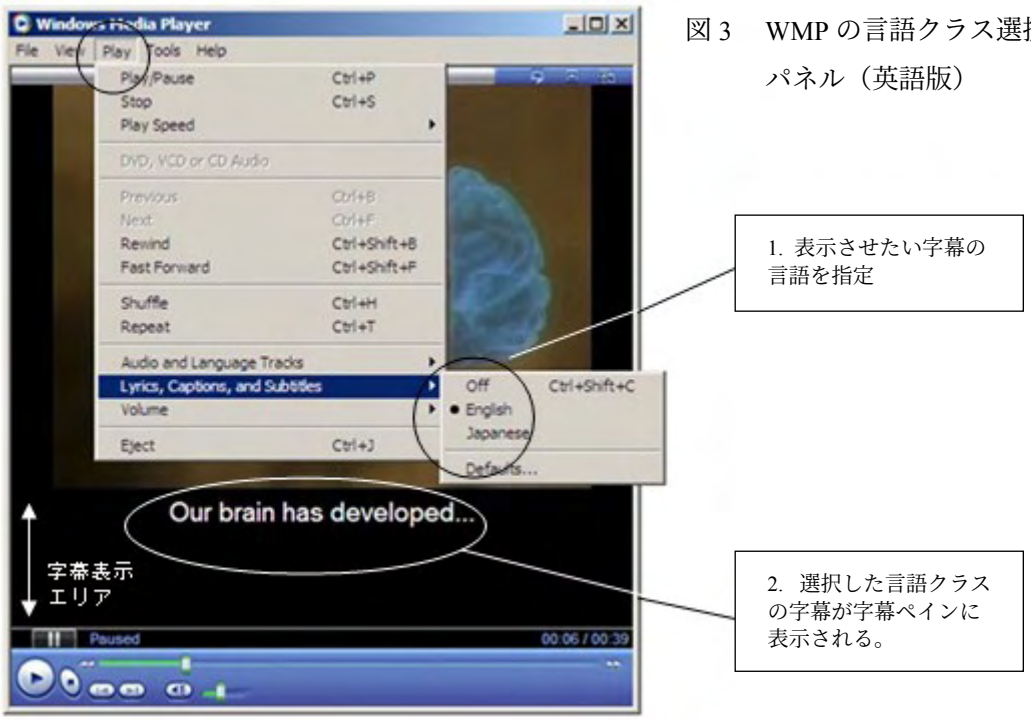


図 3　WMP の言語クラス選択パネル（英語版）

前述の GOM Player では日本語か英語（またはその他の任意の言語クラス）のどちらか一方を選べるほか、「字幕メニュー」の「言語選択」で「すべて選択」を指定すると、用意した複数の字幕を同時に表示させることができる（**図 4**）。[11)]

このように 2 種類の字幕を同時に読ませるのは視聴者の負担が大きいため、いわゆる娯楽のための視聴には向かないが、語学（または翻訳）教育上の理由で部分的にこのような多重表示をしたいときには便利な機能である。なお、WMP にはこの機能はないため、同じことをしたい場合には、同じ行に日本語と英語の字幕を続け、間に改行タグ
 を挿入する。これで、日本語と英語の字幕を、行を変えて同時に表示することができる。

図 4　GOM Player での言語クラス選択パネル

d) <u>言語クラス内での書式スタイルの指定</u>

言語クラスは、言語の指定以外に字幕ブロックの書式を指定・変更するためにも使用することができる。これは、複数の言語を使用している場合に、言語ごとに異なったフォントやマージン等を指定したい場合に有用である。たとえば、同じ映像に対して日本語と英語の両方の字幕をひとつの SAMI ファイルとして作成し、それぞれ言語ごとに一部異なったスタイルを設定したいとする。この場合、以下の例に示すように、共通のプロパティだけを段落要素 P {...} 内で指定し、異なるプロパティは言語クラスごとにそれぞれ個別に指定することができる。

```
P {margin-top:10pt; text-align:center;
   line-height:125%; color:white;
   background-color:transparent;}
.JACC {name:Japanese; lang:ja; SAMIType:CC;
       font-family: Arial,Verdana,Helvetica,
       'ＭＳ ゴシック',sans-serif; font-size:10pt;}
.ENCC {name:English; lang:en-US; SAMIType:CC;
       font-family: Arial,Verdana,Helvetica,Tahoma;
       font-size:11pt;}
```

共通する書式プロパティの指定

言語クラスごとに異なる書式プロパティの指定

この例では、左右マージン、テキストの表示位置、行間幅、文字色および背景色は共通要素としたうえで、日本語字幕では 10 ポイントの ‘ＭＳ ゴシック’（がなければゴシック系のsans-serif）を使い、英文字幕では 11 ポイントの Arial または Verdana, Helvetica, Tahoma のいずれか

をこの優先順位で使うという指定になっている。なお、日本語フォントの指定で先頭にArial がきているのは、和文中に出てくる英文字には ‘ＭＳ ゴシック’ でなくArial を適用するためである（仮名漢字に対してはArial 指定は無視され、自動的に次の ‘ＭＳ ゴシック’ が適用される＝**注 8**参照）。

このように、言語ごとに異なったスタイル指定をする場合は、共通のプロパティだけを段落要素内で指定し、異なるものは言語クラスごとに指定することで、各言語の特性に応じたよりきめ細かな書式設定が可能になる。クラススタイル内で使用した書式設定は、どの段落スタイルの書式設定よりも優先される。なお、言語クラスのほか、「文字クラス」定義をSAMIファイルに追加することで字幕テクストのサイズを複数指定することができるが、これについては別途（染谷2008b, pp. 14-19）解説する。

2.1.3　<BODY>〜</BODY>　本文（字幕情報指定部）(Lines 15-27)

本文の字幕表示部は 2 つの要素からなる。まず、<SYNC START=****> の部分では各字幕行を映像に同期 (SYNCronize) させる開始時間をミリセコンド単位（1 秒=1000ms）で指定し、次に <P CLASS =JPCC> の行に字幕として表示させたい文字列を挿入する。段落要素内の言語クラス指定は、内容に応じて CLASS=JPCC [日本語字幕] または CLASS=ENCC [英語字幕]（これ以外の言語を使う場合はその言語クラス）とする。

```
15  <BODY>
16  <SYNC START=0>                    ← 映像開始時間 12)
17     <P CLASS=JACC>            ← 空白行
18  <SYNC START=1000>                 ← 最初の字幕開始時間
19     <P CLASS=JACC>最初の字幕        ← 日本語クラス指定＋日本語字幕
20  <SYNC START=2000>
21     <P CLASS=JACC>次の字幕
--    （中略）
25  <SYNC START=60000>                ← 映像終了時間
26     <P CLASS=JACC>            ← 最終行（空白行）
27  </BODY>
```

必要なだけの字幕を記述したのち、最終行として開始時間を映像の終了時間に一致させたカラ行＝空白行（次項参照）を置き、ボディタグを閉じる（映像終了時間の明示およびカラ行の配置はオプションだが、映像開始時間と同様に、終了時間にもカラ行を置くことが推奨されている）。

1)　空白行の挿入

字幕ブロックと字幕ブロックの間に空白行を挿入するためには、HTML の非改行スペースエンティティ ( ) を使用する。たとえば、2000ms から始まる字幕 1 の発話が 3000ms で

終了し、次の発話＝字幕 2 が 6000ms から始まる場合、次のように指定すると字幕 1 は字幕 2 が始まるまでの 4 秒間ずっと画面上に表示されることになる（台詞の字幕の場合、発話の終了に合わせて字幕表示も終了させるのが普通であり、このように発話の終了後も字幕が残っている状態は好ましくない）。

```
<SYNC START=2000><P CLASS=JACC>字幕1
<SYNC START=6000><P CLASS=JACC>字幕2
```

これに対して、次のように指定することで、字幕 1 は対応する発話の終了とともに画面上から消え、その後 6000ms の時点で字幕 2 が表示されるようになる。なお、<SYNC>タグと<P>タグはこの例のように１行にまとめても差し支えない。

```
<SYNC START=2000><P CLASS=JACC>字幕1
<SYNC START=3000><P CLASS=JACC>       ← 空白行（字幕表示のない時間帯）
<SYNC START=6000><P CLASS=JACC>字幕2
```

2) SAMI ファイルで使用可能な HTML スタイルタグ

それぞれの字幕は前述の段落要素 P {...} 内で指定された書式（またはクラス定義内で指定された書式）にしたがって表示されるが、本文内でも必要に応じて HTML のタグを使って部分的に細かな指定（＝スタイルシートによる指定の局所的変更または追加）を加えることができる。SAMI で使用可能な HTML タグのうち、字幕表示に関連して主なものは以下のとおりである。なお、このほか <ruby> タグを使って字幕に「ルビ」を振ることもできるが、これについては別稿（染谷 2008b, p. 30）を参照していただきたい。

表 1　SAMI で使用可能な HTML タグ

スタイル	HTML タグ	表示例
文字色	文字列	[****は色名／色コード] [13]
文字サイズ	文字列	[****は数値] [14]
強制改行	文字列 文字列	文字列 文字列
太字	文字列	**文字列**
下線	<u>文字列</u>	<u>文字列</u>
イタリック	文字列	*文字列*
下付き文字	文字列_{文字列}	文字列$_{文字列}$
上付き文字	文字列^{文字列}	文字列^{文字列}

以上、SAMI ファイルの作成に必要な基本的な書式 フォーマットと文法について、**図 2** に挙げた例を使って簡単に解説した。解説はごく基本的なものに絞ってあるものの、本稿の目的に照らして最低限必要な項目はすべて網羅されている。すでに明らかなとおり、SAMI ファイルは独立したテキストファイルであるため、特別なソフトウエアやプログラムを必要とせず、一般的なエディタを使って作成することができる [15)]。 また、字幕情報の変更や追加等のカスタマイズも容易に行えるという利点がある。次節では、出来合いの「テンプレート」を使って、より効率的に SAMI ファイルを作成する方法について解説する。

2.2　テンプレートを使った SAMI ファイルの作成

実際に SAMI ファイルを作成するに当たっては、出来合いの「テンプレート」を使って作成するのが効率的である。そこで、読者の便宜を考えて、巻末に以下の 2 種類の SAMI ファイル･テンプレートを用意した（ファイルのダウンロード先は**注 34** 参照）。

- SAMI_Template1.smi （資料 1　SAMI 字幕ファイルテンプレート 1：日本語字幕用）
- SAMI_Template2.smi（資料 2　SAMI 字幕ファイルテンプレート 2：日英 2 ヵ国語用）

このいずれかのテンプレートを適当なエディタを使って開き、必要な追加･修正を施しながら、独自の字幕ファイルを作成していただきたい。なお、使用するエディタは Windows 付属の「メモ帳」でもかまわないが、できれば「秀丸」や「TeraPad」のようなより高機能なエディタを使うほうがスムーズな作業ができる。前者は有償のシェアウェアであるが、**TeraPad** はフリーウェアで、開発者のサイト (http://www5f.biglobe.ne.jp/~t-susumu/) からダウンロードすることができる。

エディタに SAMI ファイルのテンプレートを読み込んだあとの作業手順は、およそ以下のとおりである。

1) まず、適当な名前（例：Sample.smi）を付けていったん新規ファイルとして保存し、再度このファイルをエディタで開く。
2) タイトルやスタイルシートを必要に応じて書き換える（この作業は後回しにしてもよい）。
3) この新規ファイル＝作業用ファイルに字幕挿入用のタグセット（<SYNC START=****> と <P CLASS=JACC> の行からなるセット）を必要なだけ追加し、最終行まで字幕を作成する [16)]。
4) 字幕作成作業は大きく分けて、字幕内容そのものの作成と、同期時間（字幕を表示させるタイミング）の記入という 2 つの作業からなる。このうち、後者の作業は Windows Media Player (WMP) などのメディアプレーヤを使って動画を実際に再生し、プレーヤ上に表示される経過時間を参照しながら行う（**図 6**）。ただし、WMP を含め、一般的なメ

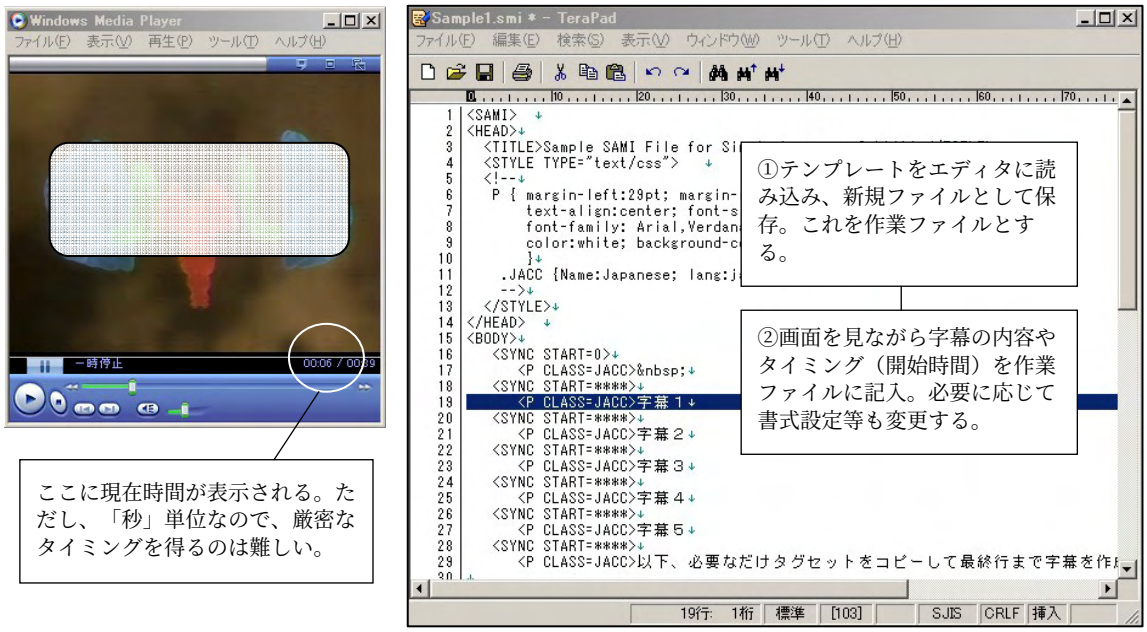


図 6　映像を見ながらエディタで字幕ファイルを作成･編集する

ディアプレーヤでは秒単位でしか経過時間が表示されないので、この段階ではとりあえず秒単位でごく大雑把に各字幕の START time を記入しておく。時間表記はミリセコンド (1 秒=1000; 1 分=60000) 単位で記入し、数字中にカンマやコロン等の区切り記号およびスペースは挿入しない。なお、できればこの段階で 1100 (= 1.1 秒)、2500 (= 2.5 秒) のように 100 分の 1 秒単位まで大体の勘で記入しておくとあとの作業が楽になる。

5) 最終行まで字幕本文および同期時間を入力し終えたら、いったんファイルを保存し、作業を終了する（なお、念のために字幕作成途中でも頻繁に保存作業をしておくのが望ましい）。

以上でとりあえず字幕ファイルができ上がったことになる（字幕ファイルの保存ファイル名および保存先については次項参照）。ただし、これはまだ暫定版という性格のもので、さらに完成度を高めていく必要がある。そのためには、作成した字幕を実際の映像に同期表示させ、これを見ながらそれぞれの字幕のタイミングや書式設定および内容等をさらに調整していく。

なお、字幕表示のタイミングをより厳密に指定したい場合、あるいは本稿で意図しているような語学学習の一環としての字幕作成／字幕翻訳ではなく、プロの字幕翻訳家養成という枠組みで指導ができる場合は、字幕作成専用のソフトウェア（＝字幕エディタ）を使うことができるが、これについては改めて別稿（染谷 2008d）で解説することとしたい。

2.3　字幕ファイルのファイル名と保存先

作成した SAMI ファイルは、対応する動画ファイルと同じ名前をつけた上で（ただし拡張子

は.smi とする）、原則として映像ファイルと同じフォルダ内に保存する。したがって、たとえば動画ファイルが Sample.wmv という名前で MyMovie というフォルダに保存されているとすれば、字幕ファイルは Sample.smi という名称で、同じく MyMovie というフォルダに保存することになる。WMP および字幕表示に対応したその他のメディアプレーヤは、とくに別途指定のない限り、動画が起動すると同時に同一フォルダ内で字幕ファイルを検索し、対応するものがあればこれを自動的に読み込むようになっている。[17)]

字幕ファイルの作成が終了すれば、次はこれを映像と同期させて表示させることになる。次章では、そのための方法について解説する。

3.　字幕付き動画の出力 ― 字幕ファイルと動画ファイルの同期表示

字幕ファイルを映像と同期させて表示させるためにはいくつかの方法があるが、本稿では、冒頭で述べた 3 つの方法、すなわち 1) WMP の字幕表示機能を使う方法、2) GOM Player や ViPlay のような独自の字幕エンジンを搭載したプレーヤを使う方法、および 3) DirectVobSub などの字幕表示用プラグインをインストールする方法のそれぞれについて、以下、順番に解説する。

3.1　Windows Media Player (WMP) を使って字幕を表示させる

前記 3 つの方法のうち、一番簡単なのは WMP を使って字幕を表示させる方法である（本稿ではバージョン 10 および 11 でテストをしているが、バージョン 7 以上であれば基本的には問題ない）。すでに述べたとおり、SAMI 形式で作成した字幕ファイルは WMP で読む込むことができるが、このためにはあらかじめ WMP に対して以下の 2 つの設定をしておく必要がある。

- [ツール (Tools)] メニューから [オプション (Options)] → [セキュリティ (Security)] を選択し、[コンテンツ (Content)] メニューの [ローカル キャプションがある場合は表示する (Show local captions when present)] のチェック ボックスをオンにする。
- [再生 (Play)] メニューから [キャプションと字幕 (Lyrics, Captions, and Subtitles)] を選択し、[利用できる場合はオン (On if Available)] をクリック。（**図 7**）[18)]

この状態（字幕機能をオンにした状態）で WMP で映像を立ち上げると、映像に対応する字幕ファイルが自動的に読み込まれ、字幕表示が可能になる（**図 8**）。なお、WMP で字幕を表示させる際はスキンモードで Corporate を選択するのがよい（メニューから「表示」→「スキンセレクタ」で Corporate を選択）。WMP 11 ではフルモードでも特に問題はないが、この場合は「クラシックメニュー」の表示を有効にしておくこと（タイトルバーから右クリック→「クラシックメニューの表示」をチェック）。なお、9 Series Default など他のスキンでは字幕を表示することができない。

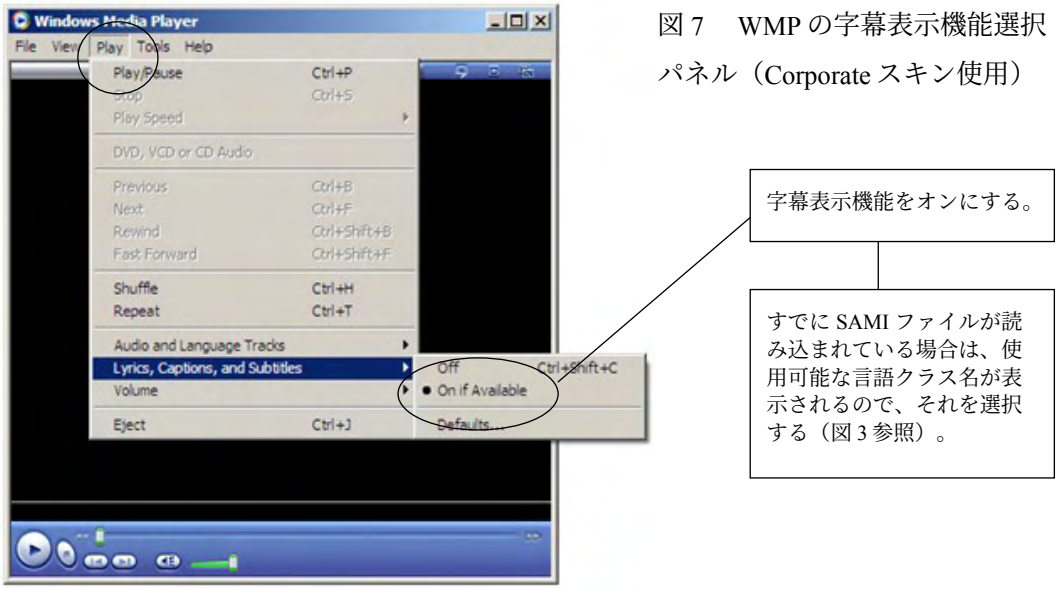


図 7　WMP の字幕表示機能選択パネル（Corporate スキン使用）

図 8　WMP での字幕表示

WMP を使うと、このようにごく簡単な手続きで字幕を表示させることができる。ただし、しいて問題点を指摘するなら、この方法だと、①字幕が画面下部の字幕表示エリア（字幕ペイン）に表示され、画面上にオーバーレイ表示させることができない [19]、②対応するメディア形式が限られている（ただし、wmv, avi, mpg [mpeg-1] といった標準的なフォーマットには対応している [20]）、③一時停止したり途中で巻き戻したりすると映像と字幕の同期が崩れることがある、といった問題点がある。

これらの問題は、いずれも前述の **GOM Player** や **ViPlay** などの字幕表示に対応したメディアプレーヤ、あるいは字幕表示用プラグインである **DirectVobSub** をインストールすることで解決することができる。

3.2　独自の字幕エンジンを搭載したメディアプレーヤ（GOM Player や ViPlay）を使う

字幕表示に対応したメディアプレーヤはいくつかあるが、そのうちフリーウェアとして無償で入手できるものとして GOM Player と ViPlay がある。いずれも SAMI を含む多くの字幕フォーマットに対応しており、字幕ファイルさえあれば、面倒な操作なしに、ごく簡単に動画上に字幕をオーバーレイ表示することができる（**図 9a**, **図 9b**）

図 9a　GOM Player の字幕表示画面

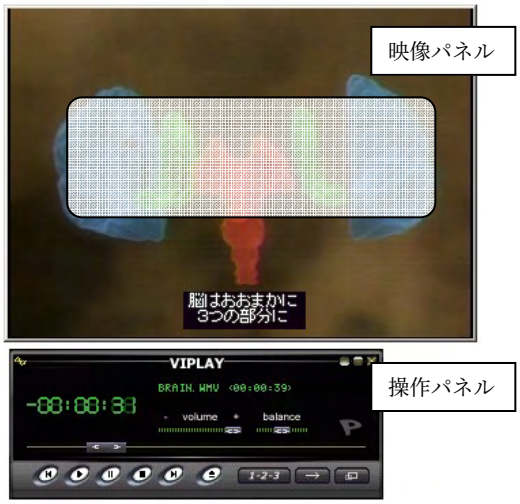


図 9b　ViPlay の字幕表示画面

3.2.1　GOM Player について

GOM Player は株式会社グレテックジャパンが開発し、フリーウェアとして頒布しているマルチメディアプレーヤである。独自の字幕エンジンを搭載しているため、特別なフィルタを導入することなしに、手軽に高品質の字幕を映像に載せることができる。また、各種のコーデックを内蔵しており、avi, mpg, wmv, m4v, mov, flv, divx などほとんどの主要な動画形式に対応している（字幕ファイルは、smi, sub, idx, rt 形式に対応）。GOM Player のダウンロードおよびインストールの手順は以下のとおり。

1) ブラウザで次のサイトにアクセス。→ http://www.gomplayer.jp/
2) 上記サイトのトップページから「今すぐダウンロード」をクリック。
3) 「このファイルをダウンロードしますか」というメッセージが表示されたら、「保存」をクリック。
4) 保存先を「デスクトップ」にして「保存」をクリック。ファイル名はデフォルトのままでよい。
5) ダウンロード終了後、デスクトップ上に保存されている GOM Player のアイコンをダブルクリック。
6) インストール確認メッセージが表示されるので、再度「実行」をクリック。あとは画

面上に表示される指示にしたがって作業を進める（とくに理由がない限り初期設定のままインストールする）。

これで、C ドライブの Program Files フォルダの中に GRETECH > GomPlayer というサブフォルダが作成され、その中に必要なファイルがすべてコピーされる。先ほどダウンロードしたセットアップファイルは削除してかまわない。

プレーヤの基本的な操作方法についてはグレテック社の関連サイト (http://www.gomplayer.jp/use_ intro.html) を参照していただくとして、ここでは動画を立ち上げて字幕を表示させるための標準的な手順、および最低限必要な初期設定について解説する。

1) まず、動画ファイルと SAMI ファイルが同じ名前（例えば Sample.wmv と Sample.smi）で、同一フォルダ内にあることを確認。
2) GOM Play を立ち上げる。
3) 画面上でマウスを右クリックして「メニュー」を表示し、以下の設定をする。
   a) 「字幕メニュー」で「字幕の表示」にチェックマークを入れる。
   b) 「環境設定」→「一般」メニューの「一般」タブで、「画面に字幕を表示する」にチェックが入っていることを確認。
   c) 「環境設定」→「字幕」メニューの「字幕」タブ→「表示方法」で「オーバーレイ表示」を選択。同じく、「フォントタイプ」で「MS　ゴシック」（またはその他の表示させたいフォント）を選択。（注：この「環境設定」画面では字幕の表示方法や字幕ファイルへのアクセスについて、より細かな設定が可能だが、ここではとりあえず必要最小限の設定を行っておく）。
   d) 「環境設定」→「字幕」メニューの「メッセージ」タブ→「表示方法」で「オーバーレイ表示」を選択。同じく、「フォントタイプ」で「MS　ゴシック」（またはその他の表示させたいフォント）を選択。
   e) 「環境設定」→「字幕」メニューの「その他」タブ→「コードページ」で Japanese（またはその他の表示させたい字幕言語）を選択。
   f) 「環境設定」→「フィルタ」メニューの「フィルタ」タブ→「ASF フォーマット再生」で「フィルタを自動で選択して再生」をチェック。
4) 以上の基本設定を確認したのち、再度、メインメニューを開き、「ファイルを開く」をクリックして動画ファイルを選択。これで、指定した動画が再生され、字幕が自動的にオーバーレイ表示される（**図 9a** 参照）。

なお、字幕の表示位置や文字サイズ、色などについてはプレーヤの「環境設定」画面上で適宜調整することができる（**図 10a**）。さらに、映像の表示中に「コントロールパネル」を開き、字幕の表示位置やサイズを、オンラインで（つまり、映像の表示中に）任意に調整することも

可能である（**図 10b**）。これは他には見られない GOM Player ならではの機能である。なお、「環境設定」画面で加えられた変更は一時的なもので、 SAMI ファイルには反映されない（＝SAMI ファイルが書き換えられることはない）。[21)]

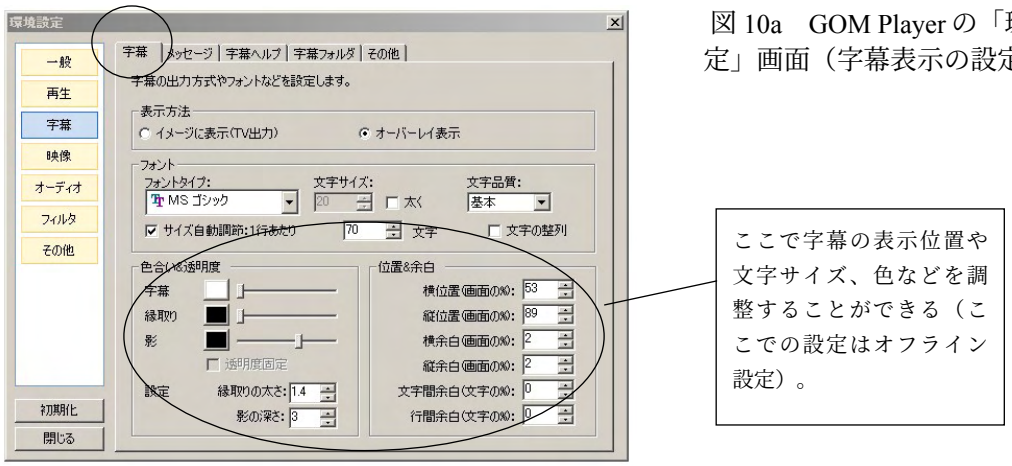


図 10a　GOM Player の「環境設定」画面（字幕表示の設定）

図 10b　GOM Player の「環境設定」画面（コントロールパネルからの字幕表示設定）

GOM Player にはこのほかにもいろいろと便利な機能が用意されているが、多機能のわりに操作が簡単で、初心者でもほとんど迷うことなく使いこなすことができる。あえて不足を挙げるとすれば、字幕ファイルの編集に対応していない点だろう。せっかくメニューの中に「字幕管理」画面があり、ここでは画面上に字幕と時間データを呼び出すことが可能になっているのだから、簡単なエディタ機能を加えてこれらのデータを直接編集することができるようにすれば、世界初の「字幕エディタつきの多機能メディアプレーヤ」ということになって、GOM Player の価値がいっそう増すように思われる。

なお、パソコン上に後述の DirectVobSub がインストールされている場合は DirectVobSub の字幕エンジンと GOM Player の字幕エンジンが同時に起動してしまうため、そのままでは字幕が 2 重表示されることになる。これを防ぐためには、GOM Player の「環境設定」→「フィルタ」設定→「遮断」タブ→「遮断フィルタリスト」画面で“ DirectVobSub ”の項目にチェックを入れ、DirectVobSub を遮断しておく必要がある（詳しくは pp. 25-27 参照）。

3.3.2 ViPlay について

ViPlay は字幕作成用ソフトウア Subtitle Workshop の姉妹品として URU Soft 社が開発したもので、本来は商用ソフトウェアであるが、現在のところフリーウェアとして一般に公開されている。[22)]

筆者の勤務校では字幕翻訳の授業[23)]で上述の Subtitle Workshop を使用していることもあって、当初から ViPlay を使っており、字幕表示を目的としたプレーヤとしては十分な機能を持っているものと評価している（ViPlay の字幕表示画面は**図 9b,** p. 17 を参照）。ただし、映像パネルと操作パネルが分離しており、全画面表示した場合などにやや使い勝手が悪かったり、字幕の背景色が外せないことがあるなどの難点があった。ViPlay を使っているおもな理由は、前述のとおり字幕の作成に Subtitle Workshop を使用しているからであるが、Subtitle Workshop で作成した字幕ファイルは ViPlay 専用というわけではなく、WMP や GOM Player（または字幕表示に対応したその他のメディアプレーヤ）でも読み込むことができる。したがって、プレーヤの選択はもっぱらプレーヤ自体の機能および使い勝手によって決定することができる。筆者の意見では、GOM Player と ViPlay を比較した場合、表示される字幕の質や操作性という点で前者のほうが優れており、新たに教室での指導用に導入する場合には GOM Player をデフォルトのプレーヤとして使うほうがよいと思われる（もっとも、映像パネルと操作パネルが分離しているほうが都合がよいケースも考えられないわけではない。したがって。どちらを選択するかは実際に両者を比較検討した上で、各自が判断していただきたい）。

なお、GOM Player の場合と同じく、ViPlay は独自の字幕エンジンを内蔵しているため、パソコン上に DirectVobSub がインストールされている環境では 2 重に字幕が表示されてしまう。これを防ぐためには、ViPlay のメインメニューから「サブタイトル」を選択し、「サブタイトル表示」という項目のチェックを外しておく必要がある（**注 11** の**図 5** 参照。なお、DirectVobSub のほうをオフにしたい場合は、DirectVobSub のメニューから“ Hide Subtitles ”にチェックを入れておけばよい（pp. 25-27 参照）。ただし、DirectVobSub のほうが字幕の品質はよい）。

3.2.3 「ソフトサブ」と「ハードサブ」について

これまで本稿で説明してきた字幕挿入技術は、再生のときに画面上に動的に字幕を合成するもので、いわゆる「ソフトサブ」と呼ばれる方法である。これに対して、画像の一部として字幕を焼き込んでしまう方法を「ハードサブ」と呼ぶ。[24)]

前者では基本的に映像と字幕データが切り離されているため、字幕表示のオンオフの切り換えが可能であり、字幕データの修正･更新も容易に行うことができる。また、再生時に直接描画するため、より鮮明な字幕テキストが得られるなどの利点がある。ただし、完成した作品として配信する場合に、動画ファイルと字幕ファイルをセットで配信しなければならないことや、ユーザ側の環境によっては字幕が文字化けすることがあるなどの問題がある。また、再生時の CPU 負荷が高いため、場合によっては画面が不安定になることがある。

これに対して「ハードサブ」は動画の一部として字幕が焼きこまれているために、映像のみを配信するだけでよく、文字化けの心配もない。ただし、字幕の質はソフトサブに比べて低くなる。また、焼き込むという手間が加わることもハードサブ方式の欠点であろう（従来の技術ではハードサブの作成にはかなりの手間と専門知識を必要とした。ただし、最近になってごく簡単な手順でハードサブを作成することのできるソフトウェア [25] が開発されたため、ハードサブの作成はもはや特別な手間のかかる作業ではなくなったといってよい）。

どちらも一長一短ということになるが、両者を使い分けるとすれば、作品の完成にいたるまでのプロセスではソフトサブ方式を採用し、完成後は必要に応じてハードサブ方式で字幕を焼きこみ保存する、というのがよいように思われる。なお、ハードサブの作成に関しては染谷 (2008c) を参照されたい。

3.3　DirectVobSub（字幕表示用プラグイン）をインストールする

DirectVobSub とは VobSub (Ver.2.23) というフリーウェアに含まれている字幕表示用のプラグインである。VobSub は字幕作成支援用の統合ソフトウェアで、VobSub for DirectShow (DirectShow 用のソフトサブ･フィルタ)、VobSub for VirtualDub (VirtualDub[26] を使用して映像に字幕を挿入するためのハードサブ･フィルタ)、VobSub for AviUti (AviUti [Avi ファイル加工編集用ツール] 用のハードサブ･フィルタ) という 3 種類のフィルタと、TextSub (字幕作成支援ユーティリティ)、Subresync (字幕フォーマット変換ユーティリティ)、 および Submux (映像･音声･字幕ファイルの統合と avi/ogm 形式での出力ユーティリティ) という各種のユーティリティ群からなる。このうち、本稿で取り上げるのは VobSub for DirectShow (aka, DirectVobSub) で、「ソフトサブ」を作成するための DirectShow[27] プラグインである。

VobSub はハードサブおよびソフトサブ双方に対応した字幕合成用ソフトウェアとして動画マニア (?) の間では定番になっている感があるが、使いこなすためにはそれなり専門知識と忍耐を要求されるため、一般のユーザにとってはいささか敷居の高い存在となっている。ただし、「ソフトサブ」の作成に関しては、前記の DirectVobSub というフィルタをインストールするだけで、簡単にこれを実現することができる。以下、この点に絞って DirectVobSub のインストールと最低限必要な初期設定、および使用に当たっての注意点等について解説する。

3.3.1　インストールの手順

1) まず、以下のいずれかのサイトにアクセスして VobSub 2.23 をダウンロードする（必ず Version 2.23 をダウンロードすること）。
   - http://www.divx-digest.com/software/vobsub.htm
   - http://www.videohelp.com/tools/VobSub
   - http://www.afterdawn.com/software/video_software/subtitle_tools/vobsub.cfm
2) ダウンロード先を「デスクトップ」にして「保存」をクリックするとダウンロードが始まり、ダウンロードが終了すると次のようなアイコンがデスクトップに作成される。

3) デスクトップ上の VobSub アイコンをダブルクリックして実行。
4) インストールする言語を選択（English を選択）して OK ボタンをクリックし、さらにセットアップ開始画面で Next ボタンをクリックして次へ進む。
5) **図 11a** のような「コンポーネント選択」画面が表示されるので、“ type of install ” ボックスで “ Typical ” が表示されていることを確認して Next ボタンをクリック（Utilities に含まれている各種プログラムは今回の作業では必要ないが、とりあえずデフォルトのままにしておく。また、Plugins の選択内容を変更すると、これ以降の作業内容が異なってくるので、ここでは変更しないこと [28)] ）。

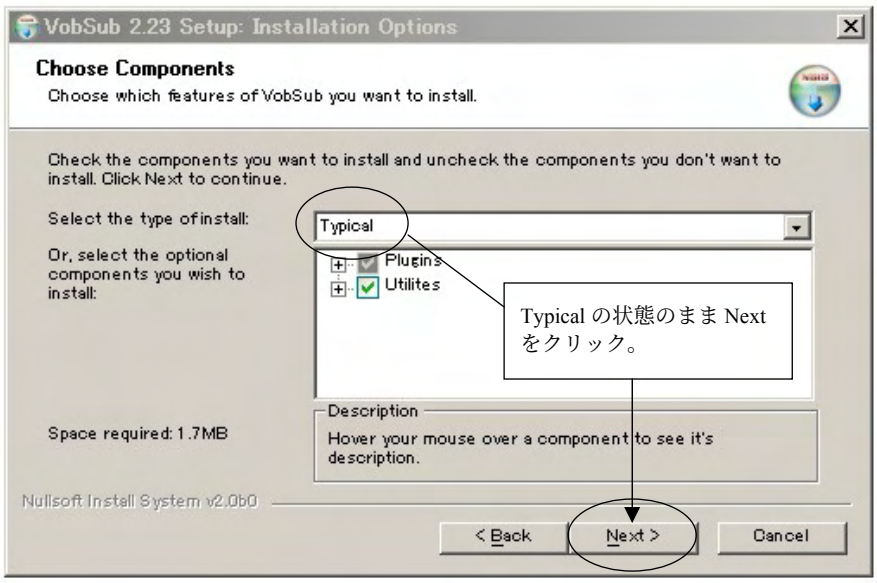


図 11a　VobSub (DirectVobSub) のインストレーション画面 1

6) 上記の状態で Next ボタンをクリックすると、**図 11b** の画面になり、ここでインストール先を指定。ただし、変更せず、そのまま Install をクリック。

7) 以上でインストレーション作業終了。Installation Complete のメッセージを確認し、Close ボタンをクリックして画面を閉じる。なお、作業環境によってはインストレーション完了画面で Close をクリックしても画面が閉じないことがある。この場合は（Ctrl+Alt+Del キーを押して）Windows のタスクマネージャを立ち上げ、アプリケーションを強制終了させてかまわない。

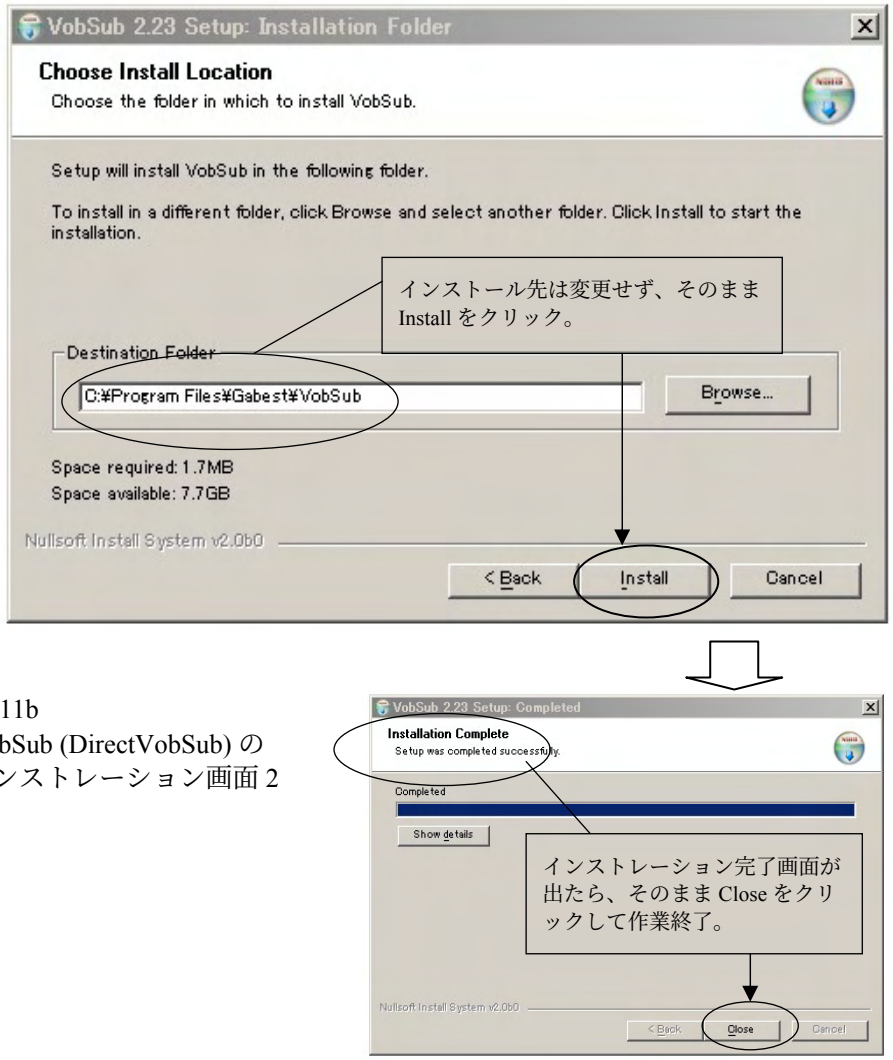


図 11b
VobSub (DirectVobSub) の
インストレーション画面 2

3.3.2　インストールの確認および初期設定

VobSub 2.23.exe のインストールが正常に完了すると、C ドライブの Program Files フォルダの中に **Gabest** という名称のフォルダが作成され、その中にさらに **VobSub** というサブフォルダが作成される。このサブフォルダの中に VobSub を構成する各種のプログラム（およびショートカット）がコピーされている [29]。このうち、**DirectVobSub Configure** という名称のショートカットをクリックすると、DirectVobSub のプロパティ設定画面（**図 12**）を呼び出すことができる（または、ウインドウズ･メニューから「スタート」→「プログラム」→ 「VobSub」の順に進

み、DirectVobSub Configure を選択してもよい）。

DirectVobSub のプロパティ設定画面を起動させたら、とりあえず “General” タブにある “Loading” 設定を “Always load” にしておく（**図 12**）。これで、動画が起動すると DirectVobSub が自動的に立ち上がり、対応する字幕ファイルがあればこれを画面上に表示する。なお、DirectVobSub がデフォルトで対応している映像フォーマットは mpg と avi のみで、Windows メディアとして標準の wmv 形式 には対応していない。したがって、wmv ファイルを DirectVobSub で使用するためには、いったんこれを mpg 形式に変換する必要がある。[30)]

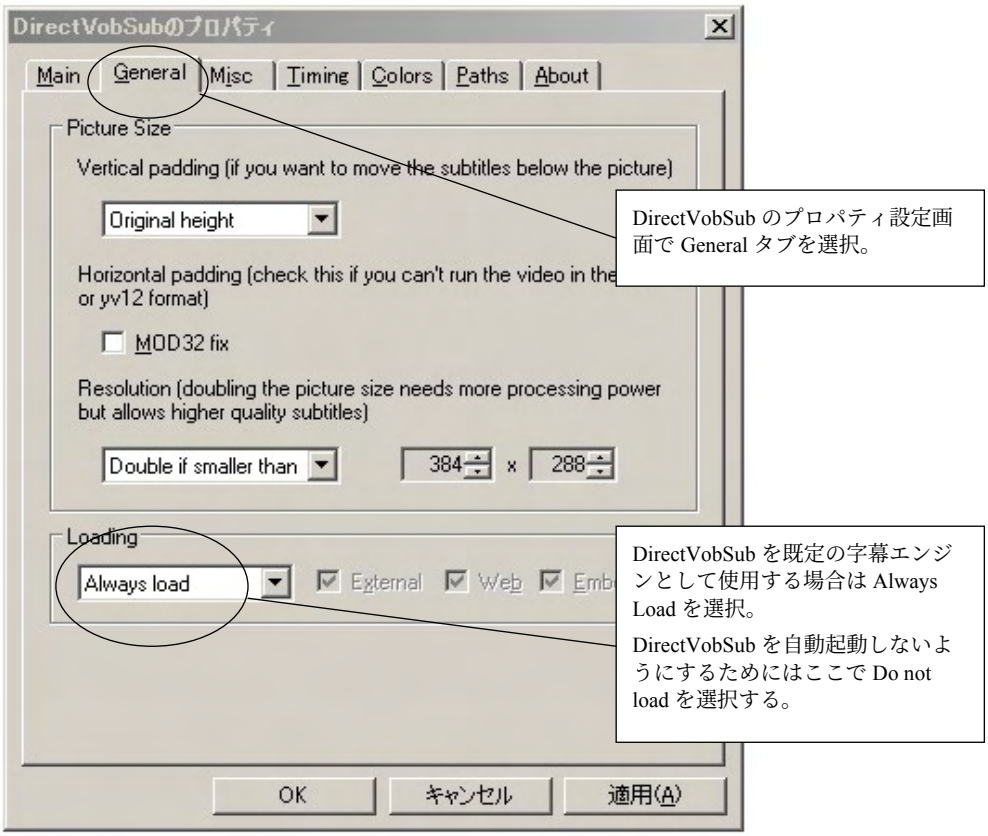


図 12　DirectVobSub のプロパティ設定画面

このほか、DirectVobSub のプロパティ設定で留意するべき点は以下のとおりである。

- 「Main」タブでは Text Settings で字幕に使用するフォントおよび表示スタイルを指定する。日本語字幕の場合は「MS ゴシック」か「MS　P ゴシック」（ただし、フォントがインストールされていれば「HG 丸ゴシック M-PRO」を推奨）を選択したうえで、「文字セット」を「日本語」に設定しておく。適切な指定がされていないと文字化けを起こすので注意。同じタブ内の “VobSub Settings” のところは何もチェックせずにデフォルトのまま使用。
- 「Paths」タブをクリックすると、字幕ファイルを検索するパスを指定することができる

（相対パスと絶対パスのいずれも可）。デフォルトでは動画ファイルのあるフォルダ内を優先的に検索するようになっているため、とりあえずそのままでよい（ただし、動画ファイルと字幕ファイルを別々のフォルダに保存してある場合はパスを適宜変更する）。

- 「Misc」タブの “Picture Output” にある “Hide subtitles” は、DirectVobSub 自体は起動状態のまま一時的に字幕表示をキャンセルする場合にチェックを入れる（したがって、ここではチェックの入っていない状態のままにしておく）。なお、同じタブ内の “Pre-buffer subpictures” はチェックを入れた状態だと動作が不安定になることが指摘されているため、ここはチェックを入れずにおく。

以上の設定が終われば、プロパティ設定画面を終了する。これで、とりあえず DirectVobSub の使用準備は完了したことになる。

3.3.3　DirectVobSub の起動

今回インストールした DirectVobSub は自動起動バージョン (auto-loading version) になっているため、いずれかのメディアプレーヤに avi または mpg (mpeg-1) 形式の動画ファイルが読み込まれれば、これに伴って自動的に起動し（ただし、“Loading” 設定が “Always load” になっているものとする＝**図 12**）、さらに動画ファイルと同じ場所（あるいは前述のプロパティ設定の Paths で指定した場所）に対応する字幕ファイルがあれば、これを読み込んで映像上にオーバーレイ表示する。また、ユーザ側のパソコンに DirectVobSub がインストールしてあれば、動画ファイルと字幕ファイルをオンライン配信してローカルに DirectVobSub を立ち上げることもできる（**注 31** 参照）。

なお、DirectVobSub が起動すると、Windows XP のタスクバー右側の通知領域に次のようなアイコンが表示される。

このアイコン上でマウスを右クリックすると字幕の表示／非表示のコントロールや、使用するフィルタの選択画面が表示され、ダブルクリックでプロパティ設定画面を呼び出すことができる。次ページの**図 13** は WMP で DirectVobSub を起動させた例である。WMP 自体の字幕表示機能では画面下の字幕ペインにしか表示できなかった字幕が、DirectVobSub を使うことで映像上にオーバーレイ表示されていることがわかる。

3.3.4　字幕の 2 重表示防止について

なお、**図 13** のコメント欄にも記したとおり、WMP で DirectVobSub を使う場合は、WMP 自体の字幕機能を「オフ」にした状態で起動させる必要がある（WMP のメニューから「再生」→「キャプションと字幕」→「オフ」を選択）。「オン」の状態（正確には「利用できる場合はオン」の状態）のままだと、**図 14** に示すように字幕が 2 重に表示されてしまうことになる。

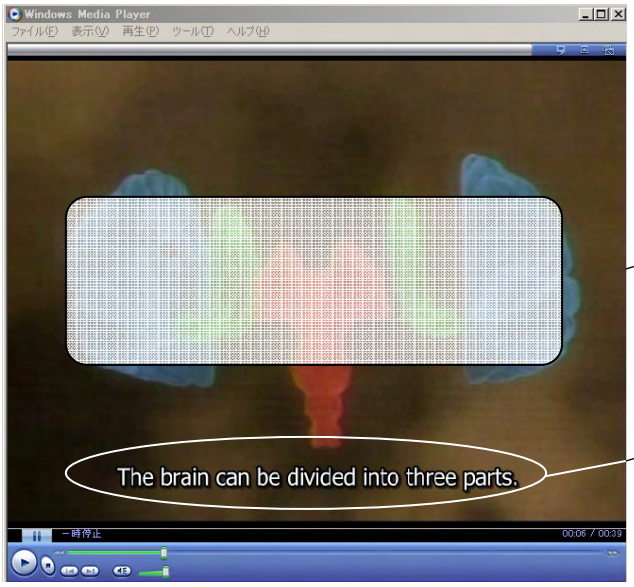


図 13　WMP での DirectVobSub による字幕表示の例

WMP の字幕機能は「オフ」の状態で DirectVobSub を起動（したがって字幕ペインは表示されていない）。

フォントは SAMI ファイル上では Arial, 10 pt の設定。これを Direct VobSub の設定画面から Tahoma, 18 pt（文字セット：欧文: ANSI) に再設定して表示。

図 14　WMP での字幕の 2 重表示

DirectVobSub による字幕表示

WMP 自体の機能による字幕表示（SAMI による書式設定は反映されない）

これは、WMP だけでなく、前述の GOM Player や ViPlay など、それ自体に字幕表示機能を内蔵したプレーヤの場合はいずれも共通した問題である。WMP の場合、DirectVobSub を使うのはオーバーレイ表示をさせるためであり、WMP 自体の字幕機能は「オフ」の状態にしておけばよいのであるが、GOM Player や ViPlay のようにもともとオーバーレイ表示機能を持っているプレーヤの場合は外部字幕フィルタは不要であり、あらかじめ DirectVobSub を無効化しておくのが好ましい。GOM Player で外部字幕フィルタ (この場合は DirectVobSub) を無効化する手順は以下のとおり。

1) GOM Player を立ち上げ、目的の動画ファイルを読み込む。
2) 画面上でマウスを右クリックし、表示されるメニューから「フィルタ情報」→「フィルタのカスタマイズ」（注：バージョンによって「フィルタの高度な設定」という表示になっている場合もある）を選択（**図 15a**）。
3) 「環境設定」画面（**図 15b**） が表示されるので、ここから「フィルタ」→「遮断」タブを選択し、遮断フィルタリストで “DirectVobSub” にチェックマークを入れる（これで DirectVobSub による字幕表示は回避されるようになる）。
4) 「閉じる」をクリックして終了。

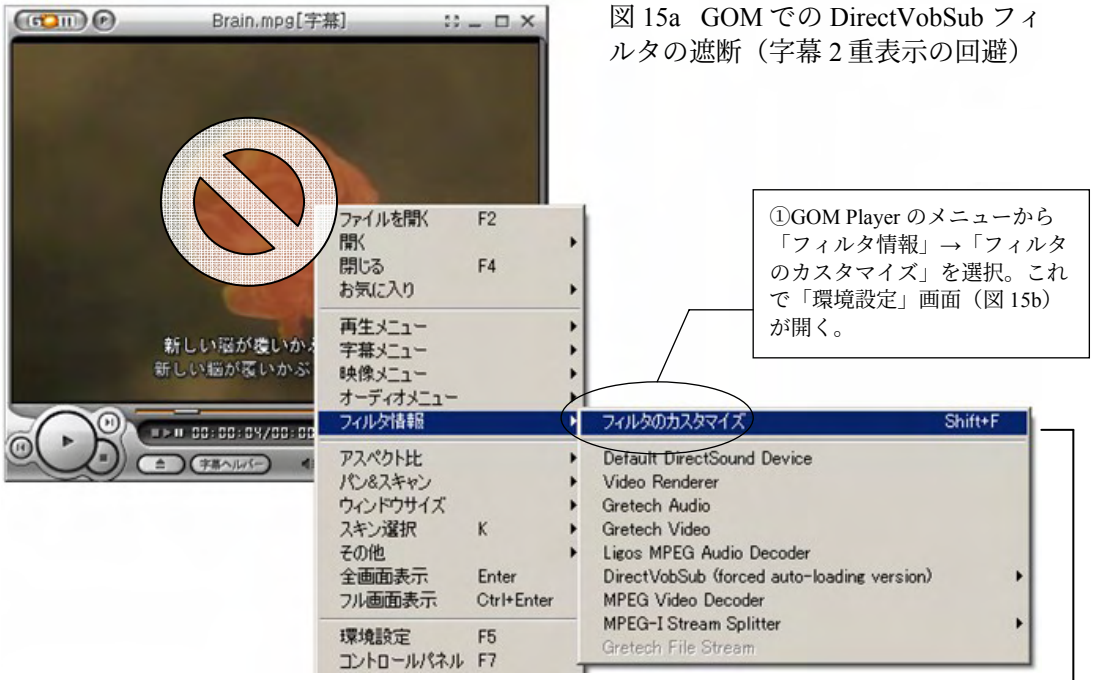
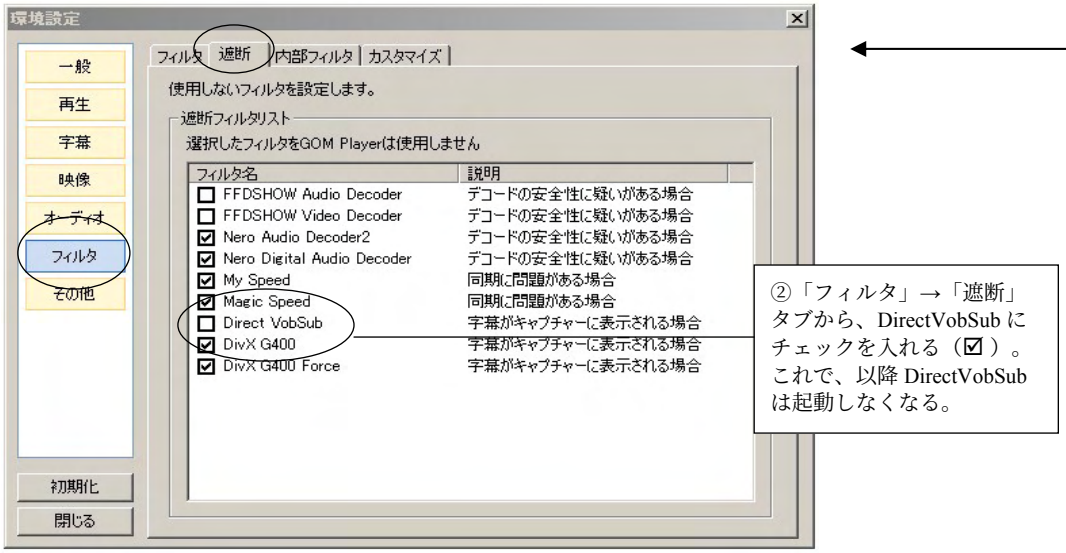


図 15a　GOM での DirectVobSub フィルタの遮断（字幕 2 重表示の回避）

図 15b　GOM **の環境設定画面**→ DirectVobSub **フィルタの遮断**



このほか、DirectVobSub のプロパティ設定画面（**図 12**, p. 24 参照）を呼び出し、ここから直接 DirectVobSub の起動を制御することもできる。この方法を使う場合は、DirectVobSub のプロパティ設定画面の General タブから、Loading で " Do not load " を選択しておく。これで、以降はいっさい字幕を読み込まなくなる（再度 DirectVobSub を使う場合は、改めてプロパティ設定画面を開き、Loading の表示を " Always load " または " Load when needed " に切り替えておくこと）。

なお、字幕の品質はプレーヤ本体の機能を使っても DirectVobSub のフィルタを使っても基本的には同等である。第一義的にはディスプレーの解像度および使用しているパソコンの CPU の性能に左右されるが、画面サイズや縦横比、あるいはフォントの種類によっても多少の違いが出てくるため、設定をいろいろ変えて試してみることをお勧めしたい。

4. まとめ

以上、パソコン上のメディアプレーヤに動画を表示し、これにリアルタイムで字幕を加えるための方法について解説した。すでに明らかなとおり、基本となるのは SAMI ファイル（＝字幕ファイル）の作成である。SAMI ファイルは、ウェブページ作成用の記述言語である HTML と似た構造を持つテキストファイルで、特別なソフトウェアやプログラムを必要とせず、一般的なエディタを使って作成することができる。また、内容の変更や追加等のカスタマイズも容易に行なうことができる。SAMI ファイルの作成に当たっては、いくつかの簡単なルールおよび書式を覚える必要があるが、本稿で解説したとおり、これらのルールはごく単純なもので、覚えるのに特に大きな負担となるほどのものではない。また、実際の作成に当たっては本稿末尾に示した「テンプレート」を使うことで、より効率的に作業を進めることができる。

字幕ファイルの作成が終われば、あとはこれを動画ファイルと同期させてパソコン上のメディアプレーヤに表示させることなるが、本稿ではその方法として 1) Windows Media Player (WMP) の字幕表示機能を使う方法、2) GOM Player や ViPlay のような独自の字幕エンジンを搭載したメディアプレーヤを使う方法、および 3) 字幕表示用プラグインである DirectVobSub をインストールした上で WMP を使って字幕を表示させる方法、の 3 つについて、スクリーンショットを多用して具体的に解説した。どの方法を使うかは、それぞれのユーザの必要と目的に応じてということになるが、基本的には WMP 既定の「字幕ペイン」方式でよい場合は 1) を使い、「オーバーレイ方式」を選択する場合は 2) または 3) を使う、ということになる。以下、参考までに、字幕表示に関する各メディアプレーヤの機能（＝何ができて何ができないか）を一覧表としてまとめておく（**表 2**）。

表 2 に示したとおり、WMP では SAMI ファイルで指定した字幕書式設定は完全に反映されるが、その代わりにプレーヤ上からオンラインで字幕の表示スタイルを制御することができない。一方、GOM Player と ViPlay では原則としてプレーヤ上で字幕表示をオンライン制御することを想定しており、SAMI ファイルでの字幕書式設定は基本的なもの以外は反映されない。

表 2　字幕表示に関する各メディアプレーヤの機能比較 ※

	メディア形式		字幕表示方式		SAMI ファイルへの対応	プレーヤ上でのオンライン字幕表示制御	ウェブページへの埋め込み	縦書きの字幕表示
メディアプレーヤ／プラグイン	WMV	MPG, AVI	字幕ペイン	オーバーレイ				
WMP	○	○	○	×	○	×	○	○
WMP＋DVobSub	×	○	×	○	△	○	○	×
GOM Player	○	○	×	○	△	○	×	×
ViPlay	○	○	×	○	△	○	×	×

※WMP= Windows Media Player; DVobSub = DirectVobSub (字幕表示用プラグイン)

これは DirectVobSub を使った場合も同じである。上記の表中「SAMI ファイルへの対応」という項目で WMP 以外はすべて△ になっているのは、この事情を反映したものである。したがって、WMP 以外のメディアプレーヤを使用する場合は、SAMI ファイル中でのスタイルシートによる書式設定は最小限のものでよい（**注 21** 参照）ということになる。

以上の方法は、いずれもユーザがそれぞれ自分のパソコン上でメディアプレーヤを立ち上げ、この上で動画と字幕を同期表示させるというもの（＝スタンドアロン方式）であるが、このほかに、ウェブページにメディアプレーヤを埋め込み、インターネット上で字幕つきの動画を配信することも可能である。前掲表中の「ウェブページへの埋め込み」という項目はこれを指している [31)]。字幕つきの動画をインターネット上で配信することができれば、例えば学生が自分の作品を他の学生と共有したり、著作権上の問題がクリアされていることを前提に [32)]、これを広く一般に公開することも可能になる。

また、ウェブページにメディアプレーヤを埋め込むことで、字幕の表示方法についてもさまざまな工夫が可能になってくる。例えば、字幕テクストの文字サイズをユーザが自由に選択できるようにしたり、CSS の writing-mode プロパティを使って日本語の字幕を「縦書き」表示させることもできる。また、必要に応じて、カラオケのように「チャンク」ごとに動的に字幕を表示したり、現在位置を色や下線、あるいは背景色の移動などで示すといったことも可能である。後者のような特殊な表現方法は、字幕そのものが焦点化されてしまうために、通常の映画やドキュメンタリー等の鑑賞用には向かないが、語学教育の場面ではさまざまな応用が可能である。このようなさまざまな工夫を実現するための方法については、前述の字幕付き動画のインターネット配信という課題と合わせて、別稿 [33)] で改めて述べることとしたい。

資料 1　SAMI 字幕ファイル テンプレート 1（日本語字幕用）<SAMI_Template1.smi>[34)]

```
<SAMI>
<HEAD>
  <TITLE>SAMI 字幕ファイル テンプレート（日本語字幕用）</TITLE>
  <STYLE TYPE="text/css"><!--
    P {margin-top:10pt; text-align:center;
       font-size:10pt; line-height:125%;
       font-family: Arial,Verdana,Helvetica,'ＭＳ ゴシック','ＭＳ Ｐゴシック',
      sans-serif; color:white; background-color:transparent;}
    .JACC {name:Japanese; lang:ja; SAMIType:CC;}
     -->
  </STYLE>
</HEAD>
<BODY>
<!-- 映像の開始時間を START=0 で指定し、空白行を置く-->
   <SYNC START=0>
      <P CLASS=JACC> 
<!-- 字幕１以降は START=**** に各字幕の実際の開始時間をミリセコンド単位で記入 -->
<!-- 例：1 秒＝1000；10 秒=10000; 1 分(60 秒)=60000; 10 分(600 秒)=600000 -->
   <SYNC START=****>
      <P CLASS=JACC>字幕１
   <SYNC START=****>
      <P CLASS=JACC>字幕２
   <SYNC START=****>
      <P CLASS=JACC>字幕３
   <SYNC START=****>
      <P CLASS=JACC>字幕４
   <SYNC START=****>
      <P CLASS=JACC>以下、必要なだけタグセットをコピーして最終行まで字幕を作成
<!-- 最終行。START 時間は動画の実際の終了時間と一致させる -->
   <SYNC START=****>
      <P CLASS=ENCC> 
      <P CLASS=JACC> 
</BODY>
</SAMI>

<!-- 注記（この注記は字幕ファイル作成時に削除してください） ====================================
(1) GOM Player を使うときは SAMI ファイルのスタイル設定は無視されます（字幕スタイルは GOM の本体でコントロール。ただし、各字幕行内での設定は有効）。
(2) 字幕と字幕の間に空白行を挿入する場合は上記、最終行に示した段落タグセットと同一のタグセットを挿入します。これで、****で指定された時間から次の字幕が始まるまでの間、字幕表示のない画面が展開されます。
(3) 作成した SAMI ファイルは、対応するメディアファイルと同じ名称で保存し、拡張子を.smi とします。
(4) 保存場所は原則としてメディアファイルと同じフォルダとします。
(5) ウェブページに埋め込んでオンライン配信する場合はメディアファイルと SAMI ファイルの保存先／アクセス先を「URL SAMI 構文」を使ってフルパスで指定します（詳しくは染谷(2008b)を参照してください）。
-->
```

※スクリプト末尾の注記 (6) はスペースの都合上、省略しています。SAMI_Template2.smi を参照してください。

資料 2　SAMI 字幕ファイル テンプレート 2（日英 2 ヵ国語用）<SAMI_Template2.smi>

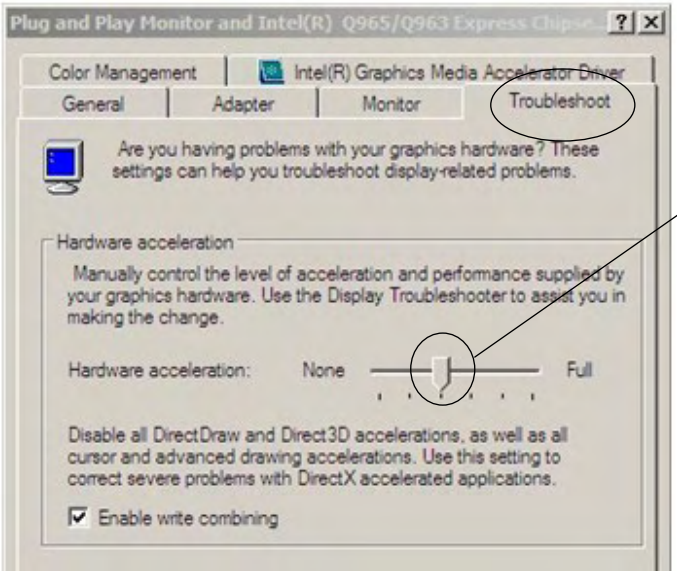
```
<SAMI>
<HEAD>
  <TITLE>SAMI 字幕ファイル テンプレート（日英 2 か国語用）</TITLE>
  <STYLE TYPE="Text/css"><!-- 言語ごとに異なったスタイル指定をする場合、共通のものだけを段落
  要素で指定し、異なる要素は言語クラスごとに指定-->
  <!--
    P {margin-top:10pt; text-align:center; line-height:125%;
       color:white; background-color:transparent;}
    .JACC {name:Japanese; lang:ja; SAMIType:CC;
           font-family: Arial,Verdana,Helvetica,'ＭＳ ゴシック','ＭＳ Ｐゴシック',
          sans-serif; font-size:10pt;}
    .ENCC {name:English; lang:en-US; SAMIType:CC;
           font-family:Arial,Verdana,Tahoma,Helvetica; font-size:11pt;}
    -->
  </STYLE>
</HEAD>
<BODY>
<!-- 映像の開始時間を START=0 で指定し、空白行を置く-->
   <SYNC START=0>
      <P CLASS=JACC> 
      <P CLASS=ENCC> 
<!-- 字幕１以降は START=**** に各字幕の実際の開始時間をミリセコンド単位で記入。例：1 秒＝1000；10 秒
=10000; 1 分(60 秒)=60000; 10 分(600 秒)=600000 -->
   <SYNC START=****>
      <P CLASS=JACC>日本語字幕１
      <P CLASS=ENCC>English Subtitle Line 1
   <SYNC START=****>
      <P CLASS=JACC>日本語字幕２
      <P CLASS=ENCC>English Subtitle Line 2
   <SYNC START=****>
      <P CLASS=JACC>以下、省略...
      <P CLASS=ENCC>Omitted...
<!-- 最終行：START 時間は動画の実際の終了時間と一致させる -->
   <SYNC START=****>
      <P CLASS=ENCC> 
      <P CLASS=JACC> 
</BODY>
</SAMI>

<!-- 注記（この注記は字幕ファイル作成時に削除してください） ====================================
(1)～(5) 省略（SAMI_Template1.smi 参照）
(6) 文字数の基準は以下のとおり（ただし、必要に応じて適宜調整する）。
    1) 日本語の場合：１秒当たり４～５文字、１行当たり最大 13 文字（縦書きの場合は 10 文字）で２行まで。
    2) 英語の場合：１秒当たり 12 文字を標準とし、1 行当たり最大 32 ないし 33 文字で、同じく２行まで。字幕
       の提示速度についてはおよそ以下のとおり（英語母語話者を対象とした場合の基準）：
       教育用または子供向けのビデオ＝120 WPM (words per minute)
       大人向けの教育用ビデオ＝150-160 WPM
       大人向けの娯楽映画＝原則として 235 WPM を超えない範囲でほぼ実際の発話の速度に準ずる。
       (120WPM = 2 words/sec.; 180WPM = 3 words/sec.; 235WPM = 3.9 words/sec.)
-->
```

補足資料：ハードウェア・アクセラレーションと画像表示

Windows Media Player や Windows Movie Maker でメディアファイルを再生したときに、音声は聞こえるのに映像が表示されないことがある。また、映像をイメージカッターで切り取ったあと、画像ファイルを開いてみると画面が真っ黒になっていることがある。このような、グラフィックハードウェアに由来するディスプレイ関係の問題は、通常、Windows の「ハードウェア・アクセラレーション」のレベルを下げる（または解除する）ことで解決することができる。手順は以下のとおり（以下は Windows XP の英語版を使用。日本語版でも操作手順は同じ）。

1) Windows の[Start] から [Settings] →[Control Panel] →[Display] を選択。
2) さらに、[Display Property] 設定画面から[Settings] →[Advanced] →[Troubleshoot] を選択し、以下の画面を表示。
3) [Hardware acceleration] のスライドバー設定が Full になっている場合はとりあえず左から 3 つ目の目盛り位置 (= Disable DirectX acceleration) まで レベルを下げる。これでも現象が改善しない場合はさらにうひとつレベルを下げる。



[Hardware acceleration] のスライドバー設定が Full になっている場合はとりあえずこの位置まで レベルを下げて、駄目ならもうひとつレベルを下げる。

注）[Hardware acceleration] のレベルを下げると、字幕の質が低下します。画像表示にとくに問題がなければ “Full” の状態で使用してください。

画像が表示されていない状態

画像が表示された状態

【注】

1) 例えばパワーポイントのスライドを静止画像として保存し、これを「Windows ムービーメーカー」などのビデオ編集ソフトを使って動画化することで、（静止画像をつなぎ合わせた）スライドショービデオを作製することができる。パワーポイントのファイルを動画にする方法については別稿（染谷 2008b, p. 27) で解説する。

2) プラグイン (plug-in)：アプリケーションソフトに新たな機能を追加するための小さなプログラムのこと。アドインともいう。DirectVobSub については本文 (pp. 21-25) 参照。

3) SAMI はインターネットアプリケーションのためのアクセシビリティ技術のひとつとして開発されたもので、もともとは聴覚障害者等のインターネット利用を支援するためのものである。SAMI の詳細情報については次のサイト参照（以下、本稿に掲載する URL はいずれも 2008 年 6 月 30 日現在のものである）。

- “Understanding SAMI 1.0” (Originally published October 2001; updated February 2003)
  [Online] http://msdn.microsoft.com/en-us/library/ms971327.aspx
- 「SAMI 1.0 の理解」（日本語版最終更新日 2001 年 11 月 15 日）
  [Online] http://msdn.microsoft.com/ja-jp/library/cc421567.aspx

なお、SAMI のほかに、字幕ファイルの標準的な仕様規格として SMIL (Synchronized Multimedia Integration Language) と呼ばれるものがある。似たような名称で紛らわしいが、アップル社の QuickTime Pro やリアルネットワークス社の RealPlayer に搭載されている字幕機能は、基本的にはこの SMIL（"smile"と発音）の規格に準拠したものである（詳しくは W3C (1998-2003) 参照）。ただし、字幕ファイルそのものはそれぞれ独自の形式で作成した上で、SMIL ファイルをポインターとして使用するという形になる。いずれにせよ、本稿で取り上げる SAMI ファイルとの互換性はないことから、本稿では QuickTime および RealPlayer を使った場合の字幕作成については取り扱わない。関心のある方は、以下のウェブサイト (WebAIM, Utah State University) に詳しい解説が掲載されているので、こちらを参照されたい。

- http://www.webaim.org/techniques/captions/quicktime/
- http://www.webaim.org/techniques/captions/real/

4) このほか、以下のような形式で著作権やメディア情報、計測単位などの情報を「パラメタ情報」としてヘッダ部に記述することができる。ただし、現時点では <SAMIParam> タグはサポート対象外となっているため、あくまでも作成者側のメモ代わりとして使うことになる（したがって省略しても差し支えない。ただし、この形式で自由にコメントを記入することができるため、各種の覚え書をメモしておくには便利な機能である）。

```
<HEAD>
<TITLE>Sample SAMI File</TITLE>
<SAMIParam>
  Copyright {(C) Copyright 2008, Taro Yamada}
  Media {Sample.wmv}
  Metrics {time:ms; duration: 60000;}
</SAMIParam>
<STYLE TYPE="text/css">
```

<SAMIParam> 情報の記述例 (オプション)

このうち、Metrics は同期の時間単位と継続時間を指定する。Microsoft DirectShow および WMP ではミリ秒 (ms) をサポートしているため、時間単位は ms（1 秒=1000 ms）とする。したがってメディアの継続時間が 1 分の場合の duration は 1000 (1×60) = 60000、3 分の場合は 1000 (3×60) = 180000 となる。

5）該当箇所を `<!--` と `-->` で括ることを「コメントアウト」（注釈化）という。JavaScript ではコメント行のエンドタグとして `// -->` を使うことになっているが、HTML および CSS では `-->` の前にダブルスラッシュ (//) を記述してはいけないことになっているので注意のこと。

6）詳しくは WC3 のウェブサイト (http://www.w3.org/TR/CSS21/) 参照。記述上の要点のみをざっと調べるためには以下のサイトを参照。

- 「CSS スタイルシート―TAG index」 (http://www.tagindex.com/stylesheet/)
- 「とほほのスタイルシート入門」 (http://www.tohoho-web.com/css/)
- 「HTML クイックレファレンス」**>**「スタイルシートリファレンス」 (http://www.htmq.com/style/)

7）相対単位の em （エム）は当該ファイルの中で使われている標準フォントの大文字 “M” の高さを、ex（エックスハイト）は小文字の “x” の高さをそれぞれ基準としたサイズ。2em という指定で基準サイズの 2 倍、3em で 3 倍の大きさになる（em の使用例については、染谷 2008b, pp. 37-38 参照）。ピクセル (px) はその実測値がディスプレーの解像度に依存するという意味での相対的な単位となる（ピクセルについては染谷 2008b, **注 7** 参照）。パイカは 1pc = 12pt の換算になる。ポイント (pt) という単位は、1 ポイント＝72 分の 1 インチ（約 0.35mm）として計算する。したがって、10pt は約 3.52mm、12pt は約 4.23mm となる。

8) フォント名にスペースが含まれる場合は、フォント名をダブルクォーテーション (`"`) またはシングルクォーテーション (`'`) のいずれかで囲むことになっているが（例：`"ＭＳ ゴシック"` または `'ＭＳ ゴシック'`）、インライン要素としてスタイル指定をする場合に、フォント名をダブルクォーテーションマークで囲むと、例えば `<span style="font-family:"ＭＳ ゴシック""> ` のように末尾の引用符が二重になって機能しないことがあるため、シングルクォーテーションで統一したほうがよい。

なお、font-family で複数のフォントを指定した場合、先にあるものが優先して表示される。例えば、`font-family: Arial,'ＭＳ ゴシック',sans-serif;` という指定では、まず該当する文字列が半角英数字であれば Arial で表示され、全角の仮名漢字であればパスされて次の 'ＭＳ ゴシック' が適用される。これらのどれにも当てはまらなかった場合は最後の sans-serif が適用される。sans-serif とはフォントの種類を示す総称で、「ゴシック系のフォント」という意味である。このように、最後に総称ファミリ名を加えておくと、指定したフォントが全て使えなかった場合にも対処できるようになる。

すでに述べたとおり、本稿では原則として Windows プラットフォームでの作業を前提としているが、Windows と Mac の両プラットフォームに対応させるためには、次のように、Windows 専用のフォント（Arial, Verdana, 'ＭＳ ゴシック'）と、Mac 専用のフォント（Osaka, Helvetica）を、半角英数字→全角仮名漢字の順に並べた上で、最後に sans-serif を加えるのがよい。

`font-family: Arial,Verdana,Helvetica,'ＭＳ ゴシック',Osaka,sans-serif;`

このうち Osaka は 'ＭＳ ゴシック' に、Helvetica は Arial にそれぞれ対応する。Arial と Verdana はほぼ同等のゴシック系英文フォントであるが、後者のほうがやや幅広で、字幕として見る場合は Arial よりもいくらか見やすくなる。ちなみに、Mac OS 用の標準ブラウザである Safari では、フォント名の中に日本語が含まれているものは無効になるため 'ＭＳ ゴシック' の代わりに 'MS Gothic' と半角アルファ

ベットで記述する必要がある。ただし、無効になっても、上記の例のように記述してあれば Osaka または sans-serif のいずれかでフォローされるので、通常は問題はない。

9)　カラーネームおよびカラーコード一覧について http://www.someya-net.com/ColorIndex.html を参照。なお、次項の background-color というプロパティは文字列の背景色を指定するもので、字幕表示画面の背景色の指定ではないことに注意。たとえば、background-color を lightgray (#D3D3D3) にすると、文字列はこのように表示される。Windows Media Player の字幕表示画面はデフォルトで黒になるので、黒字に白の文字列を表示する場合には background-color の指定を既定値の transparent（透明色）にする（あるいは black を指定しても同じ結果になる）。

10）次のサイト（一松書院）にはハングル文字を表示させる場合の方法について詳しい解説が掲載されている（2008 年 5 月 22 日現在）。http://www10.tok2.com/home/madang/memo/1010/memo.cgi

11）ViPlay (ダウンロードサイトは**注 22** 参照) を使う場合は英語と日本語（またはその他の言語）の字幕ファイルを個別に作成し、字幕ファイルの選択によって表示言語を使い分ける。字幕ファイルの読み込みおよび選択の手順は以下のとおり。

1. ViPlay を起動し、目的の動画ファイルを読み込む。（注：字幕ファイルへの関連付けがしてある場合は自動的に字幕ファイルが読み込まれる。したがって、対応する字幕ファイルがひとつしかない場合には、以下の作業は不要である）。
2. 画面上でマウスを右クリックし、表示されるメニューから、[サブタイトル] → [ファイルを開く] を選び、目的の言語の字幕を含むファイルを読み込む（仮に、ここでは Brain.smi, Brain-biling.smi, Brain_EN.smi という 3 つのファイルを読み込ませおくこととする）。
3. 次に、再度、画面上でマウスを右クリック。表示されるメニューから [サブタイトル] → [言語] を選択。
4. **図 5** の画面イメージに見られるとおり、この時点で読み込まれている字幕ファイルの一覧が表示されるので、目的の言語の字幕を含むファイルを選択（注：字幕を表示させるためには、あらかじめこの画面で [サブタイトルを表示] をチェックしておくこと）。
5. 以上で動画と字幕ファイルの読み込みが終了。プレーヤの再生ボタンをクリックすると動画の再生が開始される（他の言語による字幕を表示させたい場合は、上記 4）に戻り、必要な字幕ファイルを指定する）。

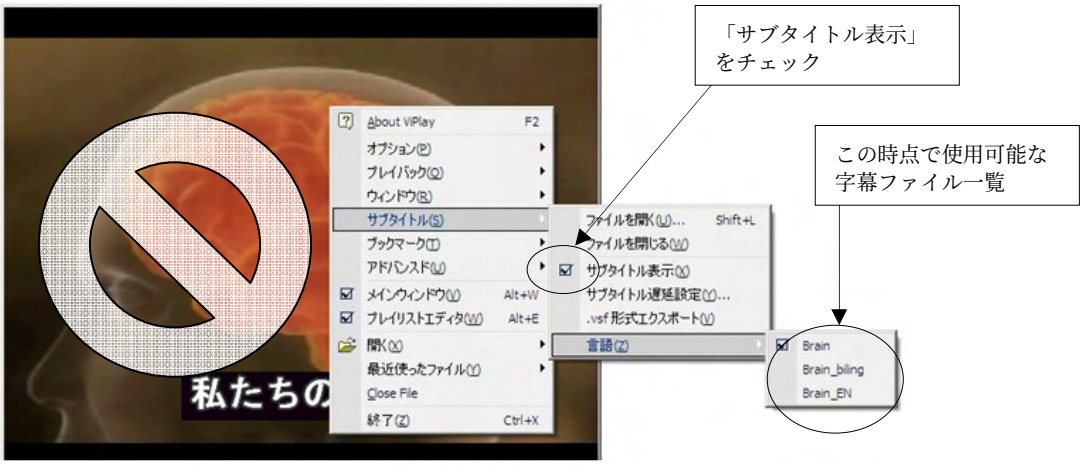


図 5　ViPlay での字幕ファイル選択パネル

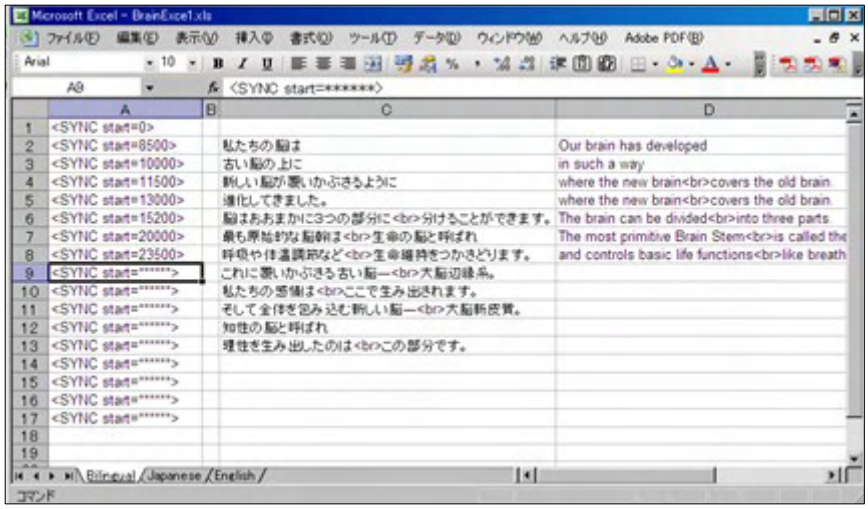
12）“START=0” の位置に字幕を配置すると映像の再生開始と同時にその字幕が表示されることになるが、字幕ファイルの読み込みが完了した時点で映像ファイルの読み込み終了していない場合（通常は映像ファイルのほうが読み込みに時間がかかる）、映像の表示に先立って字幕のみが表示されてしまうことがある。これを防ぐためには “START=0” の位置は常にカラ行としておくのがよい。

13）色指定については p. 7 および**注 9** 参照。

14）<font> タグ内でのフォントサイズは、最小値 1 から最大値 7 の範囲で指定する（標準は 3 で、何も指定しない場合はこの値が HTML ページのデフォルト値となる）。このほか、size="+1" や size="-1" のように、現在の大きさに対する相対値を指定することもできる（この場合はダブルクォーテーションで囲む）。

15） ただし、字幕を表示させるタイミングをより厳密に指定する必要がある場合や、何らかの理由で SAMI 形式以外のフォーマットで字幕ファイルを作成する必要がある場合は、字幕作成専用のソフトウェア（＝字幕エディタ）を使う。現在、入手可能なフリーの字幕エディタとしては Subtitle Workshop, ChronoSub, Subbu, SubtitleProcessor, SubCreator, DivXLand Media Subtitler などさまざまなものがあるが、これについては改めて別稿（染谷 2008d）で解説する。

16）字幕の分量が多くなる場合や、グループプロジェクトとして手分けして字幕を作成する場合は、エディタよりもエクセル (MS Excel) を使ったほうがより効率的に作業を進めることができる。とくに、翻訳字幕を作成する場合は、以下の例に示すとおりエクセル上で第 1 字幕と第 2 字幕を行ごとに対照させて作業を行うようにするのがよい。



17）ただし、HTML ページに WMP を埋め込み、字幕に同期させた映像をオンライン配信する場合は、サーバ内の異なる場所に動画ファイルと字幕ファイルを置いておくことができる（染谷 2008b, p. 4 参照）。

18） p. 9 に掲載した**図3** は、すでに SAMIファイルが読み込まれたあとの言語選択画面を示す。これに対して、**図7** は SAMI ファイルが読み込まれる前の状態を示す。**図3** と**図7** の表示内容が異なるのはそのためである。

19) WMP のように映像画面と字幕表示エリアが分かれている場合、映像を隠すことなしに多くの情報を表示できるという利点がある。前述のとおり、SAMIはもともと聴覚障害者のためのアクセシビリティ技術のひとつとして開発されたものであり、このような字幕表示方式はその目的に照らしてごく正当

なものであると思われる。また、語学教育の一環として字幕翻訳を取り入れる場合にも、この方式のほうが（画面上に字幕をかぶせる）オーバーレイ方式よりも、かえって都合がよい場合も考えられる。ただし、字幕つき映像としてのプロフェッショナルな仕上がりという観点からすると、オーバーレイ方式のほうが好ましいように思われる。その代わり、できるだけ映像の邪魔をしないように必要最小限の文字数に抑える必要があり、情報量は少なくなる（もっとも、これは必ずしも欠点ではなく、そのためにより創造的な字幕作成が要求されるという意味で、教育的な観点からはむしろ利点としてとらえることができる）。いずれにせよ、字幕ペイン方式とオーバーレイ方式はそれぞれ目的に応じて使い分ければよいということだろう。

20) wmv と avi はいずれもマイクロソフト社の規格で、WMP 標準のフォーマット。mpg (mpeg) にはいくつかバージョンがあり、WMP が標準で対応しているのは mpeg-1 と呼ばれるもの。一般の DVDムービーなどで使われている高圧縮フォーマットの mpeg-2 は標準の WMP では未対応で、別途、コーデックが必要。なお、mpg と mpeg は同じものを指す（mpg はかつて拡張子が 3 文字に制限されていた時代に使われていたもので、現在もそのまま使われている）。

　ちなみに、mpg/mpeg は、拡張子だけでは 1 と 2 の区別がつかないが、「真空波動研」というフリーウェア（WMP Ver. 6 ベースのメディアプレーヤ）を使うと、動画ファイルの種類を判別してくれるだけでなく、その動画がどのような環境でコーディングされたものかについて詳細な情報を表示してくれる。コーデック･チェッカーとして十分な機能を持っており、インストールしておくと便利。

ダウンロードサイト： http://kurohane.net/seisanbutu.phtml

21）GOM Player では SAMI ファイルのスタイルシートに記載された書式設定は基本的には無視される（ただし、各字幕行中での HTML タグによる色指定と改行は反映される）。したがって GOM を使うことを前提とした場合は SAMI ファイルでのスタイル設定は省略（または簡素化）してかまわない。

22) ViPlay は URU Soft 社のウェブサイト（http://www.urusoft.net/）から無料でダウンロードすることができる。なお、最新の ViPlay (Ver. 3) には字幕表示に関するメニューが見当たらないことから、バージョンアップに伴って字幕表示機能が削除された可能性がある（あるいはインストールの失敗か?）。Subtitle Workshop およびこれを使った授業実践については稲生 (2004) でも報告されている。技術的な詳細については上記の開発元ウェブサイト参照。

23) 担当は英米文学科の稲生衣代准教授。

24) 前者は Closed Caption に、後者は Open Caption にそれぞれほぼ対応すると考えてよい。なお、closed という名称は、必要であれば非表示 (closed) にできることから。

25) GOM Player を配布しているグレテック社が 2007 年 2 月に発表した GOM Encoder というソフトウェアを指す。GOM Encoder は基本的には動画フォーマットの変換プログラムであるが、本格的なハードサブ作成の機能も備えており、作成したハードサブ字幕付き動画はどのプレーヤでも再生可能である。現在は機能が限定された無料のベータ版（試用版）がリリースされているが、「正式バージョンは有料で販売する予定」とのことである。動作環境は OS = Windows 2000/XP/2003/Vista, CPU= Pentium４以上を推奨。アクセスサイト：http://www.gomplayer.jp/

26) VobSub はもともと VirtualDub というオープンソースプログラム用のプラグインとして開発されたらしく、“VirtualDub Plug-in” ということになっている。VirtualDub (http://www.virtualdub.org/) は、32 ビットウインドウズプラットフォーム対応の avi ビデオ編集および DVD からの字幕データのリッピング

等を行うためのユーティリティで、VobSub に含まれている各種フィルタ（*.vdf）と組み合わせることで、ハードサブおよびソフトサブ双方の方式での映像への字幕付与が可能になる。なお、染谷 (2008c) では VobSub に含まれる TextSub (textsub.vdf) というプラグインを VirtualDub に組み込んだ上でハードサブ映像を作成する方法について、詳しく解説している。VobSub のより高度な使い方について関心のある方は、次のオンライン･マニュアル（英語版）を参照していただきたい。

- “VobSub Subtitle Ripping, Playback and Permanent Subtitle Guide” (http://www.divx-digest.com/articles/article_vobsub_ page1.html) 。

また、VobSub 2.23 の日本語化パッチファイルを以下のサイトで入手することができる。

- 『VobSub 2.23 ガイド』 (http://homepage2.nifty.com/sophia0/freesoft/VobSub/)

27) DirectShow は、マイクロソフト社の DirectX（WMP の機能を強化するために提供している拡張 API [Application Program Interface] 群）に含まれる機能の 1 つで、メディアの再生、キャプチャ、フォーマット変換などさまざまな機能が用意されており、「フィルタ」というプラグインを追加することで機能拡張が行える。DirectX を使うと、アプリケーションソフトが統一的な手法を用いて直接ハードウェアを制御することが可能になるため、ゲーム機などの専用ハードウェアに負けない高度なマルチメディア処理をパソコン上で実現することができる。

28) とりあえず今回必要なものは [Plugins] の中に含まれている “VobSub for DirectShow” (= Direct VobSub) のみ。初期設定を変更するとインストールがうまくいかないことがあるので注意（このほかのプラグインについては別稿（染谷 2008c）で改めて解説する）。なお、DirectVobSub (本体は DVobSub.ax) は、VSFilter (VSFilter.dll) と名称を変えて再配布されており、こちらのほうが最新版とのことであるが、両者とも同じシステムフォルダにインストールされた状態で使うと競合して「深刻なエラー」が発生する可能性があること、および DVobSub.ax のままでとくに問題なく字幕の読み込みができることから、このあたりの技術的な詳細について十分な知識がある場合は別として、基本的には従来版の DirectVobSub をそのまま使っておく（つまり VSFilter はインストールしない）のがよい。

29) 同時に、C ドライブの WINDOWS システムフォルダ (system32) の中に `dvobsub.ax` および `vobsub.dll` という 2 つのファイルがインストールされる。このうちの前者が DirectVobSub の本体であり（ax という拡張子はこれが DirectShow フィルタであることを示す）、DirectVobSub Configure というショートカットはこのファイルにリンクされているものである。もうひとつの `vobsub.dll` は VobSub 用のデータライブラリである。ちなみに、何らかの理由で DirectVobSub を一時的に無効化したい場合は、`dvobsub.ax` (→ C:\WINDOWS\system32 \dvobsub.ax) を、例えば `xdvobsub.ax` のようにファイル名の先頭に適当な文字や記号を加えて一時的に名称変更しておくことで、プログラムの実行そのものをキャンセルすることができる。なお、仮に必要なファイルを消してしまったような場合、あるいは何らかの理由で DirectVobSub（または DirectVobSub の環境設定パネル）が立ち上がらなくなった場合は、VobSub 2.23 を再インストールすればよい。

30) 面倒ではあるが、映像を扱う以上、フォーマットの変換は避けて通れない問題であり、いつでも必要に応じて変換作業ができるように準備しておきたい。映像フォーマットを変換するためのソフトウエアは無料で高機能のものがインターネット上でたくさん出回っており、いずれか適当なものを入手しておくとよい。以下、いくつか参考になるウェブサイトを挙げておく。

- http://www.gigafree.net/media/conv/

- http://www.softnavi.com/douga.htm
- http://download.goo.ne.jp/software/category/win/art/movie/conv/

ちなみに、操作が簡単でお勧めの動画変換ソフトは **Free Video Converter V1.3**（フリーウエア）。操作メニューの日本語化パッチもあらかじめ同梱されているので、インストールしてすぐに使える（ダウンロード先：http://www.koyotesoft.com/ndexEn.html）。このほか、MPEG エンコーダとして定評のある **TMPG Encoder** もダウンロードしておきたい。本来は有償だが、制限付きの試用版が無料でダウンロード可能。エンコードに時間がかかるが、たいへんきれいな出力が得られる。また（裏技として）DirectVobSub と併用して wmv 形式の動画に字幕をエンコードしながら mpeg ファイルに出力することもできる。詳しくは染谷 (2008c, pp. 22-27) 参照。

31) ウェブページへの埋め込みは、本稿でとりあげたメディアプレーヤの中では原則として WMP のみの機能であるが、DirectVobSub が ローカルにインストールされていれば、インターネットで配信された mpg (mpeg-1, 2) または avi 形式の動画を（ウェブページに埋め込まれた）WMP で再生し、その画面上に字幕をオーバーレイ表示させることができる。インターネットで配信された字幕付き動画を DirectVobSub を使ってローカルで再生するための方法および注意点等については染谷 (2008b, p. 31-35) で改めて解説する。

32) 授業で使う映像素材は、著作権切れの映画等を含む “Public Domain” のものを使うのが望ましい。“Public Domain” の素材については、About.com の “Free Movie Downloads -- The Best Web Sites for Free Movie Downloads” (http://websearch.about.com/od/ searchtipoftheday/ qt/freemovie.htm) というページに本格的な映画からショートフィルム、ドキュメンタリー、アニメーションまでさまざまな動画素材の一覧が掲載されている。なお、米国映画をめぐるいわゆる「53 年問題」というのがあった。これは「ローマの休日」や「シェーン」など、米国映画史上の名作とされる作品の多くが 2003 年度をもって著作権上の保護期間（50 年）を終了し、パブリックドメイン化する——と思いきや、これにあわせて著作権法が改正され（2003 年 6 月 19 日交付、2004 年 1 月 1 日施行）、保護期間が 70 年に延長されたことに端を発するもので、この改正法により、1953 年以前の作品は保護期間＝50 年が過ぎたため、著作権切れ＝パブリックドメイン化が確定し、一方、1953 年以降の作品については改正著作権法が適用され、公開後 70 年間は保護下にある、ということになった。ただし、ちょうどこの境目にある 1953 年公開の米国映画について、旧著作権法が適用されるか新著作権法が適用されるか、ということが（いわゆる格安 DVD ソフトの販売差し止め訴訟を契機に）問題になった。これを「53 年問題」というわけだが、これに対する最高裁判所の判断（2007 年 12 月 18 日付）は「著作権切れ」というまことに妥当なものであった（『ウィキペディア』「1953 年問題」の項参照）。これによって、1953 年およびそれ以前に公開された米国映画（すばらしい内容のものがたくさんある）については、著作権上の心配なく字幕翻訳授業用の素材として自由に使うことができるようになった。

33) 別稿とは、染谷 (2008b) を指す。

34) 巻末資料 1 および 2 のテンプレートは次のサイトからダウンロードすることができる（***** にはテンプレートのファイル名が入る）。 http://www.someya-net.com/83-SubtitleVideo/Public/*****

【参考文献･資料※】

[01] HTMQ (2008) 「HTML クイックレファレンス」 [Online] http://www.htmq.com/style/index.shtml

[02] 稲生衣代 (2004) 「 大学教育における『映像翻訳コース』の指導手法に関する研究」日本通訳学会発行『通訳研究』第 4 号 (pp. 83-101)

[03] 稲生衣代 (2007) 「共同学習を導入した映像翻訳教育に関する考察」日本通訳学会発行『通訳研究』第 7 号 (pp. 147-165)

[04] Jmet (2005) "VobSub Subtitle Ripping, Playback and Permanent Subtitle Guide." (Article posted on the Digital Digest website) [Online] http://www.divx-digest.com/articles/article_vobsub_page1.html

[05] Microsoft Corp. (2001)「SAMI 1.0 の理解」（日本語版最終更新日 2001 年 11 月 15 日）
[Online] http://msdn.microsoft.com/ja-jp/library/cc421567.aspx

[06] ---------- (2001/2002)「デジタル メディアへのクローズドキャプションの追加」（日本語版最終更新日 2002 年 3 月 20 日）
[Online] http://www.microsoft.com/japan/msdn/windowsmedia/production/wmp7_sami.aspx

[07] ---------- (2001/2003) " Understanding SAMI 1.0" (Originally published Oct. 2001; updated February 2003)
[Online] http://msdn.microsoft.com/en-us/library/ms971327.aspx

[08] ---------- (2002) "SAMI 1.0 Demonstration Suite" (An Overview of SAMI with s sample media clip and scripts) [Online] http://download.microsoft.com/download/f/d/f/fdf0ba36-42cf-487c-ba4c-eb6f5e6e3715/SAMIOverview.exe

[09] ---------- (2004) "Advertising Solutions: Embedding the Windows Media Player in Web Pages."
[Online] http://www.microsoft.com/windows/windowsmedia/howto/articles/adsolutions2.aspx

[10] ---------- (ND/2008a) "Adding Closed Captions to Digital Media." (MSDN Library)
[Online] http://msdn.microsoft.com/en-us/library/bb248347(VS.85).aspx

[11] ---------- (ND/2008b) "Using SAMI with the Windows Media Player Control in a Browse." (MSDN Library) [Online] http://msdn.microsoft.com/en-us/library/bb249573(VS.85).aspx

[12] 染谷泰正 (2008b) 「字幕付き動画のオンライン配信について： 英語教員のための Do-It-Yourself マニュアル (2)」

[13] ---------- (2008c) 「『ハードサブ』動画の作成について： 英語教員のための Do-It-Yourself マニュアル (3)」

[14] ---------- (2008d) 「字幕付き動画作成のための SAMI ファイルの作成・編集について： 英語教員のための Do-It-Yourself マニュアル (4)」

[15] TAG index (1997-2008)「CSS スタイルシート―TAG index」
[Online] http://www.tagindex.com/stylesheet/

[16] Tetsuro (2007) 『VobSub 2.23 ガイド』 [Online] http://homepage2.nifty.com/sophia0/freesoft/VobSub/

[17] 杜甫々(1997-2002) 「とほほのスタイルシート入門」 [Online] http://www.tohoho-web.com/css/

[18] WebAIM (1999-2008) "Introduction to Captioning for Windows Media," Center for Persons with Disabilities, Utah State University, USA. [Online] http:www.webaim.org/techniques/captions/windows/

[19] W3C (1998-2003) "Synchronized Multimedia: SMIL." [Online] http://www.w3.org/AudioVideo/

[20] Wendy Boswell (2008) "Free Movie Downloads -- The Best Web Sites for Free Movie Downloads" (An article on the About.com website) [Online] http://websearch.about.com/od/searchtipoftheday/qt/freemovie.htm

※ ND [=Not Dated] は資料に初出の発行（または掲載）年月が明示されていないものを示す。URL はいずれも 2008 年 6 月 30 日現在のもの）

字幕付き動画のオンライン配信について

～ 英語教員のための Do-It-Yourself マニュアル (2) ～

染谷 泰正

（青山学院大学）

本稿は、「動画への字幕の挿入について」と題する拙文（染谷 2008a）の続編である [1)]。前稿では、大学における英語授業の一環として「字幕翻訳」を取り入れたいと考えているが、主として技術的な問題のためにその実行に踏み切れないでいる教員をおもな対象として、各種の動画にパソコン上でリアルタイムに字幕を表示させるための方法について解説した。扱った内容は SAMI 方式による字幕ファイル（SAMI ファイル）の作成と、これを動画と同期させてメディアプレーヤ上に表示させるための方法の 2 点である。後者では特に次の 3 つのケースに焦点を絞り、それぞれの具体的な手順、対応する動画メディア形式、相対的なメリットとデメリット、および使用上の注意点等について述べた。

- Windows Media Player (WMP) の字幕表示機能を使う方法
- 独自の字幕エンジンを搭載した GOM Player や ViPlay を使う方法
- 字幕表示用プラグイン DirectVobSub をインストールした上で、WMP を使って字幕を「オーバーレイ」表示させる方法

この 3 つの方法は、いずれもユーザがそれぞれ自分のパソコン上に動画ファイルと字幕ファイルを用意し、これをメディアプレーヤ上で同期表示させるというものであった。いわゆる「スタンドアロン方式」である。このほか、応用編として、ウェブページに WMP を埋め込み、字幕付きの動画をインターネット上で配信することも可能である。

前稿でも述べたとおり、これができれば、例えばクラスのグループプロジェクトとして字幕翻訳に取り組む場合、学生が各自の分担パートを相互にインターネット上で共有することができ、より効率的な作業が可能になる。また、著作権上の問題がクリアされていることを条件として [2)]、学生の作成した作品を広く一般に公開することもできる。公開を前提とした作品作りは、仮に公開範囲を学内のネットワーク上に限定したとしても、学生のモチベーションをより高めるのに貢献するものと思われる。もちろん、インターネットの利用は授業運営という点でもさまざまな利点がある。

SOMEYA, Yasumasa (2008b) “Embedding subtitled video clips in your Web pages” (Part 2 of the monograph, *Adding subtitles to video clips and embedding them in your Web pages: A Do-It-Yourself manual for teachers of English wishing to introduce subtitle translation as part of their classroom activities*. 2008)

(c) 2008 Yasumasa Someya

以上のような観点から、本稿では、字幕付き動画をインターネット上で配信するために必要なウェブページ（HTML スクリプト）の作成について、第 2 章で、いくつかの異なったケースを想定しながら、それぞれ必要最小限の技術的な解説をし、第 3 章で全体の総括をする。なお、前稿と同じく、本稿での解説はすべて Windows XP (SP2) 上での作業を前提とする。使用するブラウザは Internet Explorer (Ver. 6 以上) 、メディアプレーヤは WMP (Ver. 7 以上) とする。また、授業で使用することを前提としているため、本稿で紹介するソフトウェアは原則としてすべてフリーウェアとして無償で公開され、インターネット上で入手可能なものに限定する。

1. インターネット上で字幕付きの動画を配信するための前提条件

ウェブページに WMP を埋め込んで字幕付き動画を配信するためには、いくつかの前提条件がある。まず、いずれかのサーバに自分のスペースを持ち、ここにファイルをアップロードできる環境およびスキルが必要である。本稿では、この点はすでにクリアできているものとする。[3)] その上で、必要なのは以下の 3 点である。

1) 配信する動画ファイル
2) 上記に対応した SAMI ファイル
3) 字幕付き動画ファイルを表示させるための HTML スクリプトを記述した HTML ファイル

このうち、1) と 2) についてはすでに解決済み（染谷 2008a 参照）として、問題は 3) ということになる。ユーザは、自分のパソコンからインターネット経由でこの HTML ファイル（ウェブページ）にアクセスし、これを経由して字幕付きの動画を閲覧することになる（**図 1**）。

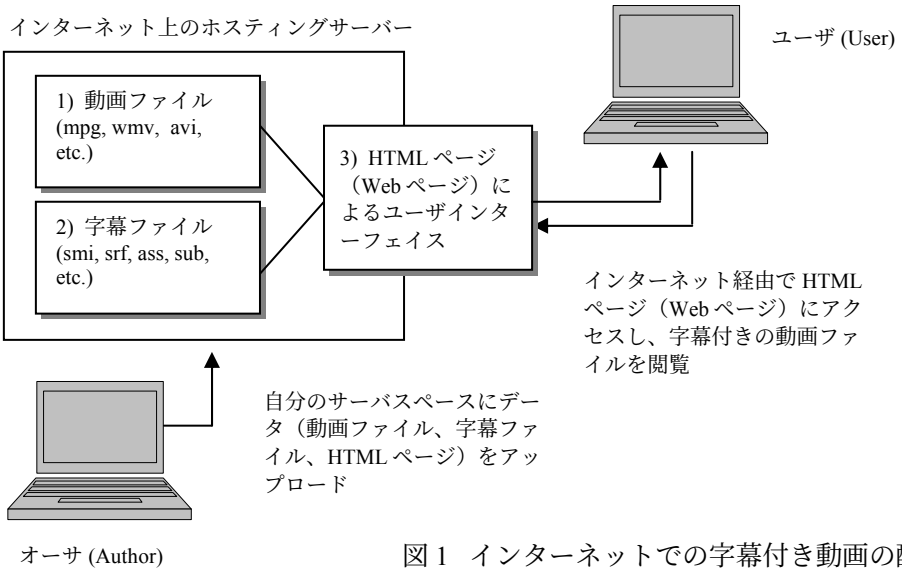


図 1 インターネットでの字幕付き動画の配信（模式図）

なお、ウェブページへの埋め込みは WMP だけの機能であり、前稿で紹介した GOM Player や ViPlay はウェブページに埋め込むことができない。[4)]

2. HTML ファイルへの WMP の埋め込み

HTML ファイルに WMP を埋め込むためには<embed> タグを使う方法と<object>タグを使う方法がある。このうち、前者は World Wide Web Consortium (W3C) による HTML または XHTML の標準的なタグとしてサポートされておらず、Internet Explorer でも <object> タグが推奨されていることから、本稿では <object> タグを使った埋め込みの方法を紹介する。[5)]

2.1 サンプルソースコード１： 単一言語クラス用ソースコード

巻末資料 1（**図 2a**, p. 48）は、<object> タグを使って WMP をウェブページ（HTML ファイル）に埋め込むためのソースコードである。この例は、字幕の言語クラスがひとつ（日本語字幕のみ）の場合の例を示したもので、複数言語（英語と日本語）対応のソースコードについては次節で解説する。なお、前述のとおり、ここではウェブページの作成に必要な基礎知識はすでにあることを前提として、おもに (1) WMP の埋め込み、(2) 動画と字幕ファイルの読み込み、 (3) プレーヤを制御するためのパラメタの指定、および (4) 字幕表示部（字幕フレーム）[6)] の設定、の 4 点に絞って解説する。

2.1.1 WMP の埋め込み (Lines 20-31)

図 2a の Lines 20-31 には <object> タグを使ってメディアプレーヤを埋め込むための標準的な例が示されている（Lines 25-27 はコメント行）。ここで重要なのは Line 21 の classid の指定で、WMP の Version 7 以降（最新の Version 11 を含む）は、すべてここに示した 16 進数による番号（0 から 9 までの数字と A から F までのアルファベットを使って数を表現する方法）で指定する。

```
20   <object id="Player" title="Windows Media Player"
21           classid="CLSID:6BF52A52-394A-11D3-B153-00C04F79FAA6"
22           width="370" height="340">
```

Line 20 の id の値にはこのオブジェクトの識別子 (identifier) として任意の名称を記入することができる。ここではとりあえず “Player” としておく。そのあとの title の値にはこのオブジェクトの任意のタイトルを記入する。ブラウザの画面上でこのオブジェクト（つまり WMP）の上にマウスのカーソルを持っていくと、ここで指定した文字列がポップアップ表示される。これはオプショナルな要素で、いわゆるアクセシビリティ機能のひとつである。

Line 22 では映像表示部の画面サイズ（映像フレーム）を指定している。width は幅、height は高さで、それぞれ任意の数値を記入することができる（単位はピクセル [7)]。370px のように単位名を添えてもよいが、何も書かなければ自動的にピクセル値として解釈される。ピクセルのほかに、%による相対指定も可）。なお、再生や停止などのコントロールボタンの表示におよそ 60 ピクセルの高さが必要になるので、画面サイズはこれを計算に入れて決める。

2.1.2 動画と字幕ファイルの読み込み (Lines 23-24)

WMP の埋め込みと画面サイズの指定のあと、「URL パラメタ」の値を設定する。URL パラ

メタとは、読み込みの対象となるメディアファイル（と必要に応じて字幕ファイル）のインターネット上のアドレス (URL = Uniform Resource Locator) を指定するためのもので、Lines 23-24 がその例である。なお、本稿のサンプル中の URL およびファイル名はすべて架空のものである。

```
23   <param name="URL" value="http://www.dummy.com/Movie/Sample1.wmv?sami=
24   http://www.dummy.com/Subtitle/Sample1.smi">
```

ここでは、URL パラメタの値として "**?sami=**" という文字記号列をはさんで前半にメディアファイルの URL、後半に字幕ファイル（SAMI 形式の字幕ファイル）の URL がそれぞれ記載されている。"*MediaFileURL*?sami=*SamiFileURL*" というシンタクスからなるこの書式を「SAMI URL 構文」と呼び、異なった場所にある SAMI ファイルをダイレクトにメディアファイルに関連付けるときにこの書式を使う。なお、この例では便宜上、分かち書きをしているが、実際のスクリプトでは途中で改行を入れず、１行（論理行）として記述する。

なお、Lines 25-27 のコメントにもあるとおり、メディアファイルと SAMI ファイルが同じフォルダ内にある場合は ?sami= から始まる SAMI ファイルの URL 指定を省略することができる（WMP は、メディアファイルが起動すると、これに対応する SAMI ファイルが同一フォルダ内にあるかどうかを自動的に検索し、あればこれを読み込むような仕様になっているため）。[8] ただし、それぞれ別のフォルダにある場合は、上記のような方法でその場所 (URL) をそれぞれ指定する。この場合、メディアファイルへのアクセス先は相対パスでもかまわないが、SAMI ファイルへのアクセス先は必ず「フルパス」（絶対パス＝インターネット経由の場合は http:// から始まる URL アドレスを、ローカルフォルダにアクセスする場合は c:\ または d\ 等のローカルドライブ名から始まるパスをすべて記入する方式）で指定しなければならない。これについては改めて後述 (pp. 7-9) する。

2.1.3 プレーヤを制御するためのパラメタの指定 (Lines 28-30)

Lines 28-30 は埋め込んだプレーヤのコントロールパラメタの指定である。autoStart はファイルの読み込み後、自動的に再生をさせるかどうかを指定するもので、値として true（自動再生する）または false（自動再生しない）のいずれかをとる。uiMode は "User Interface Mode" の略で、再生や停止などのコントロールボタンを表示するかどうかを指定する。値として取れるのは none（コントロールなし） mini（最小限のコントロール） full（すべてのコントロール）のいずれかである [9]。

Line 30 の captioningID はクローズドキャプションの表示用に割り当てられるフレームの識別子 (identifier=ID) を示す値を指定するためのパラメタで、値は任意の文字列でよい。ここではわかりやすく"captions" という名称を与えている（この ID はこの後の Line 33 で参照される）。

```
28   <param name="autoStart" value="false">
29   <param name="uiMode" value="full">
30   <param name="captioningID" value="captions">
```

このほかにも多くのパラメタが用意されているが、詳しくはマイクロソフト社の技術文書 “PARAM Elements in an OBJECT Element” ([12] Microsoft Corp., (ND/2008a) を参照されたい。ただし、ごく標準的な使い方をする限りは、ここで取り上げたパラメタ要素だけで必要十分だと思われる。

2.1.4　字幕表示部（字幕フレーム）の設定 (Lines 33-35)

<object> タグを使った WMP の埋め込みとパラメタの指定が終わったら、次は字幕表示部（字幕フレーム）の設定をする。これが Lines 33-35 である。

```
33    <div id="captions" title="Caption Window"
34         style="width:370px; height:70px; background:black;">
35    </div>
```

ここで使われている タグは、 ～ で囲まれた範囲をひとかたまのブロックとして扱い、この全体に対して何らかの設定を一括して行う場合に用いるもので、必要に応じて id 属性や class 属性、style 属性等の各種属性を割り振ることができる。ここでは、まず字幕フレームを識別するための id 属性として id= “captions” を割り当てている。Line 30 の captioningID プロパティがこの文字列を検索することで、WMP はこの 要素に指定されたフレーム内に字幕テクストを表示する。

その後の title 属性はオプション要素であるが（Line 20 参照）、通常はここに div の内容を示すタイトルを記入する。ここで指定した文字列は、この div 要素によって作成される字幕フレーム上にマウスカーソルが置かれた際にポップアップ表示される。

Line 34 では、id="captions" で指定された内容（ここでは字幕テクスト）に対して “width:370px; height:70px; background:black;” という内容のスタイルシートを適用している。字幕フレームの横幅 (width) は Line 22 で指定した映像画面の横幅に合わせる必要があるため 370 ピクセルとし、高さ (width) は 10 ポイントの文字列が最大 3 行表示できる高さとして 70 ピクセルを指定している。なお、background は字幕フレームの背景色を指し、必須要素である（省略した場合はデフォルトの透明色となり、字幕フレームが見えなくなる）。ここでは、標準色である黒 (black) を指定している。文字色やフォントサイズは SAMI ファイルで指定しているため、ここでは不要である。

基本的には以上説明した部分 (Lines 20-35) がウェブページに WMP を埋め込むために必要なスクリプトで、その前後の記述はそれぞれのユーザが必要に応じて自由に書き換えることができる。もっとも、「その前後の記述」の内容がよくわからないという場合でも、とりあえず**図 2a** のソースコード (HTML_Template1.txt) をダウンロードし、Lines 23-24 の「URL パラメタ」の値 [10] を実情に応じて書き換えた上で適当な名称で保存し（ただし拡張子は html または htm とする）、これをブラウザ (Internet Explorer) で立ち上げればよい。これで、このスクリプトの初期設定の状態で字幕付きの動画がきちんと読み込まれるはずである。参考までに、巻末資料 1（**図 2a**）の HTML スクリプトを実際に Internet Explorer で立ち上げた場合のイメージを**図 2b** に示す。

ちなみに、図 2a の Lines 38-44 には埋め込まれた WMP を使用するに当たっての注意書きが記載されている（**図 2b** の下部に表示）。文字が小さくて読みにくいと思われるので、念のために以下に再掲しておく。なお、このうちの**注 3** にもあるとおり、ユーザ側のパソコンに DirectVobSub [11] がインストールしてあり、かつ動画ファイルが mpg あるいは avi 形式の場合、そのままの状態で WMP を立ち上げると字幕が 2 重表示されてしまうことになる（wmv では 2 重表示問題は起こらない）[12]。したがって、mpg あるいは avi 形式の動画を使用する場合は、あらかじめ DirectVobSub をオフにしておくよう、画面上で明示的に指示をしておくのが望ましい。[13]

図 2b
巻末資料 1（図 2a）のソースコードに よるウェブページ画面例※

画面サイズ指定 (px)
width=370, height=340
字幕表示部サイズ指定 (px)
width=370, height=70

[動作環境]
WMP Ver. 7 and above
Internet Explorer Ver. 7.0
Windows XP (SP2)

※本ウェブページのソースコードと全画面イメージは pp. 48-49 参照。

注 1：シークバーを使って再生位置を変更･移動した場合、次の字幕タイミングから同期が始まります。うまく同期がとれない場合はいったん映像を更新 (refresh) してから再生してください。

注 2：対応するメディア形式は原則として wmv のみになります。mpg (mpeg-1) および avi でも動きますが、動作が不安定になることがあります（この場合は、いったん動画ファイルのダウンロードが終了してから再生してください）。

注 3：DirectVobSub などの字幕用プラグインがインストールしてある場合は、あらかじめそれをオフにしておいてください（mpg および avi を再生する場合）。

2.2　サンプルソースコード２： 複数言語クラス用ソースコード

巻末資料 2（**図 3a**）に掲載したスクリプトは、**図 2a** と同じく<object> 要素を使って WMP をウェブページに埋め込むためのソースコードである。ただし、この例では字幕の言語クラスがふたつ（日本語と英語）の場合を想定し、新たに「言語選択ボックス」を付け加えている。冒頭のスタイルシートの記述（<style>〜</style>）はスペースの都合上省略した。内容は資料 1 の Lines 7-12 と同じである。

以下、このスクリプトのおもな部分について解説を加える。なお、スクリプトの内容はマイクロソフト社の公式技術文書 “Adding Closed Captions to Digital Media” (Microsoft Corp., 2001) を参考にして作成した。

2.2.1　字幕ファイルと動画ファイルの読み込みと言語クラス設定情報の取得 (Lines 15-20)

スクリプト冒頭の ヘッダー部（<head>〜<.head>）では文書タイトルに続いてスタイル設定をするが、前述のとおり、この例ではスペースの都合上、スタイルシートの記述は省略した。続いて Lines 15-20 にはこのファイルの読み込み時にあらかじめ行っておく作業が指定されている。Line 15 の <script language = “JScript” . . . > という記述は、これが Microsoft JScript [14] による拡張スクリプトであることを示す。その後の for=“window” は、ブラウザでオープンされているウィンドウに対する処理を、event = “onLoad” はファイルの読み込み時に行う処理（＝イベント）であることをそれぞれ示す。Lines 16-19 がその処理の内容である。

```
15   <script language="JScript" for="window" event="onLoad">
16     Player.URL="http://www.dummy.com/Movie/Sample2.wmv?sami=http://
17   www.dummy.com/Subtitle/Sample2.smi";
18     Player.closedCaption.SAMILang=CCLang.value;
19     Player.closedCaption.captioningID="captions";
20   </script>
```

Lines 16-19 では、WMP（Ver. 7 以降）にあらかじめ組み込まれている SAMI キャプション仕様とメディアファイル検索用のオブジェクトセット（ルートオブジェクトの Player と、その下位オブジェクトのひとつである Player.closedCaption オブジェクト）を使って、まず Line 16 で Player オブジェクトの URL プロパティを使ってメディアファイルと SAMI ファイルの取得先を指定し、続いて SAMILang プロパティ (Line 18) および CaptioningID プロパティ(Line 19) という 2 つのクローズドキャプション用の専用プロパティを参照させている。前者は SAMI ファイル内の言語クラスの設定情報を取得し、後者は SAMI ファイルの内容を後述の 要素内のフレームに排出するための識別子 (id) を設定する。

なお、Lines 16-17 では前述 (p. 4) の「SAMI URL 構文」を使って動画ファイルと字幕ファイルを一括して読み込ませているが、HTML_Sample1.txt の例で見たように <object> 要素中でのパラメタ指定としてメディアファイルへのパス指定をする場合はローカルフォルダへの相対指定（例：<param name=“URL” value= “MyMovie/Sample.wmv”>）でもよく、かつメディアファイル

と SAMI ファイルが同じフォルダ内にある場合は ?sami= 以下の指定（SAMI ファイルへのパス指定）なしで SAMI ファイルが自動的に WMP に読み込まれるようになっていた [15)]。

Player オブジェクトモデルを使った場合も、ローカルフォルダに保存されているファイルへのアクセスを相対パスで指定することができる。ただし、ここでは JScript を使っているため、上記のような指定方法ではエラーになるので注意されたい。Player オブジェクトモデルでのローカルパス指定時のエラーを回避するためには、以下に述べるような方法を使う。

Player オブジェクトモデルでのローカルパス指定について

今、仮に、現在のサンプルスクリプトでの URL 指定 (Lines 16-17) を、[MyMovies] という名称のローカルフォルダ内にある Sample.wmv というメディアファイルへの「相対パス」として記述するとする。この場合、この [MyMovies] という名称のフォルダの相対的な位置に応じて、パスの指定方法が異なる。以下、ケース 1 はメディアファイル、SAMI ファイル、HTML ファイルが 3 つとも同じフォルダ (=[MyMovies]) 内にある場合を、ケース 2 はメディアファイルと SAMI ファイルが、HTML ファイルとは別のフォルダ（ここでは、ひとつ下の階層）にある場合をそれぞれ想定した場合の相対パス指定例を示したものである。

<u>ケース 1:</u> メディアファイル、SAMI ファイル、HTML ファイルが同じフォルダ内にある場合

```
├ [MyMovies]  (Current Folder)
│   ├ HTML_Sample.html
│   ├ Sample.wmv
│   └ Sample.smi
```

パス指定方法（相対パス）

```
16   Player.URL= ".\\Sample\.wmv";
```

ここで使われているドット(.) は「現在フォルダ」 (Current Folder) を表す（**注 8** 参照。なお、伝統的には「フォルダ」ではなく「ディレクトリ」と呼んでいたが、現在、Windows では前者の名称を採用しているため、本稿でも原則としてこれに従う）。「現在フォルダ」とは、現在作業をしているフォルダのことであり、ここでは当該の HTML ファイル (HTML_Sample.html) が置かれている場所を指す。円記号 (\) はフォルダの区切りを表す「区切り記号」として使用されているが、最初の円記号は「エスケープ記号」といい、次の円記号が区切り記号としてスクリプト内で有効に作用するためのものである [16)]。ファイル拡張子の前のドットにエスケープ記号が添えられているのも、このドットが別の意味に解されるのを防ぐためのものである（ただし、現在フォルダを参照する場合は始めの ".\\ " を省略することができる。したがって "Sample\.wmv " という指定でもよい）。

<u>ケース 2:</u> メディアファイルと SAMI ファイルが、HTML ファイルとは別のフォルダ（例えば、

ひとつ下の階層）にある場合

```
├ [MyFiles]  (Current Folder)
│   │  └ HTML_Sample.html
│  [MyMovies]
│   ├ Sample.wmv
│   └ Sample.smi
```

パス指定方法（相対パス）

```
16   Player.URL= ".\\MyMovies\\Sample\.wmv";
```

または

```
16   Player.URL= "MyMovies\\Sample\.wmv";
```

次のケース 3 では、HTML ファイルは [Documents and Settings] 以下の深い階層（通常の作業エリア）に置き、メディアファイルと SAMI ファイルは C:ドライブ直下に新たに作成した [MyData] というフォルダ内に置いて操作する場合を想定している。これはローカルフォルダに対する「絶対パス」による指定の例である。

<u>ケース 3:</u>　HTML ファイルは [Documents and Settings] 以下の深い階層に置き、メディアファイルと SAMI ファイルは C:ドライブ直下の [MyData] というフォルダ内に置く場合

```
[C: Drive (Root Folder)]
   |
[MyData]
   |  ├ Sample.wmv
   |  └ Sample.smi
[Documents and Settings]
   |  └ [UserName]
   |      └ [My Documents]
   |          └[MyMovies]  (Current Folder)
   |              └ HTML_Sample.html
```

パス指定方法（絶対パス）

```
16   Player.URL= "c:\\MyData\\Sample\.wmv";
```

この例のようにローカルフォルダを「絶対パス」で指定する場合、当該のフォルダが [Documents and Settings] 以下の深い階層にあると（Windows では通常、そういうことになるのだが）、必然的にパスの記述が長くなり、面倒な上にエラーも多くなるといった問題がある。また、フォルダ名が日本語になっている場合、使用環境によっては正しくパスが通らないこともある。したがって、絶対パスでの指定をする必要がある場合には、できるだけ浅い階層（で

きれば C: または D: ドライブ直下のルートフォルダ上）に新しいフォルダを作り、そこにメディアファイルと SAMI ファイルを置くようにするのがよい。

以上の 3 つの例は、いずれもメディアファイルと SAMI ファイルが同じフォルダ内に置かれていることを前提としていたが、もちろん、この 2 つをそれぞれ別々のフォルダに置いておくこともできる。その例をケース 4 に示す。ただし、「SAMI ファイルへのパス指定は絶対パス」という制約があるため、このケースでは SAMI ファイルの置き場所を C:ドライブ直下の [MyData] フォルダとし、絶対パス指定を容易にしている。

<u>ケース 4:</u>　HTML ファイルとメディアファイルは通常の「現在フォルダ」の中に置き、SAMI ファイルのみ別のフォルダ（C:ドライブ直下のフォルダ）に置く場合

```
[C: Drive (Root Folder]
   |
[MyData]
   |  └Sample.smi
[Documents and Settings]
   |  └ [UserName]
   |      └ [My Documents]
   |          └[MyMovies]  (Current Folder)
   |             ├  HTML_Sample.html
   |             └  Sample.wmv
```

パス指定方法（相対パス＋絶対パス）

```
16   Player.URL= ".\\Sample\.wmv";
17   Player.closedCaption.SAMIFileName= "c:\\MyData\\Sample\.smi";
```

ケース 4 のように、メディアファイルと SAMI ファイルを別々のフォルダに置く場合は（通常、この方法は推奨されないが）、上記の例に示したように、Player オブジェクト既定の URL パラメタに加え、SAMI ファイル指定用に別途用意されている「SAMIFileName パラメタ」を使って SAMI ファイルのファイル名 (Sample.smi) を別途指定する必要がある。

なお、上記の 2 行はすでに述べた「SAMI URL 構文」のシンタクスを使って、以下のように 1 行で書き表すことができる（このうち、? 記号は区切り記号として正しく認識される仕様になっているため、その前のエスケープ記号は省略することができる）。

```
16   Player.URL= ".\\Sample\.wmv\?sami=c:\\MyData\\Sample\.smi";
```

いずれにせよ、これらの指定は当該ファイルをサーバにあげる前にローカルでテストをする場合のパス指定方法であり、テストが終われば Lines 16-17 (p. 7) に示したような正規の URL 指定に変更しておかなければならない。

2.2.2　WMP の埋め込み とパラメタの指定 (Lines 33-38)

Lines 15-20 によって字幕（クローズドキャプション）をサポートするようにページを設定したのち、続けてウェブページに WMP を埋め込み、コントロールパラメタを設定する。この部分は HTML_Sample1.txt の Lines 20-31 と基本的には同じである。ただし、メディアファイルと字幕ファイルの読み込み、および captioningID の設定はすでに終わっているため、ここでは WMP の埋め込みと画面サイズの設定、および autoStart と uiMode という 2 つのパラメタのみを指定している。この 2 つのパラメタについてはすでに述べた (p. 4) とおりである。

```
33   <object id="Player" title="Windows Media Player"
34           classid="CLSID:6BF52A52-394A-11D3-B153-00C04F79FAA6"
35           width="370" height="340">
36   <param name="autoStart" value="false">
37   <param name="uiMode" value="full">
38   </object>
```

2.2.3　字幕表示エリアの設定と言語クラス選択ボックスの作成 (Lines 41-43)

WMP の埋め込みと必要なパラメタの指定を終えた後、字幕表示エリアの設定および、このファイル特有の言語クラス選択のためのボックスを作成する。

字幕表示エリア（字幕フレーム）の設定は 要素を使って行う（ 要素については p. 5 参照）。これが Lines 41-43 である。ここでは、Line 19 (i.e., Player.closedCaption.captioningID = “captions”) で定義された字幕内容を取得し、これを幅 370×高さ 70（単位はピクセル）のフレーム内に出力するという設定になっている。字幕フレームの背景色 (background) は標準色の黒とする [17]。なお、染谷 (2008a, pp. 2-7) で詳しく述べたとおり、字幕テクストの書式設定は原則として SAMI ファイル内で行う。

```
41   <div id="captions" title="Caption Window"
42        style="width:370; height:70; background:black;">
43   </div>
```

続いて、言語クラス選択ボックスをプルダウン形式で作成する。これが Lines 45-50 である。ここでは、前述の Player.closedCaption オブジェクトの SAMILang プロパティを使ってキャプション用言語の情報を取得し、さらに onChange イベントハンドラによって、ユーザによる選択切り替えを可能にしている。具体的には、選択が変更されるごとに “Player.closedCaption.SAMILang=CCLang.value” で定義された値（＝字幕テクスト）を新たに参照し、option で指定された言語名（ここでは “Japanese” または “English”）に一致したものを出力する、という動作をさせている。

```
45   <select id="CaptionLanguages"  language="JScript"
--           name="CCLang" class="small"
46           onChange="Player.closedCaption.SAMILang=CCLang.value">
47   <option selected value="Japanese">Japanese
48   <option value="English">English
49   </select>
50   <br><span class="small">言語選択ボックス</span>
```

なお、言語クラスの選択肢は <select> 要素内の <option> タグ内に記述するが、その内容は SAMI ファイル内で定義した名称と一致していなければならない。例えば、SAMI ファイル内で次のように言語クラスを定義してあった場合、言語選択ボックスで指定する名称は、Japanese と English ということになる（以下の下線部に対応）。[18)]

```
.JACC { name:Japanese; lang:ja; SAMIType:CC;
        font-family: Arial, Verdana,'ＭＳ ゴシック','ＭＳ Ｐゴシック';
        font-size:10pt;}
.ENCC { name:English; lang:en-US; SAMIType:CC;
        font-family: Arial, Verdana,Tahoma;
        font-size:11pt;}
```

SAMI ファイルで言語クラスが 2 つ以上定義されている場合は、Lines 47-48 を次のように拡張して、いくつでも選択肢を加えることができる。このうち、最初の “selected” という記述でマークされた選択肢がデフォルトの選択肢（ユーザによる選択がされなかった場合の初期値）になる。

```
<option selected value="Japanese">Japanese    ← デフォルトの選択肢
<option value="English">English
<option value="French">French
<option value="German">German
</select>
```

Line 50 は、言語クラス選択ボックスを配置したあと、行を変えて（ボックスの下に）「言語選択ボックス」という表示を添えるためのものである。ここで使用されている 要素は、〜 で囲んだ範囲に対して何らかの設定を行う場合に用いるもので、前述の と同じような役割をする。ただし、通例、 はブロック要素（段落や表などの大きなかたまり）に対して用いるの対し、 は（文章の一部としての）インライン要素に対して用いるという違いがある。ここでは「言語選択ボックス」という文字列に対して、スタイルシートで定義した “small” という文字クラスを適用している（資料 1, HTML_Template1.txt, Line 11 参照）。

基本的には以上説明した部分が言語クラスの選択機能を備えた WMP 埋め込みのためのスクリプトである。解説が不十分な箇所もあろうかと思われるが、とりあえずは資料 2（**図 3a**）のソースコード (HTML_Template2.txt) をダウンロードし、Lines 16-17 の URL の値（フルパス指定）を実情に応じて書き換えた上で適当な名称で保存し（ただし拡張子は.html または.htm とする）、これをブラウザ (Internet Explorer) で立ち上げれば、このスクリプトの初期設定の状態で字幕付きの動画がきちんと読み込まれるはずである。

参考までに、巻末資料 2（**図 3a**）の HTML スクリプトを実際に Internet Explorer で立ち上げた場合のイメージを**図 3b** に示す。また、**図 3a** の Line 51 以下でスペースの都合上省略されている「コメント」を、前例にならって**図 3b** の下に示す。このうち**注 2** および**注 3** は、資料 1（**図 2a**）の注意書きと同じ内容である。

図 3b
巻末資料 2（図 3a） のソースコードによるウェブページ画面例※

画面サイズ指定 (px)
width 370, height=340
字幕表示部サイズ指定 (px)
width=370, height=70

[動作環境]
WMP Ver. 7 and above
Internet Explorer Ver. 7.0
Windows XP (SP2)

※本ウェブページのソースコードと全画面イメージは pp. 50-51 参照。

注 1：言語選択ボックスの初期値は日本語になっています。英語の字幕を表示させる場合は English を選択してください。再生途中で言語選択を変更した場合、次の同期ポイントから字幕表示が変わります。うまく同期がとれない場合はいったん映像を更新 (refresh) してから再生してください。

注 2：対応するメディア形式は原則として wmv のみになります。mpg (mpeg-1, 2) および avi でも動きますが、動作が不安定になることがあります（この場合は、いったん動画ファイルのダウンロードが終了してから再生してください）。

注 3：DirectVobSub などの字幕用プラグインがインストールしてある場合は、あらかじめそれをオフにしておいてください（mpg および avi を再生する場合）。

2.2.4 字幕の文字サイズ選択ボックスの追加（サンプルソースコード 2 の改良版）

本節 (2.2) では、字幕の言語として複数の言語クラスを設定し、これを HTML 上でユーザが任意に選択できるようにするための方法についてやや詳しく解説した。複数言語を扱えるようにという配慮は、ここでは主として教育上の理由によるものであるが、これはいわゆるアクセシビリティという観点から見ても妥当な配慮であると考えられる。

アクセシビリティとは、技術、情報、公共サービスなどが、どの程度広汎な人々に利用可能であるかをあらわす概念であるが、特に、何らかのハンディを持つ人にとっての利用可能性という意味で使われることが多い。映像情報への複数言語による字幕（あるいは解説文の）付与は、いわば言語的な障壁を克服するためのアクセシビリティ技術のひとつとして位置づけることができるが、アクセシビリティという観点からすれば、字幕テクストの文字サイズについても複数の選択肢があるほうがよい。せっかくの字幕も、作成者が指定した（通常は 10〜12 ポイント程度の標準的な）フォントサイズでは小さすぎて十分に楽しむことができないユーザもいるからである。そのような観点から、ここではサンプルソースコード 2 の改良版として、言語選択ボックスのほかに、字幕の文字サイズを選択するための工夫を加えてみたい。[19)]

字幕の文字サイズを選択できるようにするためには、まず、①SAMI ファイルへの「文字クラス定義」を追加し、次に、②この文字クラス情報を取得するための特別なクローズドキャプション用プロパティ（SAMIStyle プロパティ）を HTML スクリプト上に加えた上で、前出の言語選択ボックスと同じ要領で「文字サイズ選択ボックス」を追加する必要がある。以下、この 2 点についてそれぞれ解説する。

1) SAMI ファイルへの文字クラス定義の追加

ここでは、巻末資料 2（**図 3a**）の HTML スクリプトで読み込んでいる SAMI ファイル (Sample2.smi) を使って、文字クラス定義を追加する。なお、Sample2.smi は SAMI ファイル用のテンプレート SAMI_Template2.smi（染谷 2008a, p.31）に字幕本文を追加したものである。以下に Sample2.smi（および SAMI_Template2.smi）でのスタイルおよび言語クラス定義部を引用する。

```
<STYLE TYPE="Text/css"><!--
  P {margin-top:10pt; text-align:center; line-height:125%;
     color:white; background-color:transparent;}
  .JACC {name:Japanese; lang:ja; SAMIType:CC;
         font-family: Arial,Verdana,Helvetica,'ＭＳ ゴシック',
         'ＭＳ Ｐゴシック',sans-serif; font-size:10pt;}
  .ENCC {name:English; lang:en-US; SAMIType:CC;
         font-family: Arial,Verdana,Helvetica,Tahoma;
         font-size:11pt;}
  -->
</STYLE>
```

SAMI ファイルでの「文字クラス」は、<style> 定義部の P { . . . } 要素のすぐ下の行に、先頭に # 記号を置いた上で任意の一意名称 (unique name) を与え、必要な書式設定を加えることで定義する。例えば

```
P {...}
#Emphasis {name: Big Bold; font-size: 14pt; text-align: center;
           color: blue; font-family: Arial,Verdana,Helvetica,Tahoma;
           font-weight: bold;}
```

という文字クラス定義では、Emphasis がこのクラスの定義名で、その後の { . . . } 内にその具体的な書式設定が加えられている。このうち、“name” 属性で指定された名称（これも任意の一意名称）が後述の HTML 内の “select” 要素で言及され、また、WMP を単独で起動させた場合の文字クラス参照名となる。

文字クラス定義はいくつ追加してもよいが、ここでは仮に、標準 (12 points) 、小 (10 points)、大 (15 points)、特大 (20 points) の計 4 種類の選択ができるようにしておく。なお、あまり大きなフォントサイズを指定すると既定の字幕フレームに収まりきらなくなることがあるので、フレームのサイズを考慮に入れて最大サイズを決める。

以下に、上記 4 つの文字クラスを設定した定義例を示す。このうち最初に定義した #Standard がデフォルトの選択肢となる。なお、すでに述べたとおり（染谷 2008a, pp. 10-11）言語ごとに異なったスタイル指定をする場合、共通のものだけを段落要素 P {. . . } 内で指定し、異なる要素は言語クラスごとに指定する。したがって、前掲の言語クラス設定のうち .JACC および .ENCC のフォントサイズ指定は削除し、代わりに文字クラス指定の中でそれぞれフォントサイズを指定する。

```
<STYLE TYPE="Text/css"><!--
  P {margin-top:10pt; text-align:center; line-height:125%;
     color:white; background-color:transparent;}
  #Standard {name:NormalTxt_12pt; font-size:12pt;}  ←追加（デフォルトの選択肢）
  #Small {name:SmallTxt_10pt; font-size:10pt;}      ←追加
  #Big1 {name:LargeTxt_15pt; font-size:15pt;}       ←追加
  #Big2 {name:X-LargeTxt_20pt; font-size:20pt;}     ←追加
  .JACC {name:Japanese; lang:ja; SAMIType:CC;
         font-family: Arial,Verdana,Helvetica,‘ＭＳ ゴシック’,
         ‘ＭＳ Ｐゴシック’,sans-serif;}
  .ENCC {name:English; lang:en-US; SAMIType:CC;
         font-family:Arial,Verdana,Helvetica,Tahoma;}
  -->
</STYLE>
```

SAMI ファイルに以上のような変更を加えた後、これを対応する動画ファイルと同じフォルダ内に同一名称で保存する（ただし、拡張子は.smi）。この動画ファイルを WMP で立ち上げ、メニューから「再生」→「歌詞、キャプション、および字幕」を選択すると、**図 4** に示すように SAMI ファイルで指定した 4 つの文字クラスがポップアップ表示され、文字クラスの選択が可能になる。

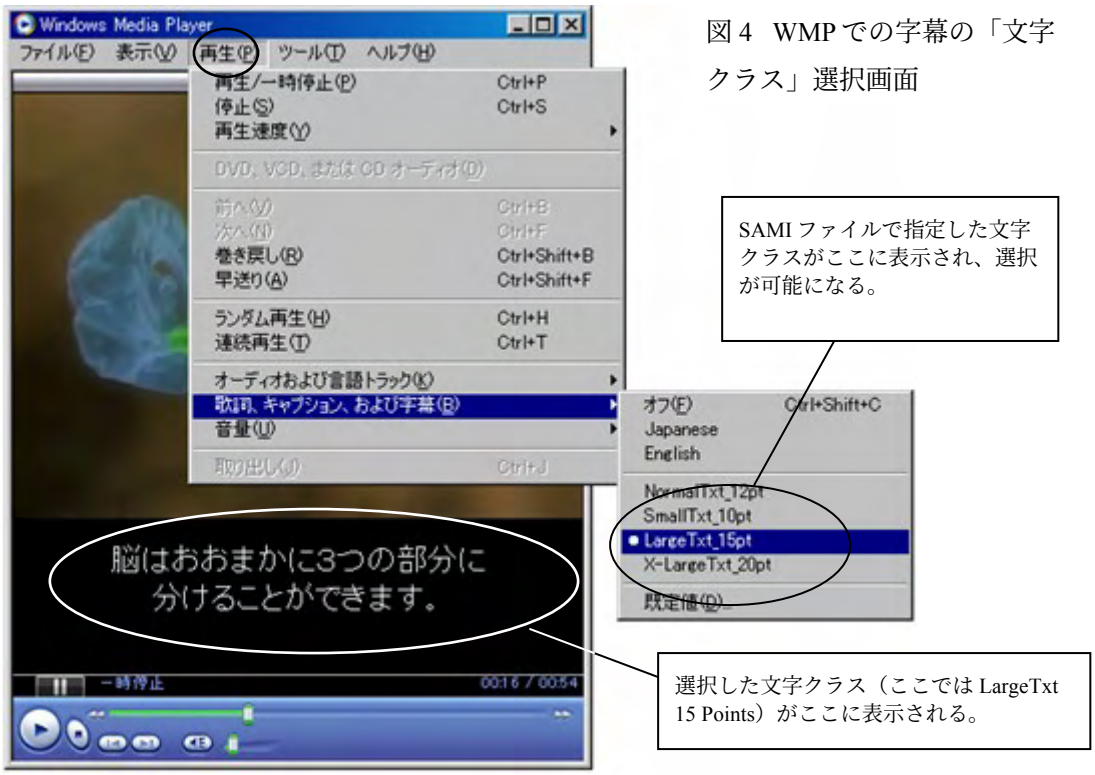


図 4 WMP での字幕の「文字クラス」選択画面

WMP をスタンドアロンで使う場合はこれだけでよいが、WMP を HTML ページに埋め込んで使う場合は、前述のとおり HTML ページの設定が必要になる。次に、これについて述べる。

2) HTML ファイルへの SAMIStyle プロパティおよび文字サイズ選択ボックスの追加

ここでは、巻末資料 2（**図 3a**）に掲載した複数言語クラス用のサンプルソースコード 2 (HTML_Template2.txt) を使って、前記の SAMI ファイルで定義した文字クラス情報を読み込み、合わせて文字サイズ選択ボックスを追加するための修正を行う。

a) `SAMIStyle` プロパティの追加

サンプルソースコード 2 で使用されている ClosedCaption オブジェクトには、SAMIStyle という専用プロパティが用意されており、これを追加することで、SAMI ファイル内で定義された文字クラスの設定情報を取得することができる。SAMIStyle プロパティは、以下に示す形式で、現在の Line 18 (Player.closedCaption.SAMILang=CCLang.value) の下にそのまま追加する(太字部

分)。なお、Lines 16-17 の “URL” 情報とファイル名（wmv ファイルと smi ファイル）はいずれも仮のものなので、これはそれぞれ実際の URL とファイル名に書き換えておく。

```
16      Player.URL="http://www.dummy.com/Movie/Sample2.wmv?sami=
17   http://www.dummy.com/Subtitle/Sample2.smi";
18      Player.closedCaption.SAMILang=CCLang.value;
        Player.closedCaption.SAMIStyle=CCStyle.value;          ←追加
19      Player.closedCaption.captioningID="captions";
```

画面サイズは初期設定のままでもよいが、大きな字幕文字サイズを扱うことを考慮して、現在の設定サイズ width=“**370**” height=“**340**” (Line 35) から、やや大きめの width=“**500**” height=“**440**” に変更するとともに、Line 42 の字幕フレームのサイズも width を **500** に、height を **90** にそれぞれ変更しておく（画面サイズは使用する動画のサイズに合わせて任意に変更可能。ただし、必ず字幕フレームと横幅サイズを揃えておくこと）。

b) 文字クラス選択ボックスの追加

次に文字クラスの選択ボックスを追加する。まず、現在の Lines 45-49 の select 要素（字幕言語の選択ボックス）の下に以下の 8 行を追加する。<option> タグに記入する値 (value) は、SAMI ファイルで定義した文字クラスの名称 (name) と同じものにする。ただし、HTML 画面上に表示するテクスト（現在は Standard Txt (12pt), Small Txt (10pt), Large Txt (15pt), X-Large Txt (20pt) という表示）は任意のものでよい。もちろん日本語表示でも問題ない。

```
48   <option value="English">English
49   </select>
     <select id ="CaptionsStyle" languag="JScript" name="CCStyle"
             class="reg" title="Choose Font Size"
             onChange="Player.closedCaption.SAMIStyle=CCStyle.value">
     <option selected value="NormalTxt_12pt">Standard Txt (10pt)
     <option value="SmallTxt_10pt">Small Txt (10pt)
     <option value="LargeTxt_15pt">Large Txt (15pt)
     <option value="LargeTxt_20pt">X-Large Txt (20pt)
     </select>
50   <br><span class="small">言語選択ボックス</span>
```

追加

なお、上記のうち <select> タグ内の class=“reg” という指定は文字クラス名を画面上で標準サイズで表示させるためのもので、title=“Choose Font Size” は、ボックスの上にマウスのカーソルを置いたときに画面上にポップアップ表示されるメッセージである。いずれもアクセ

シビリティに配慮した追加であるが、同じものを言語選択ボックスにも追加しておくのが望ましい（ただし title は “Choose Subtitle Language” に変更。ポップアップ表示は英語でも日本語でもよい）。

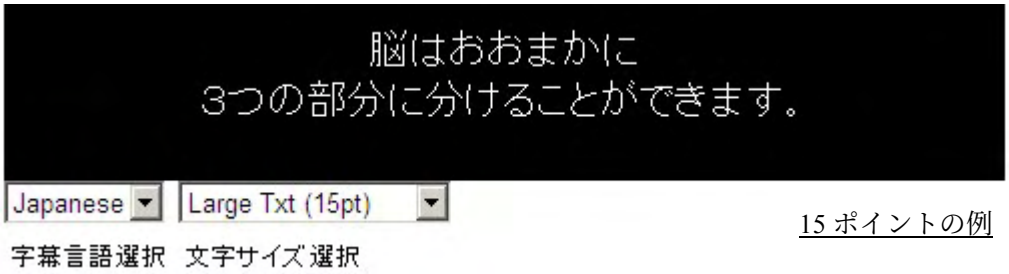
次に、上記の Line 50 を丸ごと削除して、代わりに以下の 6 行を追加する。ここでは、選択ボックスの下にテーブルを開き、2 つのセル内に指示文をそれぞれ配置しているが、セルの横幅を調整して、指示文をそれぞれのボックスの下の適切な位置に置いている。なお、テーブルの横幅は前述の新しい画面サイズの横幅（width =“**500**”）に合わせる。指示文の文字クラス（＝文字サイズ）も現在の small（8 ポイント）から reg（10 ポイント）に変更する。

追加

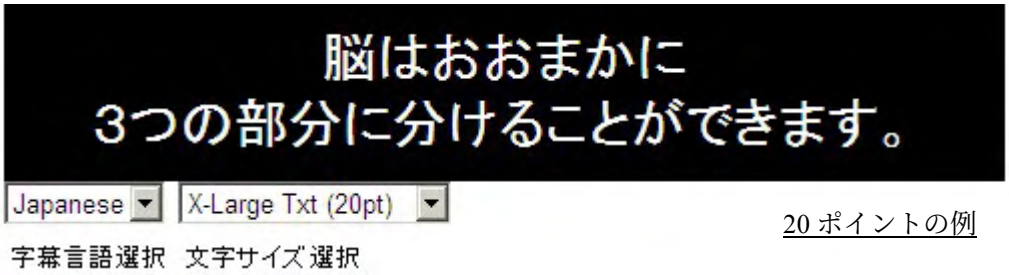
```
<table border="0" style="width:500; height:30;">
  <tr class="reg">
    <td width="85" align="left">字幕言語選択</td>
    <td width="415" align="left">文字サイズ選択</td>
  </tr>
</table>
```

以上の変更を加えた後、この HTML ファイルを保存し（拡張子は .html または .htm）、前述の Lines 16-17 の URL 情報とファイル名（wmv ファイルと smi ファイル）を実際のものに書き換えれば、すべての作業は完了である。参考までに、完成した HTML ファイルを立ち上げたときの文字サイズ選択ボックスのイメージと、選択肢として用意した 4 つのフォントサイズを比較したものを以下に示す（ウェブページ全体のイメージは**図 3b**, p.13 に同じ）。

図 5　「文字サイズ選択ボックス」のイメージおよびフォントサイズ比較※



15 ポイントの例



20 ポイントの例

※ [URL] http://www.someya-net.com/83- SubtitleVideo/Public/HTML_Template2B.html

※ フォントサイズは実寸比較ですが、画像化に伴う多少のひずみがあります。

※ SAMI ファイル (Sample2B.smi) でのスタイル設定

```
<!--
P {margin-top:10pt;
   text-align:center;
   line-height:125%;
   color:white;
   background-color:transparent;
   }
#Standard {name:NormalTxt_12pt; font-size:12pt;}
#Small {name:SmallTxt_10pt; font-size:10pt;}
#Big1 {name:LargeTxt_15pt; font-size:15pt;}
#Big2 {name:LargeTxt_20pt; font-size:20pt;}
.JACC {name:Japanese; lang:ja; SAMIType:CC;
       font-family: Arial,Verdana,Helvetica,'ＭＳ ゴシック',
       'ＭＳ Ｐゴシック', sans-serif;}
.ENCC {name:English; lang:en-US; SAMIType:CC;
       font-family: Arial,Verdana,Helvetica,Tahoma;}
-->
```

なお、本節で解説した方法のほかに、SAMI ファイルおよび HTML の中でフォントサイズを「相対指定」することで、ユーザが一定の制限内でブラウザ上で任意に文字表示サイズを変更することが可能になる。これについては改めて後述 (pp. 37-38) する。

2.3 サンプルソースコード３： プレーヤの横に大きく字幕フレームを開く

これまでに紹介したソースコードは、いずれも字幕フレーム（字幕ペイン）を映像の下に配置する例であった。しかし、場合によっては字幕フレームを映像の右側に大きく配置したいこともあるだろう。とくに長い解説文を表示させたり、映像を見ながら問題を解かせたりという応用編を考えた場合、字幕フレームの位置やサイズを自由に変えられると便利である。また、字幕フレームを映像の右側に大きく配置することができれば、例えば講義用のパワーポイントを動画化し、その画面の右側により詳細な説明を加えたものを「スライドショー」としてオンライン配信するといった応用も可能になる。

WMP をスタンドアロンで使用する場合は字幕フレームの位置やサイズは原則として固定であるが[20]、ウェブページに埋め込む場合は**図 2a** および **図 3a** のスクリプトで見たように、その仕様をユーザがかなり自由に変更することができる。本節では、これを利用して字幕フレームを映像の右側に配置し、かつフレームのサイズを映像画面と同じ大きさに設定するためのスクリプトを作成する。

2.3.1 ページ設計の基本的な考え方とタグ構造

まず、ページ設計の基本的な考え方から整理する。字幕フレームを映像の右側に配置するためには、< table> で表を作り、この表を次の**図 6** のように 2 つのセルに分割して、左側にプレーヤ、右側に字幕フレームを配置すればよい。

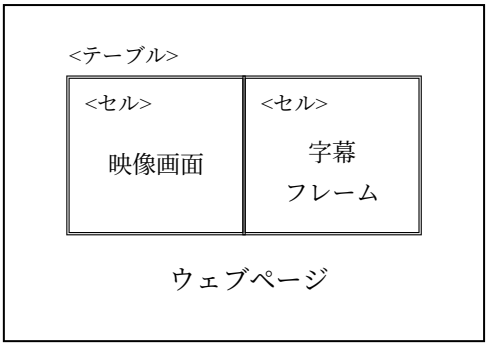


図 6　ウェブページ上の画面配置イメージ
（テーブルを 2 つのセルに分割）

HTML ではこれは次のような基本タグ構成で実現することができる。このうち <tr> は Table Row の略で、テーブルの行を記述する。<td> は Table Data の略で、テーブル内の各セルの内容を記述する。テーブルおよびセルのサイズは自由に設定できるので、当然、映像画面および字幕フレームのサイズも（ウェブページ上に表示可能な範囲で）任意のサイズを指定することができる。なお、以下の例で最初の <td> と次の <td> の間に改行タグ
 を挿入すると、セルが縦方向に並ぶことになる。

```
<table>
  <tr>
    <td>映像画面</td>
    <td>字幕フレーム</td>
```

```
    </tr>
  </table>
```

<td>～</td> のセル内には任意の文字列またはオブジェクトを置くことができる。また、セル内の文字列やオブジェクトの配置位置を、align 属性と valign 属性を使って指定することができる。前者は横位置を、後者は縦位置をそれぞれ指定し、いずれも center, left, right, top, bottom のいずれかの値をとる。例えば

```
<td align="center" valign="top">映像画面</td>
```

のように指定した場合、「映像画面」という文字列＝テクストはこのセル中において横位置＝中央寄せ (center)、縦位置＝上寄せ (top) という配置で表示されることになる。なお、どちらの属性も <td> 内でとくに明示的に指定をしなかった場合は、それぞれのデフォルト値（align=left, valign=center）が自動的に適用される。

HTML タグの詳しい解説は本稿の範囲を超えるため、ここではごく基本的な枠組みについて言及するのみに止めておくが、前記のようにテーブルを 2 分割した上で、左側のセルに WMP を埋め込み、さらに右側のセルに字幕フレームを開くための HTML ファイルの基本構造は、およそ次のようになる（**図 7**）。あとは、これにテーブルおよび各セルのサイズや枠線、背景色、<object> 要素内での WMP の埋め込みと各種パラメタの設定、さらに 2 つ目の <td> 内での タグを使った字幕フレームの設定等の作業を順次加えていけばよい。

```
<table ....>                 テーブルを開く
  <tr>                       テーブルの行を開く
    <td ....>                最初のセルを開く
      <object .....>         OBJECT タグを使ってセル内に WMP を埋め込む
        <param ....>         WMP のパラメタを指定する
        <param ....>         同上
        <param ....>         同上
      </object>              WMP の埋め込みを終了し、OBJECT タグを閉じる
    </td>                    セルを閉じる
    <td ....>                次のセルを開く
      <div ....>             DIV 要素を使ってセル内に字幕フレームを開く
      </div>                 DIV を閉じる
    </td>                    セルを閉じる
  </tr>                      行を閉じる
</table>                     テーブルを閉じる
```

図 7　テーブルを 2 分割し、左側のセルに WMP を埋め込み、右側のセルに字幕フレームを開くための HTML ファイルの基本構造

資料 3（**図 8a**）のサンプルソースコード 3 では、Line 12 でテーブルを開き、その後、Lines 15-28 で左側セル（＝映像表示画面）のスタイル設定と WMP の埋め込みを行い、さらに Lines 30-36 で右側セル（＝字幕フレーム）のスタイル設定と字幕ファイルの読み込み設定を行っている。Line 15 および Line 30 ではいずれも <td> タグ内で valign="top" という指定を加えている。これは、左右のセル内の表示（WMP と字幕フレーム）をセルのトップ位置で揃えるためである。横位置 (align) はデフォルト（左寄せ）のままでよいため、ここでは特に指定していない。

WMP の埋め込みとパラメタ設定の詳細については資料 1, Lines 20-31 の解説 (pp. 3-5) および資料 2, Lines 33-38 の解説 (p. 11) を、字幕フレームの設定に関しては資料 1, Lines 33-35 の解説 (p. 5) および資料 2, Lines 41-43 の解説 (p. 11) をそれぞれ参照していただきたい。

これらの設定を済ませた HTML スクリプトを巻末資料 3（**図 8a**）に、この HTML スクリプトを実際に Internet Explorer で立ち上げた場合の画面イメージを**図 8b** にそれぞれ示す。

なお、このサンプルではテーブルの背景色を silver とし (Line 12)、その中に組み込むセルの背景色を white とした (Line 16, Line 31)。これによって、結果的に silver のテープル枠線が残ることになる。テーブルの枠線を消したい場合には Line 12 の background 指定を white にすればよい（ただし、ページ全体の背景色も白であることを前提とする）。また、Line 31 では字幕フレームの背景色＝白、文字色＝黒という初期設定にしてあるが、この指定は自由に変更することができる。たとえば、color: white; background: black と変更すれば、真っ黒な画面に白い文字で字幕が表示されることになる（この場合は左セルの背景色も同じく黒にするほうがよい）。もちろん、これ以外にも任意の色の組み合わせが可能である。使用可能な色名およびカラーコードについては染谷 （2008a, p. 7 および**注 9**）を参照していただきたい。

図 8b
巻末資料 3（図 8a） のソースコードによるウェブページ画面例※

画面サイズ指定 (px)
height=300, width 310
字幕表示部サイズ指定 (px)
width=310, height=280

[動作環境]
WMP Ver. 7 and above
Internet Explorer Ver. 7.0
Windows XP (SP2)

※本ウェブページのソースコードと全画面イメージは pp. 52-53 参照。

なお、このサンプルで使用している字幕ファイルには、字幕テクストを「チャンク」単位で動的に表示するための工夫が加えられている。現在チャンク (current chunk) の表示は、1) 色＋太字、2) 下線、3) 背景色、の 3 つの方法で例示している。このうち、1) と 2) は HTML タグを使い、3) はスタイルシートを使う。それぞれの具体的な方法については次項で解説する。

2.3.2　字幕の表示方法に関する工夫

通常の字幕は、映像中の人物の発話またはナレーションの開始タイミングにあわせて [21] その内容を一括して表示し、発話の進行とともに順次、字幕表示を更新していく。それぞれの発話はその終了とともに直ちに消去され、いつまでも画面上に残っているということはない。画面上に表示する文字数については、日本語の場合 1 秒当たり 4〜5 文字、1 行当たり最大 13 文字（縦書きの場合は 10 文字）で 2 行まで、英語を含む欧文では 1 秒当たり 12 文字を標準とし、1 行当たり最大 32 ないし 33 文字で、同じく 2 行まで、というのが標準的なところとされている（田中 2005; Gottlieb 1998; Hatim & Mason 1997; DCMP 2007）。[22]

ただし、教材として使用する場合は、必ずしも通常の字幕表示の約束事にとらわれる必要はない。とくに、外国語によるドキュメンタリーやニュース放送などのテクストをその外国語のまま字幕表示させる場合、字幕テクストをセンテンス単位で（または複数センテンスごとに）一括表示し、音声の進行にあわせてその現在位置を「チャンク」ごとに動的に示していくスタイルのほうが、学習者の理解をより促進させるために効果的である。また、この方法を、いわゆる「フレーズ・リーディング」や「チャンク・リーディング」と呼ばれる読解スキルを養成する目的で意図的に活用することもできる。

以下、これを**図 8b** に示したようなウェブ画面上で（かつ、そのための新たな技術の導入を必要とせずに）実現する方法について述べる。いずれも、すでに学習ずみの「SAMI ファイル」（の中に記述されている字幕文字列）にいくつかの HTML タグまたはスタイルシートを追加するだけで実現可能な方法である。なお、SAMI ファイルで使用可能な HTML タグについては染谷 (2008a, **表 1**, p. 12) を参照されたい。

1)　色と太字（強調文字）で現在チャンクを動的に示す

特定の文字または文字列に色を加える場合は HTML タグ 〜 を使い、アステリスクの部分に色名または色コードを記入する。太字（強調文字）にする場合は同じく HTML タグ 〜 を使う。両者を組み合わせて使うこともできる。以下に例を示す。

[例]　`<font color=red>文字列</font>`

↓

文字列　（※ウェブ画面上では赤色で表示）

[例]　`<b>文字列</b>`

↓

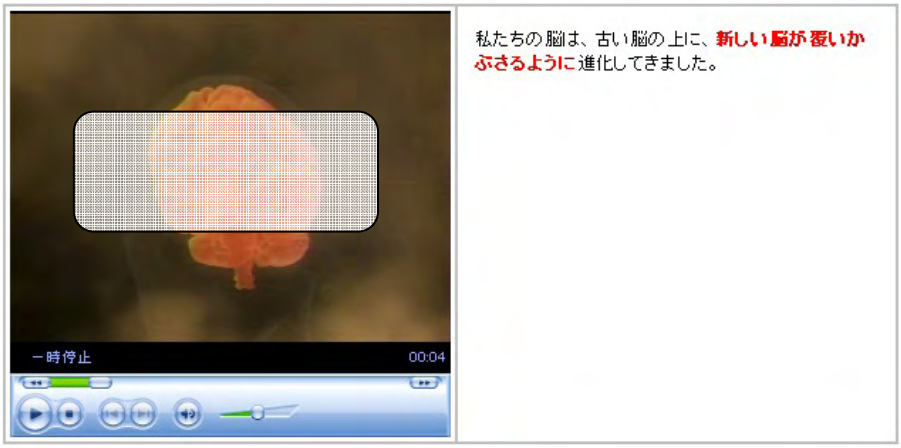
文字列

ただし、これだけでは特定の文字列に静的に変化を加えたにすぎない。これをフレーズごとに動的に表示展開していくためには、SAMI ファイルの中で以下のように指定する必要がある。例文中、太字で示した部分（HTML タグで囲まれた部分）が、時間の経過とともに順次変化していく部分である。ウェブ画面上での実際の表示例を**図 9** に示す。

SAMI ファイルでの指定例

```
<SYNC START=500><P CLASS=JACC>
  <font color="red"><b>私たちの脳は</b></font>、古い脳の上に、新しい脳が覆いかぶ
  さるように進化してきました。
<SYNC START=2000><P CLASS=JACC>
  私たちの脳は、<font color="red"><b>古い脳の上に</b></font>、新しい脳が覆いかぶ
  さるように進化してきました。
<SYNC START=3500><P CLASS=JACC>
  私たちの脳は、古い脳の上に、<font color="red"><b>新しい脳が覆いかぶさるように
  </b></font>進化してきました。
<SYNC START=5000><P CLASS=JACC>
  私たちの脳は、古い脳の上に、新しい脳が覆いかぶさるように<font color="red"><b>進
  化してきました。</b></font>
```

図 9　ウェブ画面上での表示例 1：色と太字で現在チャンクを動的に示す



※HTML ページは図 8b の例を使用。字幕フレームの背景色＝白、文字色＝黒、フォント＝ＭＳゴシック、フォントサイズ＝10 pt 、行間＝125%（以上、初期設定値）

2)　下線で現在チャンクを動的に示す

特定の文字または文字列に「下線」を加える場合は HTML タグ <u>～</u> を使う。用例およ

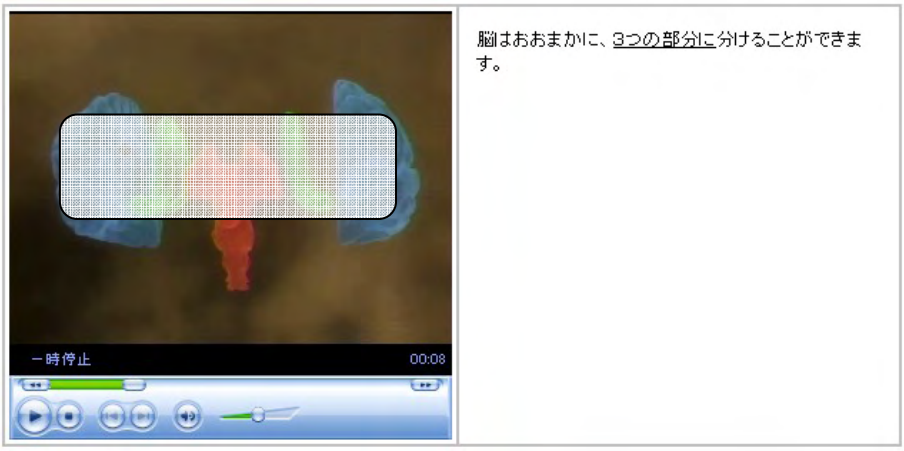
び SAMI ファイルでの指定例は以下のとおり。

[例]　<u>文字列</u>
↓
文字列

SAMI ファイルでの指定例

```
<SYNC START=7200><P CLASS=JACC>
  <u>脳はおおまかに</u>、3 つの部分に分けることができます。
<SYNC START=8000><P CLASS=JACC>
  脳はおおまかに、<u>3 つの部分に</u>分けることができます。
<SYNC START=9000><P CLASS=JACC>
  脳はおおまかに、3 つの部分に<u>分けることができます。</u>
```

図 10　ウェブ画面上での表示例 2：下線で現在チャンクを動的に示す



※HTML ページは図 8b の例を使用。字幕フレームの背景色＝白、文字色＝黒、フォント＝ＭＳゴシック、フォントサイズ＝10 pt 、行間＝125%（以上、初期設定値）

3) 文字背景色を使って現在チャンクを動的に示す

文字背景色は HTML タグでは直接操作できないため、スタイルシートを使う。前述 (p. 11) のとおり 、このケースのようなインライン要素についてスタイルシートを設定する場合は HTML の タグを使う。用例および SAMI ファイルでの指定例は以下のとおり。なお、ここでは文字色 (color) を黒、文字背景色 (background-color) をシルバーとしている。

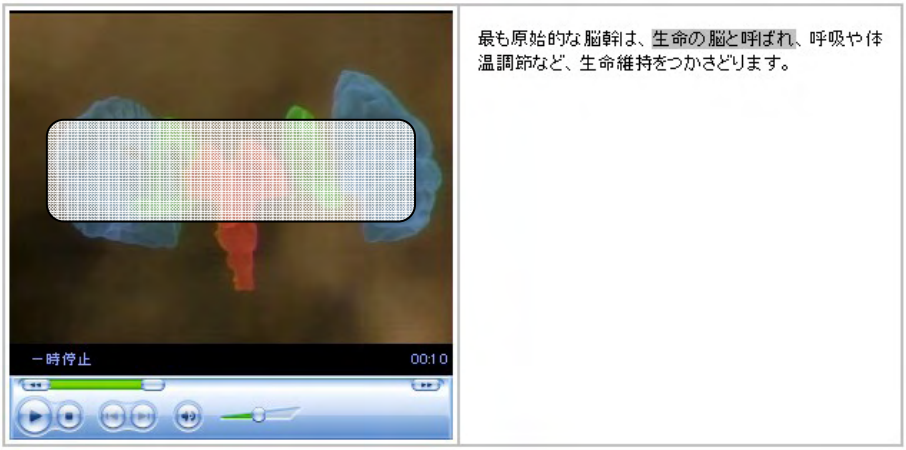
[例]　文字列
↓

文字列

SAMI ファイルでの指定例

```
<SYNC START=12000><P CLASS=JACC>
  <span style="color: black; background-color: silver">最も原始的な脳幹は
  </span>、生命の脳と呼ばれ、呼吸や体温調節など、生命維持をつかさどります。
<SYNC START=13500><P CLASS=JACC>
  最も原始的な脳幹は、<span style="color: black; background-color: silver">
  生命の脳と呼ばれ</span>、呼吸や体温調節など、生命維持をつかさどります。
<SYNC START=15000><P CLASS=JACC>
  最も原始的な脳幹は、生命の脳と呼ばれ、<span style="color: black; background-
  color: silver">呼吸や体温調節など</span>、生命維持をつかさどります。
<SYNC START=17000><P CLASS=JACC>
  最も原始的な脳幹は、生命の脳と呼ばれ、呼吸や体温調節など、<span style="color:
  black; background-color: silver">生命維持をつかさどります。</span>
```

図 11　ウェブ画面上での表示例 3：文字背景色を使って現在チャンクを動的に示す



※HTML ページは図 8b の例を使用。字幕フレームの背景色＝白、文字色＝黒、フォント＝ＭＳゴシック、フォントサイズ＝10 pt 、行間＝125%（以上、初期設定値）

以上、通常の字幕表示の応用編として、字幕テクストをセンテンス単位で一括表示し、音声の進行にあわせてその現在位置を「チャンク」ごとに動的に示していくための方法について述べた。細かな指定をしようとするとかなり手間のかかる作業になるという難点はあるが、指導の目的に応じて創造的に活用していくことのできる技術のひとつである。

なお、上述の例はいずれもセンテンスを最大表示単位としているが、実際には字幕フレームの範

囲内で任意の量のテクストを表示させることができる。テクストの行数が字幕フレームの縦幅を超える場合には、一定の範囲内で自動的にフレームが拡大されるが、読みやすさ (readability) という観点から、基本的にはテクストの行数は既定フレーム内に収めておくことが好ましい。

2.3.3　パワーポイントファイルの動画化について

本節の冒頭で、サンプルソースコード 3 に示したHTMLページの応用例のひとつとして、講義用のパワーポイントを動画化して「スライドショー」を作成するというアイデイアを提示しておいた。その具体的な方法は以下のとおりである。

1) 動画化したいパワーポイントのファイルをひらく。
2) パワーポイントの編集画面でスライドを表示させ、実際に動画上で表示させたい大きさに調整する（注： 動画再生時にサイズを拡大縮小すると画像が汚くなるので、ここできちんと表示サイズを決めておく）。
3) 各スライドを画像キャプチャーソフトでキャプチャーする（画像キャプチャーソフトはフリーのものがインターネット上でたくさん公開されているので、適当なものを入手して使用。ビスタには標準で画像クリップソフトがついてくるので、それを使ってもよい）。
4) ビデオ編集ソフト（凝った編集をするなら有料のソフトウェアがほしいところだが、そうでなければ Windows 標準の「Windows ムービーメーカ (WMM)」でもよい）で、キャプチャーしたイメージを読み込む。
5) WMM上に取り込んだ各イメージの表示時間を設定し、必要に応じて音声や各種効果を追加する。
6) 編集が終わったらビデオ用にコーディングし、適当な名称をつけて保存。
7) 解説文表示用のSAMI ファイルを作成し、拡張子を .smi とした上で上記ビデオと同じ名称で保存。
8) サンプルソースコード3 (HTML_Template3.txt) を適宜書き換え、上記ビデオの URLを追加。

基本的な手順は以上のとおりである。これだけですぐに作業にかかれるユーザも多いと思われるが、そうでない読者のために、参考までに上記の手順をより具体的に図解入りで解説したものを本稿の付録として巻末に添付しておく（→「パワーポイントファイルの動画化について」 pp. 39-47）。関心のある方はこちらを参照していただきたい。

2.4　サンプルソースコード４： 字幕を縦書き表示させる

前出のサンプルソースコード 3 では、WMP 既定の字幕表示方法（映像画面下部の字幕ペインへの表示）に代えて、プレーヤの右横に大きく字幕フレームを開く方法について解説した。プレーヤの右横に字幕フレームを開くことができれば、字幕フレームを「縦方向」にも利用することができる。つまり、日本語字幕の「縦書き表示」が可能になるということである。現在、Microsoft Internet Explorer（Ver. 5.5 以上）では “writing-mode” という CSS 属性を使用した縦書

きレイアウトがサポートされており、これを使って日本語の縦書き表示を実現することができる。

巻末資料 4（**図 12a**）に、字幕を縦書きで表示するためのサンプルソースコードを示す。HTML ページ設計の基本的な考え方はサンプルソースコード 3 と同じで、まず Line 21 でテーブルを作成し、これを 2 つのセルに分割したうえで、左側のセルにプレーヤを埋め込み (Lines 24-37)、右側のセルに字幕フレームを配置する (Lines 39-44)、というものである。このうちプレーヤの埋め込みについてはすでに見たとおりであり、ここでは解説を省略する。なお、Line 35 にある SAMILang パラメタ（以下に再掲）は、SAMI ファイルの言語（字幕言語）を指定するもので、ここでは使用言語が日本語のみに限定されていることから、これを明示的に指定したものである（言語が指定されていない場合、SAMI ファイルに定義された最初の言語クラスが既定のクローズドキャプション言語になる）。

```
35   <param name="SAMILang" value="Japanese">
```

サンプルソースコード 4 に示したスクリプトで重要なのは Lines 39-44 の部分である（Line 42 はコメント行）。ここでは右側セル（<td>〜</td>の範囲）の字幕フレームのスタイルを設定し、この中で前述の writing-mode 属性を使った縦書き表示を指定している。

```
39   <td valign="top"
40       style="writing-mode:tb-rl; background:black; color:white;
41       width:65px; height:304px;">
43   <div id="captions" title="Caption Window"></div>
44   </td>
```

まず、Line 39 では valign 属性を使ってセル内での縦方向のテクスト位置を指定し（valign 属性については p. 21 参照）、続いて style 属性を使ってテクストの表示フレーム（幅＝65px、高さ＝304px）、背景色＝黒、および文字色＝白を指定している。なお、Line 40 で使用されている writing-mode 属性は次の 2 つのいずれかの値をとる。

- `lr-tb`
- `tb-rl`

これらの値は、writing-mode 属性が適用される要素内のテクストコンテンツが表示される方法（方向）を指定するもので、`lr-tb` は「左 (left) から右 (right)、上 (top) から下 (bottom)」を表し、`tb-rl` は「上 (top) から下 (bottom)、右 (right) から左 (left)」を意味する。前者はアルファベットをベースにしたテクストを対象としたもので、通常の日本語テクストの縦書き表示は後者の値をとる。[23)]

Line 43 では 要素を使って id="captions"という ID で定義された字幕ファイル (Line 34) を参照し、これを指定のフレーム内に読み込んでいる。なお、字幕フレームのスタイル指定は Lines 39-41 の<td> タグ内で行っているため 内では不要である。

巻末資料 4（**図 12a**）に挙げた HTML スクリプトを、実際に Internet Explorer で立ち上げた場

合の画面イメージを以下に示す（**図 12b**）。

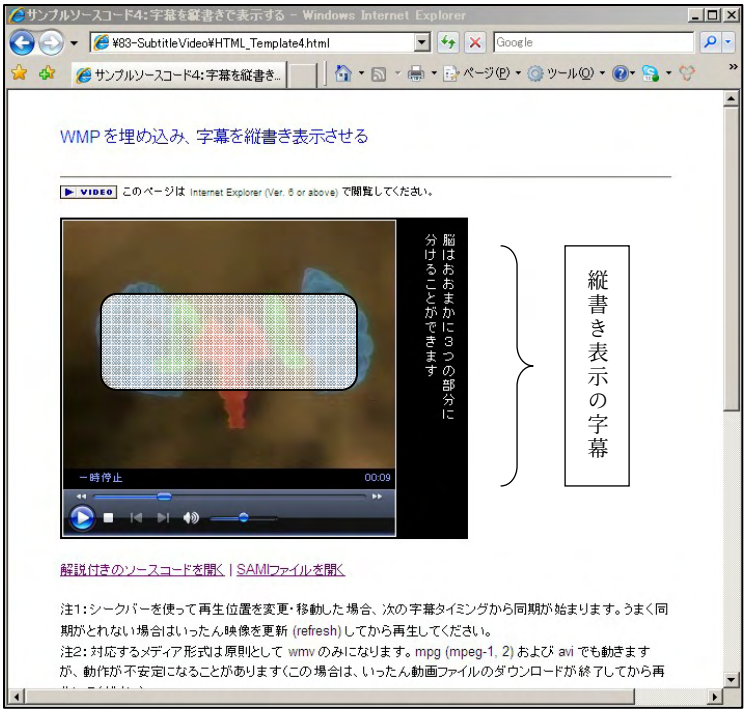


図 12b
巻末資料 4（図 12a） のソースコードによるウェブページ画面例※

画面サイズ指定 (px)
width 320, height=304
字幕表示部サイズ指定 (px)
width=65, height=304

[動作環境]
WMP Ver. 7 and above
Internet Explorer Ver. 7.0
Windows XP (SP2)

[注] フルスクリーンモード時は縦書き表示は無効になります。

※本ウェブページのソースコードと全画面イメージは pp. 54-55 参照。

なお、参考までに縦書き字幕（和文）のフォントサイズを比較したものを**図 13** に示す。いずれも 9 ポイントから 12 ポイントまでのフォント（MS　ゴシック）を、高さ＝304px、幅＝65px の字幕フレームに同一条件で表示した場合の実寸比較である（ただし、画像化に伴う多少のひずみがある）。

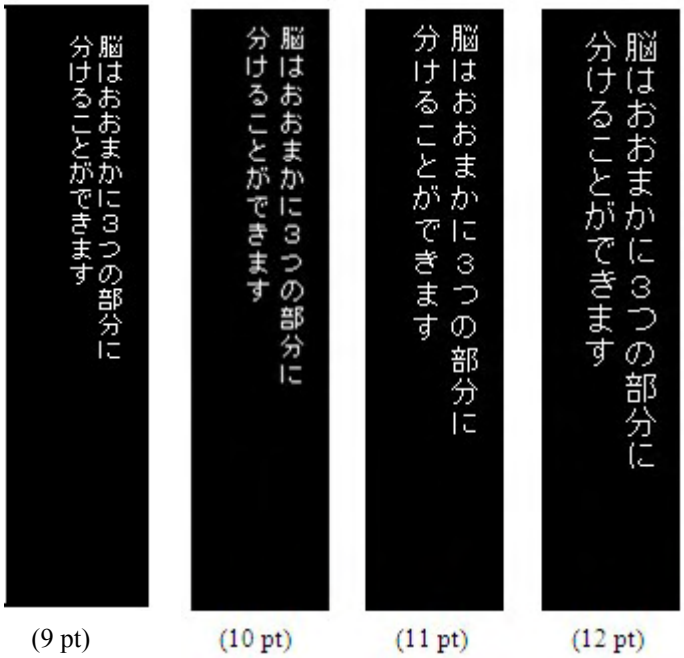


図 13
縦書き字幕（和文）のフォントサイズ比較

SAMI ファイルでのスタイル設定
margin-left: 0pt;
margin-right: 5pt;
margin-top: 10pt;
text-align: left;
font-size: *;
line-height: 125%;
font-family: 'ＭＳ ゴシック', sans-serif
letter-spacing:1px;
color: white;
background-color: transparent;

ルビの表示

日本語の場合、文字列に「ルビ（添え字）」を加えたい場合がある。たとえば、金大中を「キム･デジュン」と読ませたり、「桃源郷」という字幕に「シャングリラ」とルビを振りたい、といったケースである。

HTMLによる縦書きは、World Wide Web Consortium (W3C) で仕様を策定中のCSS3 (Cascading Style Sheets, Level 3) に含まれる「ルビ」用のタグ <ruby> を使って実現することができ、Internet Explorer および Windows Media Player でもこれを使って任意の文字列にルビを添えることができる[24)]。字幕にルビを振る場合、<ruby> タグは前述の強調文字や下線、および文字背景色の指定と同じように、「SAMIファイル」の中で特定の文字または文字列に対して使用する。また、横書き字幕および縦書き字幕のいずれにも適用することができる。以下に使用例を示す。

<u>ルビ用HTMLタグの使用例</u>

```
<ruby><rb>文字列</rb>
  <rp>（</rp><rt>もじれつ</rt><rp>）</rp>
</ruby>
↓
表示例（横書き）：  文字列（もじれつ）
表示例（縦書き）：  文字列（もじれつ）
```

なお、上例で <ruby> ～ </ruby> の間に埋め込まれる形で使われている <rb>, <rt>, <rp>という 3 つのタグはそれぞれ次のような意味である。

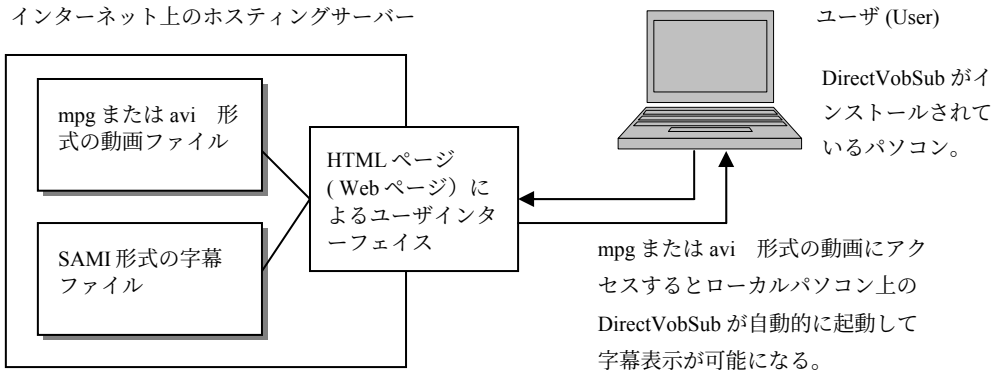
`<rb>`	ruby base	ルビをふる対象の文字列
`<rt>`	ruby text	ルビテクスト（ルビとして使う文字列）
`<rp>`	ruby parenthesis	ルビ非対応のブラウザ向けのカッコ

このうち <rp> はルビ非対応のブラウザ向けのカッコを記述するためのもので、これを加えておくと、<ruby> タグ未対応のブラウザやメディアプレーヤ（および字幕用プラグイン）では、前記の指定は「文字列（もじれつ）」のように表示されるようになる。現在のところ、<ruby> タグに対応しているブラウザは Internet Explorer (Ver. 5.5以降) のみであるため、<rp> タグによる括弧の挿入は必須である。

2.5 サンプルソースコード 5： DirectVobSub を使って字幕をオーバーレイ表示させる

これまで解説してきた方法は、いずれもウェブページに埋め込んだ Windows Media Player の下部あるいは右側に字幕フレームを開き、そのフレームの中に字幕を表示させるというものであった。ただし、字幕表示用プラグイン DirectVobSub が ローカルにインストールされていれば、インターネットで配信された mpg (mpeg-1, 2) または avi 形式の動画ファイルをウェブページに埋め込まれた WMP で再生し、その映像画面上に直接字幕をオーバーレイ表示させることができる [25)] 。ユーザ側で必要なのは、上述のとおり、自分のパソコンに DirectVobSub をインストールし、自動起動可能な状態に設定しておくことのみである。これを図式的に示すと以下のようになる（**図 1,** p. 2 を一部修正したものを**図 1b** として以下に引用）。

図 1b　mpg/avi 動画へのアクセスと DirectVobSub の起動（模式図）



オーサ（情報作成・提供者）側では、mpg または avi　形式の動画ファイルとこれに対応した SAMI 形式の字幕ファイルを用意した上で、これをユーザがウェブページ上で閲覧するための HTML ページを作成する。本節では、その HTML ページの内容と作成上の注意点等について解説する。なお、DirectVobSub については染谷（2008a, p. 2 および pp. 21-28）を参照していただきたい（以下、DirectVobSub がすでにユーザのパソコン上にインストールされ、適切に設定されているものとして話を進める）。

巻末資料 5（**図 14a**） に HTML ソースコードの例を示す。このうち映像表示にかかわる部分は <object>～</object> タグで囲まれた Lines 21-36（うち Lines 25-26 はコメント行）のみであり、以下、この部分について簡単な解説を加えておく。

```
20   <!--映像表示部-->
21   <object id="WMP"
22           classid="CLSID:6BF52A52-394A-11D3-B153-00C04F79FAA6"
23           width="640" height="544">
24   <param name="URL" value="http://www.dummy.com/Movie/Sample.mpg">
```

```
25   <!--URL 指定に関するコメント行（Line 26 まで省略）-->
27   <param name="autoStart" value="false">
28   <param name="uiMode" value="full">
29   <param name="stretchToFit" value="true">
30   <!--以下は IE 以外のブラウザ用-->
31   <embed type="application/x-mplayer2"
32          pluginspage="http://www.microsoft.com/isapi/redir.dll?prd=
33   windows&sbp=mediaplayer&ar=Media&sba=Plugin&"
34          width="640" height="544"
35          src="http://www.dummy.com/Movie/Sample.mpg"
36          showcontrols="1"
37          autostart="0"/>
38   </object>
```

<object> タグ内に埋め込まれた
<embed> 要素

ここでは、まず <object> 要素を使ってウェブページに WMP を埋め込み (Lines 21-23)、その下に、URL, autoStart, uiMode, stretchToFit の 4 つのパラメタを設定している (Line 24, Lines 27-29)。 このうち Line 29 の “stretchToFit” パラメタはこの例で初めて明示的に使用したものであるが、これを有効 (true) に設定しておくと、指定した映像表示枠のサイズに合わせて映像を表示することができる [26]。この例では、映像表示部のサイズをこれまでの例よりも大きく設定してあるが、これは字幕をより鮮明に表示するためである（字幕フォントの解像度は、画面サイズ、フォントサイズのほか、ディスプレーの解像度やパソコンの CPU 性能によっても左右される）。なお、この例では字幕データの読み込みは外部プラグイン（DirectVobSub）を使って行うため、サンプルソースコード 1〜4 に見られる “captioningID” パラメタの設定は不要である。

Line 24 の “URL” パラメタではメディアファイルの取得先を指定するが、ここでは http:// から始まる URL をフルパスで指定しなければならない（例として記述されている URL は架空のものであり、使用に当たっては実際の URL に置き換える。Line 33 も同じ）。なお、字幕ファイル（SAMI ファイル）は mpg (mpeg) または avi 形式のメディアファイルの起動とともに DirectVobSub によって自動的に読み込まれるため、ここでは字幕ファイルの URL は指定する必要がない。ただし、字幕ファイルはメディアファイルと同じファイル名（拡張子は .smi とする）を付けた上で、同一フォルダ内に保存しておかなければならない（別のフォルダから「URL SAMI 構文」や SAMIFileName パラメタを使って呼び出すことはできない ）。

なお、サンプルソースコード 1〜4 では、<object> 要素の下に 要素を使って字幕フレームを開き、ここに字幕データが読み込まれるように設定していたが、この例では字幕は外部プラグインによって映像上に直接オーバーレイ表示させるため、字幕フレームを開く必要がない。

<embed> タグによる WMP の埋め込み

サンプルソースコード 5 がこれまでの例と最も大きく異なっている点は、Lines 31-37 に見られる <embed> 要素の使用である。冒頭（p. 3 および**注 5** 参照）でも述べたとおり、これまで本稿で紹介してきた HTML サンプルではいずれも原則として <object> タグを使って WMP を埋め込んでいる。これは、主として WMP の字幕表示機能を使うための措置であり、そのためブラウザも Internet Explorer (IE) を使うことを前提としていた。ただし、DirectVobSub を使って字幕を画面上にオーバーレイ表示させる場合は基本的にブラウザの種類を問わない。例えば Safari や Firefox でも WMP 対応のプラグインがインストールされていれば IE と同じように字幕をオーバーレイ表示させることができる（Safari 3.1.2 および Firefox 3.0.1 で確認済み）。ただし、そのためには、<object> タグのほかに <embed> タグを使って WMP を埋め込んでおく必要がある。前者は IE 用、後者はその以外のブラウザ用である。

<object> タグと<embed> タグを併記する場合の注意点は、後者を前者の中に埋め込む（つまり、<object>〜</object> の間に <embed> 要素を挿入する）という点である。また、<embed> タグの場合はメディアタイプ、メディアソース、コントロール表示、自動スタートなどのプレイヤーパラメタをすべて <embed> 要素のプロパティ（属性）として記述する。上記の例では、Line 31 (type)、Line 35 (src) 、Line 36 (showcontrols)、Line 37 (autostart) がそれぞれこれに対応する。Lines 32-33 の pluginspage は <object> モデルの classid に相当するもので、WMP のプラグインの取得先をブラウザに知らせる役目をするものである。表 1 に <embed> タグの主なプロパティとその内容をまとめた。なお、<embed> のエンドタグは、開始タグの末尾に <embed . . . /> のようにスラッシュを入れることで代用することができる。

表 1　<embed> タグの主なプロパティとその内容※

プロパティ名	内容
type=	再生するファイルの MIME タイプを指定。WMP の場合は “application/x-mplayer2” となる。
pluginspage	必要なプラグインがインストールされていない場合にユーザがダウンロードできるように、プラグインの場所を指定しておく。このプロパティは <embed> タグでのみ使用される。
width=	ピクセル単位（数字）でプレーヤの幅を指定。
height=	ピクセル単位（数字）でプレーヤの高さを指定。
src=	再生するファイルの URL を指定。
showcontrols=	コントロールパネルの表示・非表示を指定。“1” または “true” で表示、 “0” または “false” で非表示。
autostart=	再生を自動的にスタートさせるかどうかを指定（autoplay= も同じ）。“1” または“true”　で自動再生、“0” または “false” で手動再生。

※参照サイト：http://www.tohoho-web.com/html/embed.htm

以上、巻末資料 5（**図 14a**）に挙げた HTML スクリプトは、字幕処理のための内部ルーチンがないため、全体としてこれまでのものと比べて格段にシンプルなものになっている。単に WMP を開いて mpg（または avi）形式の動画を読み込んでいるだけである。なお、この方法によってオーバーレイ表示される字幕の書式は原則として DirectVobSub 上での設定が適応される。ただし、SAMI ファイルの字幕本文内で HTML タグによって設定された書式はそのまま反映される。

サンプルソースコード 5 を実際に Internet Explorer で立ち上げた場合のイメージを次ページの**図 12b** に示す。また、参考までに、資料 5 の Line 39 以下にある「コメント」の全文を図の下に再掲する（資料 5 のソースコード上ではスペースの都合上、注記 2〜7 を省略）。

「ハードサブ」の作成について

サンプルソースコード 5 で示した字幕表示方法はいわゆる「ソフトサブ」方式である。染谷 (2008a, pp. 20-21) でも述べたとおり、ソフトサブでは基本的に映像データと字幕データが切り離されており、再生時に両者を動的に合成出力する。したがって、必要に応じて字幕表示のオンオフの切り換えが可能であり、字幕データの修正･更新も容易に行うことができる。また、再生時に直接描画するため、比較的鮮明な字幕テクストが得られるなどの利点がある。ただし、前述のとおり、この方式で字幕をオーバーレイ表示させるためには、あらかじめユーザ側のパソコンに DirectVobSub がインストールされている必要があり、対応するメディアファイルも mpg または avi 形式に限られているという問題がある。さらに、ユーザ側の環境によっては（とくに海外のユーザで、日本語フォントがパソコンにインストールされていない場合）字幕が文字化けすることがあるという問題もある。

このような問題を回避するためには、「ソフトサブ」方式ではなく、字幕データを画像の一部として焼き込んでしまう「ハードサブ」のほうがよいことがある。ハードサブであれば、ユーザ側のパソコンに DirectVobSub がインストールされている必要もなく、メディアファイルも任意のフォーマットで作成することができる。また、字幕の質はソフトサブに比べてやや劣るとはいえ、あらかじめ字幕データが焼き込まれているため、少なくとも文字化けの心配がないという利点もある。

従来、ハードサブの作成にはかなりの手間と専門知識を必要としたため、素人には手が出しにくい分野であったが、VirtualDub というフリーウェア（より正確には VirtualDub に TextSub というプラグイン（字幕用フィルタ）を追加し、画像と字幕データを合成してエンコーディングする方法）を使うことで、比較的簡単にハードサブを作成することができる。また、VirtualDub 以外にも、ごく簡単な手順でハードサブを作成できるソフトウエア（一部は有料）が最近になって開発･公開されており、ハードサブの作成はもはや特別な手間のかかる作業ではなくなったといってよい。ただし、十分な解説をするためにはそれなりの紙幅を必要とするため、これについては別稿（染谷 2008c）で改めて述べることとしたい。

図 14b
巻末資料 5（図 14a）のソースコードによるウェブページ画面例※

画面サイズ指定 (px)
width=640, height=544

[動作環境]
WMP Ver. 7 and above
Internet Explorer Ver. 7.0
Windows XP (SP2)

[動画ファイル形式]
MPEG-1

[外部字幕フィルター]
DirectVobSub 2.23

※本ウェブページのソースコードと全画面イメージは pp. 56-57 参照。

注 1：このファイルは、ウェブページに WMP を埋め込み、DirectVobSub を使って字幕を表示させるための方法を記述しています。使用に当たってはローカルパソコンに DirectVobSub がインストールされていることを前提とします。DirectVobSub については染谷（2008a, p.2 および pp. 21-27）を参照してください。

注 2：対応しているメディアファイルは原則として mpg (mpeg-1) および avi 形式です。

注 3：字幕ファイル（SAMI ファイル）はメディアファイルと同じフォルダ内に置いてください（「URL SAMI 構文」や SAMIFileName パラメタを使って別のフォルダから呼び出すことはできません）。

注 4：メディアファイルが読み込まれると、DirectVobSub が自動的に起動します。画面右下に Direct VobSub のアイコンが表示されるのを確認してください。このアイコンをダブルクリックすると「Direct VobSub のプロパティ」設定画面が表示されます。ここで字幕の書式設定をすることができます。このサンプルでの推奨設定は以下のとおりです。

※DirectVobSub (auto-loading version) の「プロパティ」推奨設定
[Text Settings] で Outline, Shadow, Advanced Renderer をそれぞれ選択。
[フォント設定] フォント名＝HG 丸ゴシック M-PRO；スタイル＝太字；
サイズ＝ 14 ；色＝白；文字セット＝日本語

注 5：回線の状態や CPU 性能、あるいは動画のエンコーディングの方法等の影響で、ファイルの読み込みが映像の表示に追いつかないことがあります。この場合は、いったんファイルの読み込みを終了させてから映像を表示させてください。

注 6：全画面表示で閲覧する場合（非推奨）は、ローカルの WMP の字幕機能を「オフ」にしておいてください。ウェブページに埋め込んだままの状態で閲覧する場合はローカル WMP の字幕機能はオンでもオフでもかまいません。

注 7：ローカルパソコンに QuickTime（またはその他のメディアプレーヤ）がインストールされている場合は mpg/mpeg との関連付けをはずしておいてください（QuickTime は勝手に関連付けを戻してしてしまうことがあるので、できればアンインストールしておくほうがよい）。

3. まとめ

以上、本稿では字幕付き動画をオンライン配信するための技術的な問題――とくにユーザインターフェイスとしてのウェブページ（HTML ページ）の作成方法について述べた。本稿で取り上げたのは主として次の 5 つの HTML ソースコードである。

- サンプルソースコード 1：単一言語クラス用ソースコード
- サンプルソースコード 2：複数言語クラス用ソースコード
- サンプルソースコード 3：プレーヤの横に大きく字幕フレームを開く
- サンプルソースコード 4：字幕を縦書き表示させる
- サンプルソースコード 5：DirectVobSub を使って字幕をオーバーレイ表示させる

このほか、サンプルソースコード 2 の改良版として、複数言語クラス用ソースコードに文字サイズ選択ボックスを追加したものを紹介した（**サンプルソースコード 2 B**）。なお、それぞれのソースコードに伴う字幕ファイルは、いずれもインターネット上で閲覧ないしダウンロードすることができる。

サンプルソースコード 1 は、日本語または英語による単一言語の字幕を表示させるためのソースコードで、ここでは WMP の埋め込み、動画と字幕ファイルの読み込み、プレーヤを制御するためのパラメタの指定、字幕表示部の設定等の問題について詳しく解説した。

サンプルソースコード 2では、複数言語による字幕表示を扱うための方法について扱った。例として取り上げたのは日本語と英語であるが、もちろん他の言語でも構わない。ここでは、字幕ファイルと動画ファイルの読み込み、および言語クラス設定情報の取得のために Player オブジェクトモデルを使い、Microsoft JScript による拡張スクリプトとして HTML ページに追加している。他のソースコードと比べてやや複雑になっているが、ユーザとしては所与のスクリプトをそのままコピーして使うだけでよい。

サンプルソースコード 2 の改良版 (HTML_Template2B.html) では、複数言語による字幕表示のほか、いわゆるアクセシビリティへの配慮という観点から字幕の文字サイズ選択ボックスを追加した。ここでは 12 ポイントを初期値として、10, 12, 15, 20 ポイントの 4 つのサイズの中から任意に文字サイズを選択できるようにした。選択肢の数に制限はなく、必要に応じていくつでも追加することができる。文字サイズにも明示的な制限がないが、あまり大きくすると字幕表示枠に収まらなくなるため、実質的には 20 ポイントあたりが上限と思われる。

サンプルソースコード 3 では、WMP 既定の画面下部に字幕ペインを開く方法に変わって、プレーヤの右横に大きく字幕フレームを開く方法について解説するとともに、字幕の表示方法に関するいくつかの工夫について紹介した。ここで紹介した、字幕テクストを「チャンク」ごとに動的に示していく方法は、とくにリーディング教材への応用が考えられる。また、サンプルソースコード 3 に示した HTML ページの応用例として、パワーポイントを動画化して「スライドショー」を作成する方法についても解説した（詳しい解説は巻末資料「パワーポイントファイルの動画化について」を参照）。

サンプルソースコード 4 では、 字幕を縦書き表示させるための方法について解説した。当然

のことながら、字幕を縦書き表示させるためには、字幕表示ペインを画面の横に配置し、縦に長く字幕表示部を確保する必要がある。したがって、これもサンプルソースコード 3 の応用例ということになる。

なお、日本語の字幕表示で、縦書きと横書きのどちらが読みやすいかについての調査がいくつか行われている。例えば、宮本他 (2006) では聴覚障害者 (N=5) を対象に、手話・字幕同時呈示映像に関して、縦書きと横書きのどちらが読みやすいと感じるかについて調査している。これによると、5 名中 4 名が縦書きのほうが見やすい（視線移動が楽）と答えている。一方、65 才以上の高齢者 (N=65) を対象にした香取 (2007) の調査では、被験者の 56.9% が横書き字幕の方が「好き」と回答している。「好き」というのは、「見やすい」という意味であるが、縦書きに慣れているであろう高齢者の半数をやや超える数の被験者が、横書き字幕を指示しているというのはやや意外な結果である。もちろん、字幕の「読みやすさ」は単なる慣れの問題だけではなく、文字のサイズ、文字数、文字間隔、フォント、配置、背景とのコントラスト、映像と字幕の距離、漢字の画数、あるいは提示情報そのものの内容的なわかりやすさや視聴者側の情報処理能力といった、さまざまな要素・条件がかかわってくる複雑な問題であり、このあたりは、さらなる研究が求められるところであろう。

サンプルソースコード 5 では、字幕表示用プラグインである DirectVobSub を使って字幕をオーバーレイ表示させる方法について解説した。画面上への字幕のオーバーレイ表示は WMP 単独では実現できない機能であるが、DirectVobSub を使うことで一般の映画字幕と同じような字幕表示ができるようになる。なお、いわゆる「ソフトサブ」方式の字幕表示に伴う制限を解消するための方法として、字幕データを画像の一部として焼き込んでしまう「ハードサブ」動画を作成することができる。これについては紙幅の都合上、別稿（染谷 2008c）で解説した。こちらもあわせて参照していただきたい。

「アクセシビリティ」に関する補足

最後に、「アクセシビリティ」に関する補足をしておく。前述のとおり、サンプルソースコード 2 の改良版では、アクセシビリティ (p. 14) への配慮という観点から字幕の文字サイズ選択ボックスを追加した。ユーザが必要に応じてより大きな字幕文字を選択できるようにという配慮である。ここでは、SAMI ファイル内で「文字クラス」を定義し、さらに HTML ファイル内のプレーヤオブジェクトに SAMIStyle プロパティを追加した上で、文字サイズ選択ボックスを配置するという、やや複雑な手順が必要であった。もっとも、実際にはサンプルソースコード 2B (HTML_Template2B.html) をそのままコピーして使えばよいのであって、ウェブページを作成するに当たって作成者にとくに大きな負担はかからない。しかし、実は、ウェブページ上のフォントサイズの任意変更については、ウェブページに埋め込まれた WMP 上での字幕表示を含めて、これをもっと簡単に実現する方法がある。HTML ファイルと SAMI ファイルの中で、それぞれフォントサイズを「相対指定」すればいいのである。これについては、すでに本稿 (p. 19) および染谷（2008a, p. 6）でも簡単に言及しておいたが、HTML ファイルのソースコードおよびウェブページのイメージについては本稿の巻末に添付した**資料 6** (HTML_

Template6.txt, pp. 58-60) を参照していただきたい。また、これに合わせて SAMI ファイル中でも font-size を相対指定しておく必要があるが、**資料 6** で使用した SAMI ファイル (Sample6.smi) では font-size: 1.5em とした（相対単位の em については染谷 (2008a) の**注 7** 参照)。このようにしておくことで、ユーザはブラウザのメニュー表示（「表示」→「文字のサイズ」）から、最大、大、中、小、最小と 5 段階にわたって任意の表示サイズを選択することができるようになる。この方法の欠点は、表示文字サイズの拡大によってウェブページのデザイン（レイアウト）が崩れることであるが、これはアクセシビリティを優先する上でやむをえない代償であるということになる。

なお、**資料 6** のファイルには、巻末の**注 5** で解説した Internet Explorer 以外のブラウザユーザのためのアラート用 JavaScript も含めておいた (Lines 14-21)。このほか、**資料 5** のファイルには、使用ブラウザの種類にかかわらず、ファイルの読み込み時に自動的に表示される onload アラートを <body> タグ内に追加した (Line 15)。あわせて参考にしていただきたい。

「字幕付き動画のオンライン配信について」（染谷 2008b）付録

パワーポイントスライドの動画化について

本稿の第 2 章第 3.3 節 (p. 27) で、パワーポイントスライドの動画化について簡単な説明を加えておいたが、以下、その手順をより具体的に図解入りで解説する。

用意するもの

- 動画化したいパワーポイントのファイル
- 画像キャプチャーソフ 注 1)
- ビデオ編集ソフト注 2)
- マイクロホン注 3)
- HTML ファイル注 4)
- SAMI ファイル 注 5)

注 1)　画像キャプチャーソフトはフリーのものがインターネット上でたくさん公開されているので、適当なものを入手する。お薦めは「mini-カッター (mCut.exe)」。以下の解説ではこれを使用。
[入手先] http://www.portnet.ne.jp/~suga16/soft/mcutter/mcutter.htm
なお、Windows Vista には標準で画像クリップソフトがついてくるので、それを使ってもよい。

注2)　本稿ではWindows 標準の「Windows ムービーメーカ」を使用。
[XP] http://www.microsoft.com/japan/WindowsXP/using/moviemaker/default.mspx
[Vista] http://windowshelp.microsoft.com/Windows/ja-JP/help/ec3fff68-e53c-4168-ae74-8557325e57e21041.mspx

注3)　ただし、「ナレーション」を録音追加する場合のみ。マイクロホンはあらかじめパソコンに接続して、使用準備をしておく（必要に応じてナレーション用の原稿も用意）。

注4)　染谷 (2008b) 資料3（**図6a**）サンプルソースコード 3 <HTML_Template3.txt>）

注 5)　染谷 (2008a) 資料 2 SAMI 字幕ファイルテンプレート 2（日英 2 か国語用）
<SAMI_Template2.smi>

作業手順（※手順を示す番号は、本稿の第 2 章第 3.3 節で示したものとは異なります）

1. **パワーポイントの起動**：まず、動画化したいパワーポイントのファイルをひらく。
2. **スライドの選択**：パワーポイントの編集画面で、切り取りたいスライドを表示させ、編集画面いっぱいに表示させる（パワーポイントの本文は20ポイント以上の大きさを推奨）。
3. **スライドのキャプチャー**：次に、画像キャプチャーソフト＝ミニカッター (mCut.exe) を起動させ、スライドをキャプチャーする（→**図1** 参照）。[27)]
4. 以上の要領で、必要なスライドをすべてキャプチャーして保存。

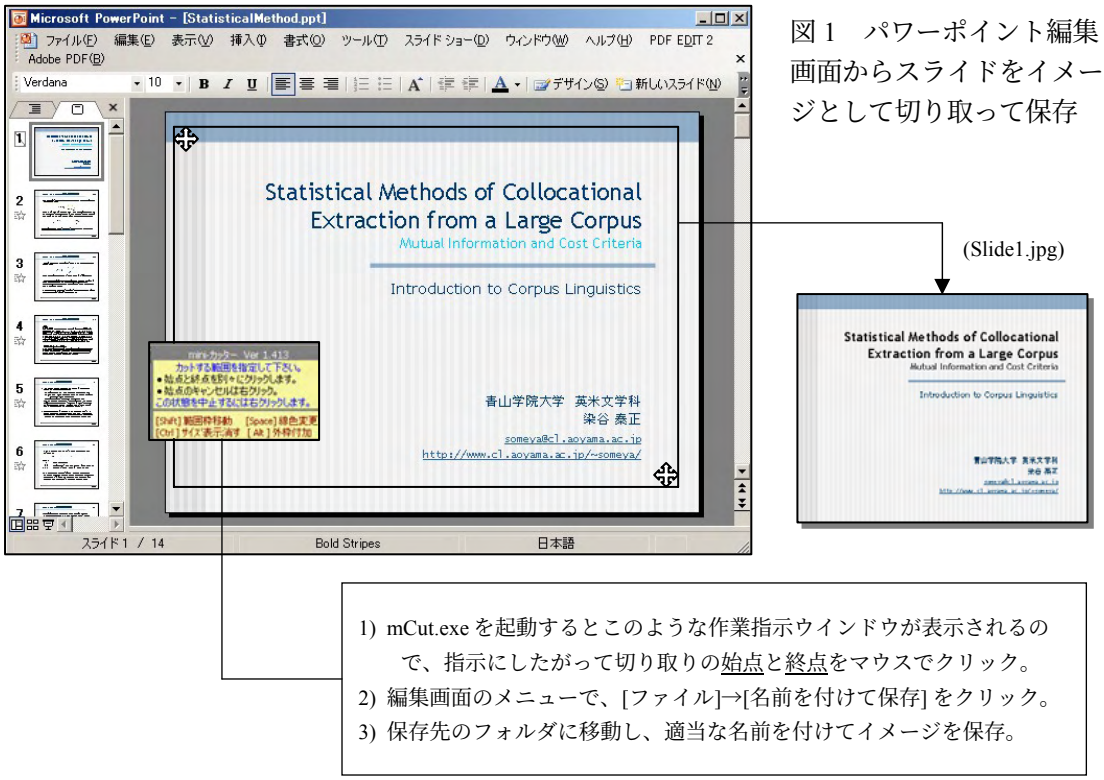
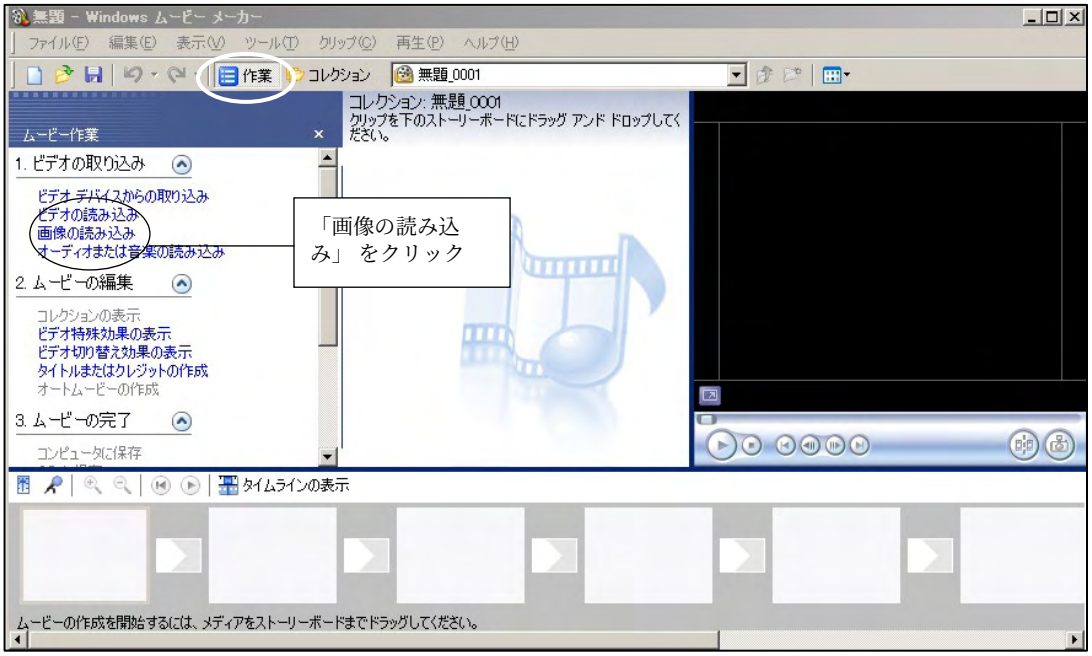


図 1　パワーポイント編集画面からスライドをイメージとして切り取って保存

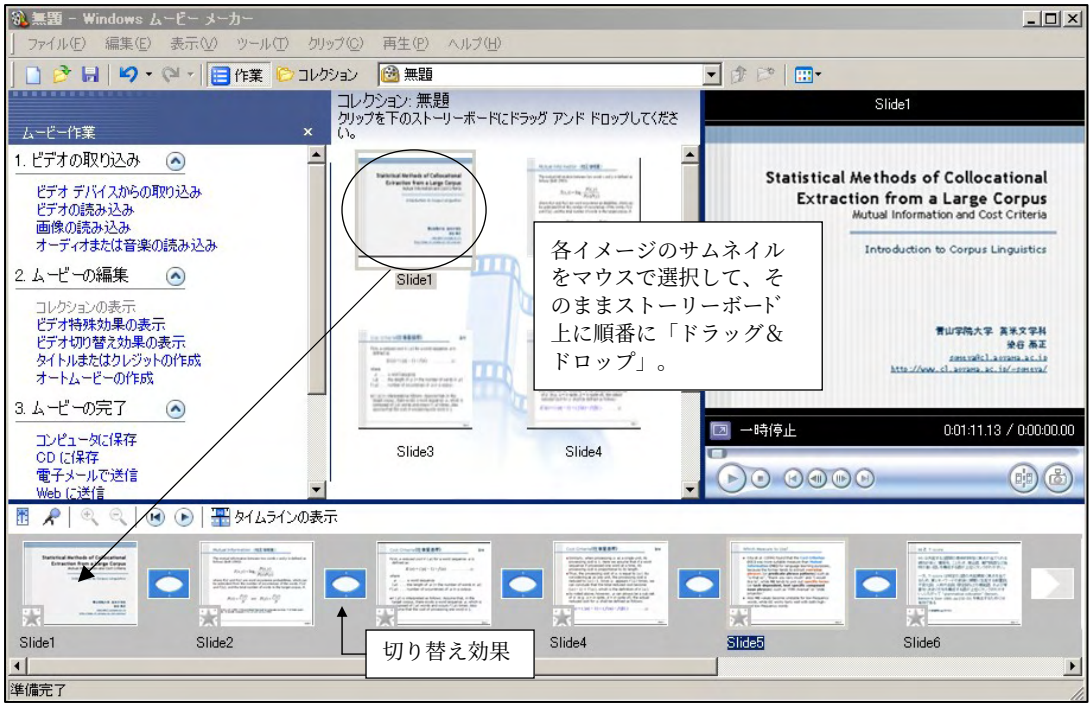
5. **Windows ムービーメーカの起動**：スライドのキャプチャーを終了したら、次に Windows の [スタート] → [プログラム] から「Windows ムービーメーカ」(WMM) を選択して起動。
6. **WMMへのキャプチャーイメージの読み込み**：WMM 初期画面（**図2**）の左側に「ムービー作業」のメニューが表示されているので（表示されていない場合はメニューバーの [作業] アイコンをクリックして表示させる）、ここで「1. ビデオの取り込み」メニューの中の「画像の読み込み」をクリック。

図 2　WMM 初期画面



7. 「読み込みファイル」の選択画面が表示されるので、先ほどキャプチャーしたイメージが保存されているフォルダに移動。イメージを選択し、[読み込み] ボタンをクリック。
8. 読み込んだイメージが**図2** の画面に追加されたことを確認したら、同様の手順で次のイメージを読み込む。これを必要なだけ繰り返す。
9. すべてのイメージの読み込みが終わったら、次に、各イメージを順番に画面下の「ストーリーボード」に追加（**図3**）。

図 3　イメージをストーリーボードに追加



10. **特殊効果やタイトルの挿入**：次に、「ムービー作業」メニューの「2. ムービーの編集」から、「特殊効果の表示」「切り替え効果の表示」「タイトルまたはクレジットの作成」を必要に応じて選択し、それぞれ画面上の表示される指示に従って作業を行う（ただし、この作業はオプション。省略してもかまわない）。
11. **タイムラインの設定**：ストーリーボードの上部に表示されている「タイムラインの表示」をクリック。ストーリーボード表示が「タイムライン」表示に切り替わる（**図4**）。ここで、タイムライン上のスライドイメージの右端にマウスを合わせ、クリックしてからドラッグすることで、各スライドの再生時間を任意に設定することができる。後で変更することができるので、最初はとりあえずおおまかに各スライドごとの再生時間を設定しておく。
12. **音声（ナレーション）の追加**：タイムラインの上部に表示されているマイクのアイコンをクリック。画面が**図5** に示したように切り替わるので、ここで必要に応じてナレーションを追加する（ナレーションの追加はオプション）。ナレーション追加の手順は以下のとおり。
    1) ナレーションを加えたいスライドを選択。
    2) 「ナレーションの開始」をクリックして録音開始（あらかじめマイクロホンを接続）。

図 4　タイムラインの設定

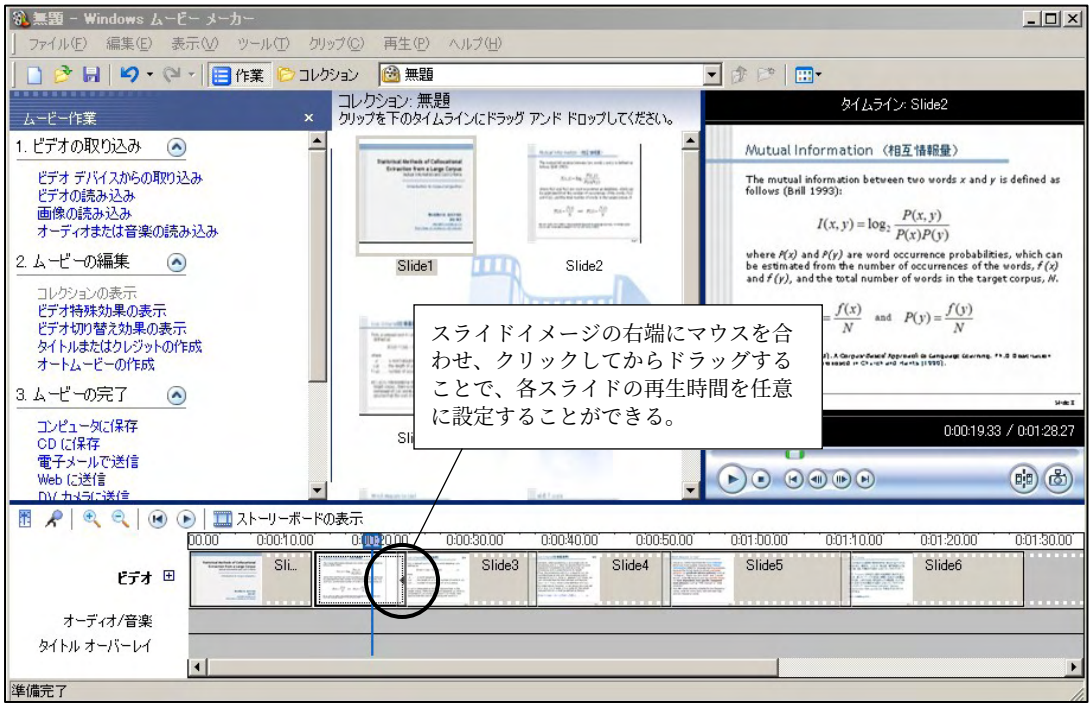
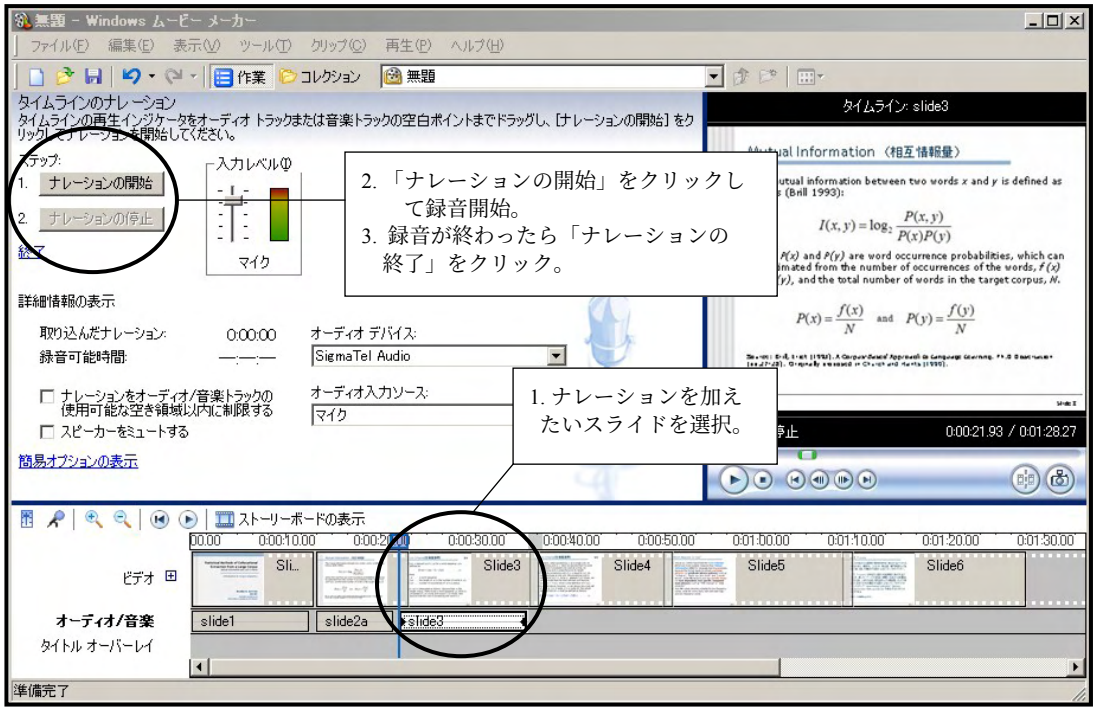


図 5　音声（ナレーション）の追加



3)　録音が終わったら「ナレーションの終了」をクリック。

4)　「メディアファイルの保存」画面が表示されるので、適当な保存場所を選択して保存（デフォルトの保存場所は「マイドキュメント」→My Video→「ナレーション」フォルダ）。

5) 保存が終了すると、「オーディオ／音楽トラック」上に、各スライドに対応する音声ファイルの長さを示すインジケータが追加される（再録音する場合は、インジケータ上にマウスを置いて右クリック→削除を選択し、既存ファイルを削除したのち上記 1～5 を繰り返す）。
6) すべてのスライド（あるいは必要なスライド）に対してナレーション追加作業を終えたら、画面上部メニューバーの [作業] ボタンをクリック。

13. **完成したムービーの保存**：「ムービー作業」メニュー（**図4** 参照）から、「3. ムービーの保存」→「コンピュータに保存」を選択。
14. 「ムービーの保存ウイザード」（**図6**）が表示されるので、ここでビデオに適当な名称をつけた上で、保存場所を指定し、「次へ」をクリック。

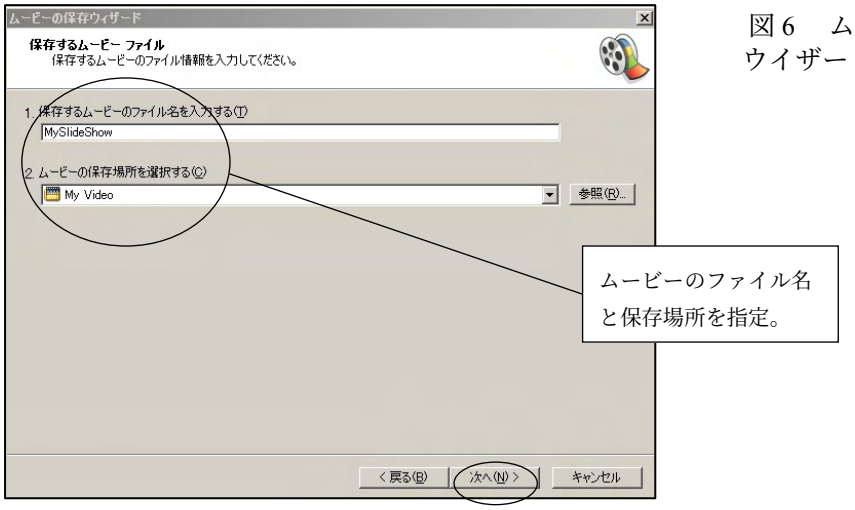


図 6　ムービーの保存ウイザード画面

15. 「ムービーの設定ウイザード」（**図7**）が表示されるので、ここで「コンピュータの最高の品質で再生」にチェックを入れて、「次へ」をクリック。

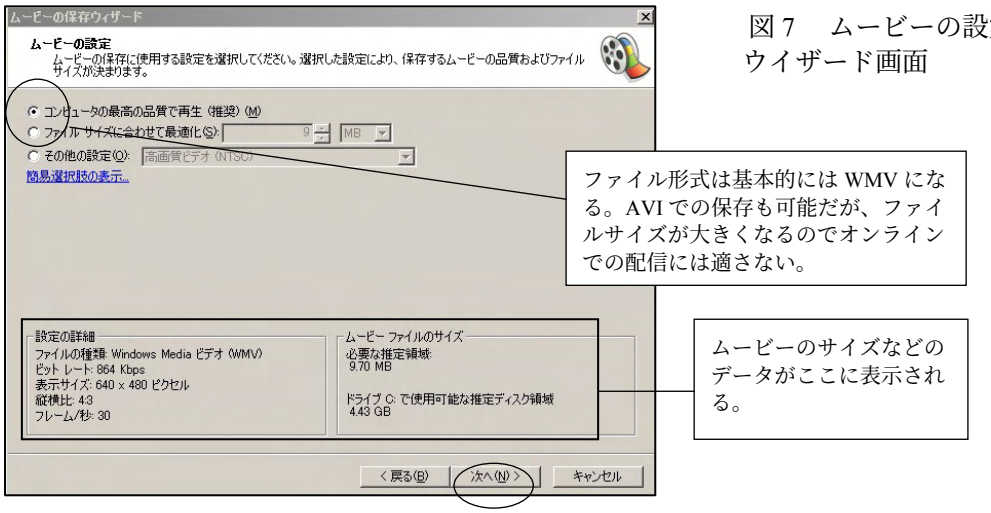


図 7　ムービーの設定ウイザード画面

16. 選択された形式でのムービーの保存が開始される、保存が終わると終了のメッセージが表示される。ここで、「完了をクリックしたときに、ムービーを再生する」を選択すると、保存後にムービーが再生される。

パワーポイントファイルの動画化は以上で終了。次に、「SAMI ファイル」と「HTMLファイル」を作成する。

17. **SAMI ファイルの作成**：染谷 (2008a) の資料2「SAMI 字幕ファイルテンプレート2（日英 2 か国語用）」<SAMI_Template2.smi> を読み込んで、内容を適宜書き換える。

1) <u>スタイルの設定</u>（SAMIファイルについて、詳しくは染谷 (2008a, pp. 2-15) 参照）

字幕のスタイルは必要に応じて任意に設定すればよいが、ここでは仮に以下のような設定にしておく。フォントの初期設定は日本語が ‘MS Gothic’（または sans-serif）、英文が Arial（またはVerdana ないし Helvetica）で、フォントサイズはいずれも 10 ポイントとした。テクストは左寄せ、行間は 125% という設定である。

```
<STYLE TYPE="text/css"><!--
P {margin-left:8pt; margin-right:8pt;  margin-top:10pt;
   font-size:10pt; line-height:125%; text-align:left;
   font-family: Arial,Verdana,Helvetica,‘MS Gothic’,sans-serif;}
.JACC {Name:Japanese; Lang:ja; SAMIType:CC;}
-->
</STYLE>
```

2) <u>スライドに加える解説文の追加</u>

各スライドに加える解説文は、字幕フレームのサイズ（HTMLファイルで設定）に応じてその量を調整する必要があるが、基本的にはその範囲内で自由に記述することができる。上記の CSSスタイルシートで設定した基本スタイル以外のスタイルを適用する場合には、それぞれのブロックごとに （または ）要素を使ってスタイル設定を適用する。

以下のサンプルでは、最初のスライド (Slide 1) にプレゼンテーションのタイトル画面を、ムービーの開始後 100ms から10000ms の間、約10 秒間表示し、その後、2 枚目のスライド (Slide 2) を 10000ms から 21000ms の間、11秒間表示する、という設定とした（Slide 3 以降の設定は省略）。Slide 1 では、タイトルページ用にフォント、文字サイズ、テクスト表示位置、行間などをインラインで別途指定している。なお、ここでは <img> タグを使って大学のロゴイメージを挿入している。SAMIファイルで使用可能な HTML タグ一覧の中に <img> タグは含まれていないが、WMP をウェブページに埋め込んで使う場合には、<img> タグを使って静止画像を埋め込むことができる（WMPを単独で立ち上げる場合は不可）。

[スライドに加える解説文のサンプル]

開始行
```
<SYNC start=0><P class=JACC> 
```

Slide 1
```
<SYNC start=100><P class=JACC>
<center><img src="images/aoyama-logo.jpg" height=41 width=126></center>
<br>
<div style="font-family:'MS Gothic',sans-serif; font-size:12pt; font-weight:bold; text-align:center; line-height:150%;">
大規模コーパスからのコロケーション抽出<br>
のための統計学的手法について</div>
<br>
<div style="font-family: Arial,Verdana,Helvetica,'MS Gothic',sans-serif; font-size:10pt; text-align:center; line-height:150%;">
～ 相互情報量 (MI) と仕事量基準 (CC) ～ <br>
<br>
染谷泰正<br>
青山学院大学</div>
```

Slide 2
```
<SYNC start=10000><P class=JACC>
[要約] 相互情報量 (Mutual Information) とは、2つの単語の共起関係（相互の結びつき＝コロケーション）の強さを表す指標のひとつ。具体的には、ある単語ともうひとつの単語とが共起する確率と、それぞれが個別に生起する確率との比である。実際の計算では、正規化するためにコーパス全体の総単語数をかけ、その対数（底は2）をとる。
<br><br>
<span style="font-size:8pt;">Note: This simple method can only extract collocations of length two; however, Jelinek (1990) proposed a generalization of this method to include collocations of arbitrary length. See Jelinek, F. (1990), "Self-organized language modeling for speech recognition." In Waibel, A., and Lee, K.F. (Eds.), <i>Readings in Speech Recognition</i>, pp. 450-506. Morgan Kaufmann Publishers.
</span>
```

Slide 3
```
<SYNC start=21000><P class=JACC>
次のスライドに加える解説文
```

Slide *n*
```
<SYNC start=*****><P class=JACC>
n 枚目のスライドに加える解説文（***** には実測時間が入る。以下同）
```

最終行
```
<SYNC start=*****><P class=JACC> 
</BODY>
```

Slide 2 では英文部分を注として表示するために、初期設定値より小さい 8 ポイントに再設定した（フォントは変更なし）。なお、Slide 2 ではブロックの一部についてのみのインライン（＝行内）指定になるため、 要素を使ってスタイル設定をしている（ と の使い分けについては染谷 (2008b, p. 12) 参照）。

18. **SAMIファイルの保存**：すべてのスライドについて解説文の追加が終わったら、このファイルを前述のビデオと同じ名前（拡張子は.smi）で保存する。保存先は原則としてビデオの保存先と同じとする（異なるフォルダ内に保存する場合の注意点は次項の注記参照）。
19. **HTMLファイルの作成**：次に、ビデオとSAMIファイルを同期させてウェブページ上に表示させるための「HTMLファイル」を作成する。そのために、染谷 (2008b) 資料3 の「サンプルソースコード 3」 <HTML_ Template3.txt> を読み込み、内容を適宜書き換える。ただし、現在の設定のまま使用する場合は、このうち Line 22-23（以下に再掲）にある URL パラメタ行のみを実際の URL に書き換えるだけでよい（イタリックの部分を書き換え。サーバにファイルをあげる前にローカルにテストする場合は、ローカルフォルダを指定する）。なお、サンプルソースコード3 についての詳しい解説は染谷 (2008b, pp. 19-26) を参照。

```
22   <param name="URL" value="http://www.dummy.com/Movie/Sample3.wmv?sami=
23   http://www.dummy.com/Subtitle/Sample3.smi"> <!-- このURLはダミーです。-->
```

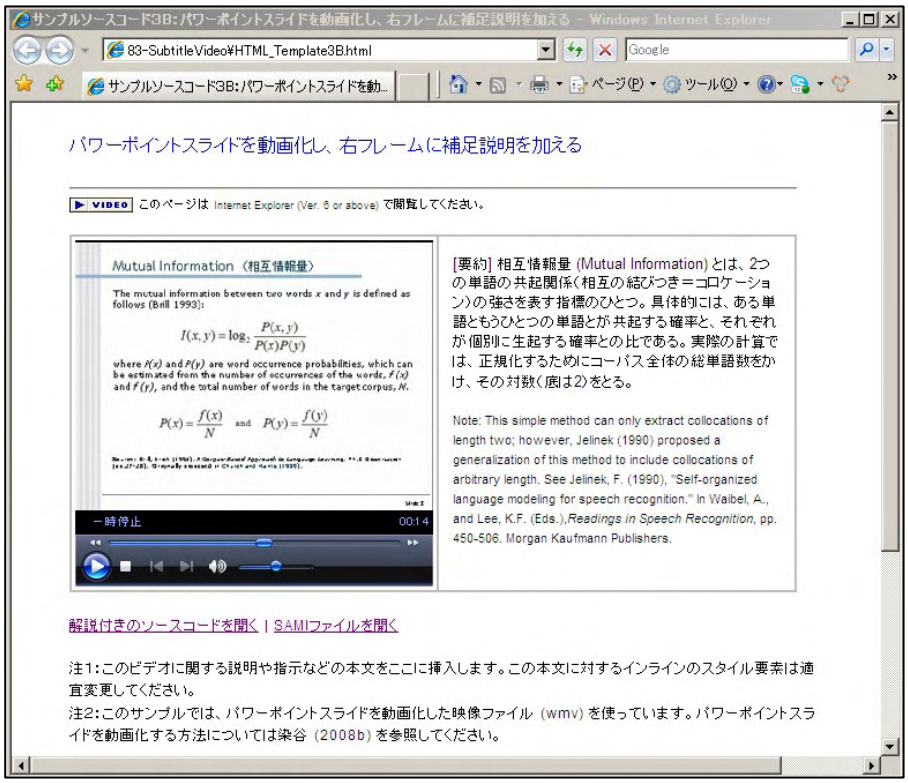
注） ビデオと SAMI ファイルが同じフォルダ内に保存されている場合はビデオの URL のみを指定。それぞれ異なるフォルダに置いてある場合は、上記のように ?sami= という記述をはさんで、そのあとに SAMI ファイルの URL をフルパスで指定する。

ビデオの表示サイズや字幕フレームの表示サイズを変更したい場合は、Line 12 (テーブルのサイズ)、Line 16 (左側セル＝映像用フレームのサイズ)、 Line 20 (Windows Media Player の画面サイズ)、および Line 31 (右側セル＝字幕フレームのサイズ) の縦横サイズの数値をそれぞれ適宜変更する。次ページの**図 8** に掲載した HTML_Template3B.html の例では、以下のようなサイズとした（太字で表示されている部分が変更箇所）。

```
12   <table style="height:370; width:820; background:silver;">
16    style="height:370; width:410; background:white;">
20            height="370" width="410"
31    style="height:370; width:410; color:black; background:white;">
```

20. **HTMLファイルの保存**：以上、必要な編集作業が終わったらこのファイルに適当な名前を付けて保存。なお、サンプルソースコード 3 は拡張子が txt になっているので、これを html（または htm）に変更して保存すること。**図8** に、この HTML ファイルを実際に立ち上げた画面例を示す（左右フレームのサイズは初期値のまま）。

Slide 1



Slide 2

図 8 パワーポイントから作成したビデオと SAMI ファイルを埋め込んだウェブページの表示画面例

※[URL] http://www.someya-net.com/83-SubtitleVideo/Public/HTML_Template3B.html

資料 1（図 2a） サンプルソースコード１（単一言語クラス用）<HTML_Template1.txt>

```
01  <html>
02  <head>
03  <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
04  <title>サンプルソースコード１: WMP を HTML ファイルに埋め込み、字幕表示させる（単一言語クラス用）
05  </title>
06    <style type="text/css"><!--
07     .big {line-height:150%; font-size:12pt; font-family:Arial,Verdana,
08      Helvetica,'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:blue;}
09     .reg {line-height:150%; font-size:10pt; font-family:Arial,Verdana,
10      Helvetica,'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:black;}
11     .small {line-height:150%; font-size:7pt; font-family:Arial,Verdana,
12      Helvetica,'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:black;} -->
13    </style>
14  </head>
15  <body text="#000000" bgcolor="#FFFFFF">
16  <blockquote>
17  <p class=”big”>WMP を HTML ファイルに埋め込み、字幕表示させる（単一言語クラス用）</p>
18  <hr width="100%">
19  <!--映像表示部：映像表示サイズの指定、およびメディアファイルと字幕ファイルの場所の指定-->
20  <object id="WMP" title="Windows Media Player"
21          classid="CLSID:6BF52A52-394A-11D3-B153-00C04F79FAA6"
22          width="370" height="340">
23  <param name="URL" value="http://www.dummy.com/Movie/Sample1.wmv?sami=
24  http://www.dummy.com/Subtitle/Sample1.smi">
25  <!--この URL はダミーです。実際の URL と入れ替えてください。SAMI ファイルが動画と同じフォルダ
26  内にある場合は ?sami= 以下の指定なしで SAMI ファイルが自動的に読み込まれます。別のフォルダにあ
27  る場合は上記のようにその場所 (URL) をフルパスで指定してください。-->
28  <param name="autoStart" value="false">
29  <param name="uiMode" value="full">
30  <param name="captioningID" value="captions">
31  </object>
32  <!--字幕表示部：表示領域のサイズと画面背景色の指定-->
33  <div id="captions" title="Caption Window"
34       style="width:370px; height:70px; background:black;">
35  </div>
36  <!--以下、コメント行-->
37  <p class="reg">
38  注 1：シークバーを使って再生位置を変更･移動した場合、次の字幕タイミングから同期が始まります。
38  うまく同期がとれない場合はいったん映像を更新(refresh)してから再生してください。<br>
40  注 2：対応するメディア形式は原則として wmv のみになります。mpg および avi でも動きますが、動作が
41  不安定になることがあります（この場合は、いったん動画ファイルのダウンロードが終了してから再生し
42  てください）。<br>
43  注 3：DirectVobSub などの字幕用プラグインがインストールしてある場合は、あらかじめそれをオフにして
44  おいてください（mpg および avi を再生する場合）。</p>
45  <span class="small">
46  <a href="javascript:parent.window.close()">CLOSE</a></span>
47  <hr width="100%">
48  <table BORDER=0 CELLSPACING=0 CELLPADDING=0 WIDTH="100%">
49    <tr><td class="small">This page created by Yasumasa Someya, June 5,
50  2008 (HTML_Template1.txt) </td></tr>
51  </table></blockquote>
52  </body>
53  </html>
```

サンプルソースコード１のウェブページイメージ

WMP を HTML ファイルに埋め込み、字幕表示させる（単一言語クラス用）

VIDEO このページは Internet Explorer (Ver. 6 or above) で閲覧してください。

Source Info.: 640x480, 521kbps, WMV9, 2.98MB

解説付きのソースコードを開く | SAMIファイルを開く

注1：シークバーを使って再生位置を変更・移動した場合、次の字幕タイミングから同期が始まります。うまく同期がとれない場合はいったん映像を更新 (refresh) してから再生してください。
注2：対応するメディア形式は原則として wmv のみになります。mpg (mpeg-1, 2) および avi でも動きますが、動作が不安定になることがあります（この場合は、いったん動画ファイルのダウンロードが終了してから再生してください）。
注3：DirectVobSub などの字幕用プラグインがインストールしてある場合は、あらかじめそれをオフにしておいてください（mpg および avi を再生する場合）。

> HTML_Template1.html（単一言語クラス）
HTML_Template2.html（言語クラスを選択）
HTML_Template2B.html（言語クラスと文字クラスを選択）> Click to see sample from The Wizard of Oz
HTML_Template3.html（プレーヤの横に字幕フレームを開く）
HTML_Template3B.html（パワーポイントスライドを動画化したスライドショー）
HTML_Template4.html（縦書き表示）
HTML_Template5.html（DirectVobSub を使って字幕を表示）
HTML_Template6.html（フォントサイズを相対指定）
HTML_Template7.html（「ハードサブ」動画の配信）

関連論文
染谷 (2008a)「動画への字幕の挿入について：英語教員のための Do-It-Yourself マニュアル (1)」
染谷 (2008b)「字幕付き動画のオンライン配信について：英語教員のための Do-It-Yourself マニュアル (2)」
染谷 (2008b, Appx)「パワーポイントスライドの動画化について」（染谷 2008b 付録）
染谷 (2008c)「ハードサブ動画の作成について：英語教員のための Do-It-Yourself マニュアル (3)」
染谷 (2008d)「字幕付き動画作成のための SAMI ファイルの作成・編集について：英語教員のための Do-It-Yourself マニュアル (4)」
染谷 (2008d, Appx)「Windows Media ファイルエディタを使った動画ファイルの編集について」（染谷 2008d 付録）

This page was created by Yasumasa Someya, June 5, 2008 (HTML_Template1.html)

※ナビゲーションリンクや書誌情報等のデータはサンプルソースコード中では省略しています（以下同）。
※ウェブ画面のスクロールキャプチャーには EasyCapture (http://www.easy-capture.com/) を使用させていただきました（以下同）。

資料 2（図 3a） サンプルソースコード 2（複数言語クラス用）<HTML_Template2.txt>

```
01  <html>
02  <head>
03  <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
04  <title>ｻﾝﾌﾟﾙｿｰｽｺｰﾄﾞ 2: WMP を HTML ﾌｧｲﾙに埋め込み、字幕表示させる（複数言語クラス用）</title>
05    <style type="text/css"><!--
--      （ここには HTML_Template1.txt にある Lines 07-12 までのスタイル設定をそのまま挿入）-->
12    </style>
13  <!-- onLoad イベントハンドラにより、このファイルの立ち上がりとともに自動的に字幕ファイルと動画ファ
14   イルを読み込み、同時に言語クラス設定情報の取得と字幕データの ID 定義を行う。-->
15  <script language="JScript" for="window" event="onLoad">
16   Player.URL="http://www.dummy.com/Movie/Sample2.wmv?sami=http://www.dummy.com/
17  Subtitle/Sample2.smi";
18   Player.closedCaption.SAMILang=CCLang.value;
19   Player.closedCaption.captioningID="captions";
20  </script>
21  <!-- この URL はダミーです。実際の URL と入れ替えてください。SAMI ファイルが動画と同じフォルダ
26  内にある場合は ?sami= 以下の指定なしで SAMI ファイルが自動的に読み込まれます。別のフォルダにあ
27  る場合は上記のようにその場所 (URL) をフルパスで指定してください。なお、ローカルパスを指定する場合
24 の方法については本文(pp.8-10) を参照してください。-->
25  </head>
26  <!-- ここから HTML の本文開始-->
26  <body text="#000000" bgcolor="#FFFFFF">
27  <blockquote>
28  <br>
30  <p class="big">WMP を HTML ファイルに埋め込み、字幕表示させる（複数言語クラス用）</p>
31  <hr width="100%">
32  <!--ウェブページに WMP を埋め込み、コントロールパラメタを設定-->
33  <object id="Player" title="Windows Media Player"
34          classid="CLSID:6BF52A52-394A-11D3-B153-00C04F79FAA6"
35          width="370" height="340">
36  <param name="autoStart" value="false">
37  <param name="uiMode" value="full">
38  </object>
39  <br>
40  <!--DIV を使って字幕表示のフレームを設定し、字幕内容(id="captions")を出力-->
41  <div id="captions" title="Caption Window"
42       style="width:370; height:70; background:black;">
43  </div>
44  <!--SELECT を使って言語クラス選択メニューを作成。選択肢は SAMILang での指定内容に一致-->
45  <select id="CaptionLanguages" language="JScript" name="CCLang" class="small"
46          onChange="Player.closedCaption.SAMILang=CCLang.value">
47  <option selected value="Japanese">Japanese
48  <option value="English">English
49  </select>
50  <br><span class="small">言語選択ボックス</span>
51  <!--以下、コメント行-->
--  （中略）
63  <hr width="100%">
64  <table BORDER=0 CELLSPACING=0 CELLPADDING=0 WIDTH="100%">
65    <tr><td class="small">This script was written by Yasumasa Someya, June
64  30, 2008, based on a sample script provided in "Adding Closed Captions to
65  Digital Media" (Kim Vum, Microsoft Corp., August 2001). </td></tr>
67  </table></blockquote>
68  </body></html>
```

サンプルソースコード２のウェブページイメージ

WMP を HTML ファイルに埋め込み、字幕表示させる（複数言語クラス用）

VIDEO このページは Internet Explorer (Ver. 6 or above) で閲覧してください。

Subtitle Video Sample
(Soft Sub Version)

一時停止　00:00

Japanese　Source Info.: 640x480, 521kbps, WMV9, 2.98MB

言語選択ボックス

解説付きのソースコードを開く | SAMIファイルを開く

注1：言語選択ボックスの初期値は日本語になっています。英語の字幕を表示させる場合は English を選択してください。再生途中で言語選択を変更した場合、次の同期ポイントから字幕表示が変わります。うまく同期がとれない場合はいったん映像を更新 (refresh) してから再生してください。
注2：対応するメディア形式は原則として wmv のみになります。mpg (mpeg-1, 2) および avi でも動きますが、動作が不安定になることがあります（この場合は、いったん動画ファイルのダウンロードが終了してから再生してください）。
注3：DirectVobSub などの字幕用プラグインがインストールしてある場合は、あらかじめそれをオフにしておいてください（mpg および avi を再生する場合）。

HTML_Template1.html（単一言語クラス）
> HTML_Template2.html（言語クラスを選択） > Click to see sample from *The Wizard of Oz*
HTML_Template2B.html（言語クラスと文字クラスを選択）
HTML_Template3.html（プレーヤの横に字幕フレームを開く）
HTML_Template3B.html（パワーポイントスライドを動画化したスライドショー）
HTML_Template4.html（縦書き表示）
HTML_Template5.html（DirectVobSub を使って字幕を表示）
HTML_Template6.html（フォントサイズを相対指定）
HTML_Template7.html（「ハードサブ」動画の配信）

関連論文
染谷 (2008a)「動画への字幕の挿入について：英語教員のための Do-It-Yourself マニュアル (1)」
染谷 (2008b)「字幕付き動画のオンライン配信について：英語教員のための Do-It-Yourself マニュアル (2)」
染谷 (2008b, Appx)「パワーポイントスライドの動画化について」（染谷 2008b 付録）
染谷 (2008c)「ハードサブ動画の作成について：英語教員のための Do-It-Yourself マニュアル (3)」
染谷 (2008d)「字幕付き動画作成のための SAMI ファイルの作成・編集について：英語教員のための Do-It-Yourself マニュアル (4)」
染谷 (2008d, Appx)「Windows Media ファイルエディタを使った動画ファイルの編集について」（染谷 2008d 付録）

This script (HTML_Template2.html) was writtenh by Yasumasa Someya, June 30, 2008,
based on a sample script provided in "Adding Closed Captions to Digital Media" (Kim Vum, Microsoft Corp., August 2001).

※このサンプルは Internet Explorer 専用です。

資料 3（図 8a） サンプルソースコード３（プレーヤの横に字幕フレームを開く）<HTML_Template3.txt>

```
01  <html>
02  <head>
03   <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
04   <title>サンプルソースコード３：プレーヤの横に大きく字幕フレームを開く</title>
05  </head>
06  <body text="#000000" bgcolor="#FFFFFF">
07  <blockquote><p style="font-family:Arial,Verdana,Helvetica,'ＭＳ ゴシック',
08  sans-serif; font-size:12pt; color:blue;">プレーヤの横に大きく字幕フレームを開く</p>
09  <hr align="left" width="630"><br>
10  <!--テープルを作成。左側のセルにプレーヤ、右側のセルに字幕フレームを配置-->
11  <!--テープルの背景色を silver に、セルの背景色を white にして silver のテープル枠線を残す-->
12  <table style="width:630; height:300; background:silver;">
13    <tr>
14      <!--左側セル：映像表示画面のスタイルを設定-->
15      <td valign="top"
16          style="height:200; width:300; background:white;">
17      <!--Windows Media Player の埋め込み-->
18      <object id="WMP" title="Windows Media Player"
19              classid="CLSID:6BF52A52-394A-11D3-B153-00C04F79FAA6"
20              width="310" height="300"
21              style="margin:2; border:0;">
22      <param name="URL" value="http://www.dummy.com/Movie/Sample3.wmv?sami=
23  http://www.dummy.com/Subtitle/Sample3.smi"> <!-- この URL はダミーです。-->
24      <param name="autoStart" value="false">
25      <param name="uiMode" value="full">
26      <param name="captioningID" value="captions">
27      </object>
28      </td>
29      <!--右側セル：字幕フレームのスタイルを設定。フレームの背景色と文字色は自由に変更可-->
30      <td valign="top"
31          style="width:310; height:280; color:black; background:white;">
32      <!--字幕ファイルを読み込み（スタイル指定は td 要素の中で行っているため div 内では不要）-->
33      <div id="captions" title="Caption Window">
34      <br><br><p style="font-family:Arial,Verdana,'ＭＳ ゴシック',sans-serif;
35  font-size:10pt; color:black; text-align:center;">字幕はここに表示されます。</p>
36      </div></td>
37    </tr>
38  </table>
39  <!--以下、コメント行-->
40  <p style="line-height:150%; font-size:10pt; font-family: Arial,Verdana,
41  Helvetica,'ＭＳ ゴシック',sans-serif; color:black;">
42  注 1：このビデオに関する説明や指示などの本文をここに挿入します。この本文に対するインラインのスタイル要素
43  は適宜変更してください。<br>
--  （中略）
53  <hr align="left" width="630">
54  <table BORDER=0 CELLSPACING=0 CELLPADDING=0 WIDTH="100%" >
55    <tr><td style="font-size:7pt;font-family:Arial,Verdana,Helvetica; color:
56  black;">This page was created by Yasumasa Someya, June 30, 2008 (HTML_
57  Template3. txt).</td></tr>
58  </table></blockquote>
59  </body>
60  </html>
```

サンプルソースコード３のウェブページイメージ

プレーヤの横に大きく字幕フレームを開く

VIDEO このページは Internet Explorer (Ver. 6 or above) で閲覧してください。

最も原始的な脳幹は、生命の脳と呼ばれ、呼吸や体温調節など、生命維持をつかさどります。

Source Info.: 320x240, 1983kbps, WMV9, 5.31MB

解説付きのソースコードを開く | SAMIファイルを開く

注1:このビデオに関する説明や指示などの本文をここに挿入します。この本文に対するインラインのスタイル要素は適宜変更してください。

注2:このサンプルでは、字幕を文単位で表示し、現在位置をチャンク単位で動的に示しています。現在チャンクの表示は、(1) 色＋太字、(2) 下線、(3) 背景色、の3つの方法で例示しています。字幕後半部は、先行する文を消去せずに複数の文を積み重ねていく例を示しています。

注3:上記の3つの方法のうち、(1) (2) は HTMLタグを使い、(3) はスタイルシートを使っています。具体的な指定方法については SAMI ファイルを参照してください。

HTML_Template1.html（単一言語クラス）
HTML_Template2.html（言語クラスを選択）> Click to see sample from *The Wizard of Oz*
HTML_Template2B.html（言語クラスと文字クラスを選択）
> HTML_Template3.html（プレーヤの横に字幕フレームを開く）
HTML_Template3B.html（パワーポイントスライドを動画化したスライドショー）
HTML_Template4.html（縦書き表示）
HTML_Template5.html（DirectVobSub を使って字幕を表示）
HTML_Template6.html（フォントサイズを相対指定）
HTML_Template7.html（「ハードサブ」動画の配信）

関連論文
染谷 (2008a)「動画への字幕の挿入について: 英語教員のための Do-It-Yourself マニュアル (1)」
染谷 (2008b)「字幕付き動画のオンライン配信について: 英語教員のための Do-It-Yourself マニュアル (2)」
染谷 (2008b, Appx)「パワーポイントスライドの動画化について」（染谷2008b 付録）
染谷 (2008c)「ハードサブ動画の作成について: 英語教員のための Do-It-Yourself マニュアル (3)」
染谷 (2008d)「字幕付き動画作成のための SAMI ファイルの作成・編集について: 英語教員のための Do-It-Yourself マニュアル (4)」
染谷 (2008d, Appx)「Windows Media ファイルエディタを使った動画ファイルの編集について」（染谷2008d 付録）

This page was created by Yasumasa Someya, June 30, 2008 (HTML_Template3.html)

資料 4（図 12a） サンプルソースコード 4（字幕の縦書き表示）<HTML_Template4.txt>

```
01  <html>
02  <head>
03   <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
04   <title>サンプルソースコード４：字幕を縦書きで表示する</title>
05    <style type="text/css"><!--
--    （ここには HTML_Template1.txt にある Lines 07-12 までのスタイル設定をそのまま挿入）-->
13    </style>
14  </head>
15  <body text="#000000" bgcolor="#FFFFFF">
16  <blockquote><br>
17  <p class="big">WMP を埋め込み、字幕を縦書き表示させる（IE Ver. 5.5 以上）</p>
18  <hr width="100%">
19  <br>
20  <!--テープルを作成。左側のセルにプレーヤ、右側のセルに字幕フレームを配置-->
21  <table width="380" height="304" border=0" bordercolor="black" bgcolor="black">
22  <tr>
23    <!--左側セル：映像表示画面のスタイルを設定-->
24    <td valign="top" style="width:300; height:200; background:white;">
25    <!--Windows Media Player の埋め込み-->
26    <object id="Player" title="Windows Media Player"
27            classid="CLSID:6BF52A52-394A-11D3-B153-00C04F79FAA6"
28            width="320" height="304">
29    <param name="URL" value="http://www.dummy.com/MyMovie/Sample4.wmv?sami=
30  http://www.dummy.com/MySubtitle/Sample4.smi">
31    <!-- この URL はダミーです。実際の URL と入れ替えてください。-->
32    <param name="autoStart" value="false">
33    <param name="uiMode" value="full">
34    <param name="captioningID" value="captions">
35    <param name="SAMILang" value="Japanese">
36    </object>
37    </td>
38    <!-- 右側セル：字幕フレームのスタイルを設定。writing-mode:tb-rl で縦書き指定-->
39    <td valign="top"
40        style="writing-mode:tb-rl; background:black; color:white;
41        width:65px; height:304px;">
42    <!--字幕ファイルの読み込み（スタイル指定は td 要素の中で行っているため div 内では不要）-->
43    <div id="captions" title="Caption Window"></div>
44    </td>
45  </tr>
46  </table>
47  <!--以下、コメント行-->
--  （中略）
64  <span class="small">
65  <a href="javascript:parent.window.close()">CLOSE</a></span>
66  <hr width="100%">
67  <table BORDER=0 CELLSPACING=0 CELLPADDING=0 WIDTH="100%">
68    <tr>
69      <td class="small">This page was created by Yasumasa Someya, June 5, 2008
70 (HTML_Template4.txt) </td>
71    </tr>
72  </table></blockquote>
73  </body>
74  </html>
```

サンプルソースコード４のウェブページイメージ

WMP を埋め込み、字幕を縦書き表示させる

VIDEO このページは Internet Explorer (Ver. 6 or above) で閲覧してください。

Subtitle Video Sample
(Soft Sub Version)

一時停止　00:00

Source Info.: 640x480, 521kbps, WMV9, 2.98MB

解説付きのソースコードを開く | SAMIファイルを開く

注1：シークバーを使って再生位置を変更・移動した場合、次の字幕タイミングから同期が始まります。うまく同期がとれない場合はいったん映像を更新 (refresh) してから再生してください。
注2：対応するメディア形式は原則として wmv のみになります。mpg (mpeg-1, 2) および avi でも動きますが、動作が不安定になることがあります（この場合は、いったん動画ファイルのダウンロードが終了してから再生してください）。
注3：DirectVobSub などの字幕用プラグインがインストールしてある場合は、あらかじめそれをオフにしておいてください（mpg および avi を再生する場合）。
注4：字幕ペインは3行までの字幕を表示できる幅 (width:65px) に設定されています。必要に応じて width を変更してください。
注5：字詰め (letter-spacing) は SAMI ファイル上で現在 1px に設定されています。もう少し広げたい場合は 2px に 変更してください。
注6：縦書き表示は、ウェブページに埋め込んだ WMP を Internet Explorer (Ver. 5.5 or above) で表示させた場合にのみ有効になります。縦書き指定はウェブページの中でしていますので、スタンドアロンの WMP で字幕を表示させた場合は横書きになります。（SAMI ファイル中に "writing-mode tb-rl" を加えるとスタンドアロンの WMP でも縦書きになるが、字幕ペインは通常の（＝横書き用の）画面下部のままなので、適切な表示にならない。また、縦書き指定は GOM Player および DirectVobSub では無効になる）。
注7：フォントはプロポーショナルフォントではなく、等幅のフォントを用いると文字が揃ってきれいに見えます。また、数字やアルファベット、記号などは全角で記述してください（半角だと横倒しになってしまう）。

HTML_Template1.html（単一言語クラス）
HTML_Template2.html（言語クラスを選択）＝Click to see sample from The Wizard of Oz
HTML_Template2B.html（言語クラスと文字クラスを選択）
HTML_Template3.html（プレーヤの横に字幕フレームを開く）
HTML_Template3B.html（パワーポイントスライドを動画化したスライドショー）
＝HTML_Template4.html（縦書き表示）
HTML_Template5.html（DirectVobSub を使って字幕を表示）
HTML_Template6.html（フォントサイズを相対指定）
HTML_Template7.html（「ハードサブ」動画の配信）

関連論文
染谷 (2008a)「動画への字幕の挿入について：英語教員のための Do-It-Yourself マニュアル (1)」
染谷 (2008b)「字幕付き動画のオンライン配信について：英語教員のための Do-It-Yourself マニュアル (2)」
染谷 (2008b, Appx)「パワーポイントスライドの動画化について」（染谷2008b 付録）
染谷 (2008c)「ハードサブ動画の作成について：英語教員のための Do-It-Yourself マニュアル (3)」
染谷 (2008d)「字幕付き動画作成のための SAMI ファイルの作成・編集について：英語教員のための Do-It-Yourself マニュアル (4)」
染谷 (2008d, Appx)「Windows Media ファイルエディタを使った動画ファイルの編集について」（染谷2008d 付録）

This page was created by Yasumasa Someya, June 5, 2008 (HTML_Template4.html)

※このサンプルは Internet Explorer 専用です。

資料 5（図 14a） サンプルソースコード 5（DirectVobSub による字幕表示）<HTML_Template5.txt>

```
01  <html>
02  <head>
03  <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
04  <title>サンプルソースコード 5：WMP を埋め込み、DirectVobSub を使って字幕をオーバーレイ表示させる
05  </title>
06    <style type="text/css"><!--
07     .big {line-height:150%; font-size:12pt;font-family: Arial,Verdana,
08      Helvetica,'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:blue;}
09     .reg {line-height: 150%; font-size:10pt; font-family: Arial,Verdana,
10      Helvetica,'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:black;}
11     .small {line-height: 150%; font-size:7pt; font-family:Arial,Verdana,
12      Helvetica,'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:black;} -->
13    </style>
14  </head>
15  <body text="#000000" bgcolor="#FFFFFF" onload='alert("DirectVobSub がインストール
16   されていることを確認してください。");'>
17  <blockquote><br>
18  <p class="big">WMP を埋め込み、DirectVobSub を使って字幕をオーバーレイ表示させる</p>
19  <hr width="100%">
20  <!--映像表示部-->
21  <object id="WMP"
22          classid="CLSID:6BF52A52-394A-11D3-B153-00C04F79FAA6"
23          width="640" height="544">
24  <param name="URL" value="http://www.dummy.com/Movie/Sample.mpg">
25  <!-- この URL はダミーです。実際の URL と入れ替えてください。SAMI ファイルは動画ファイルと同じ
26   フォルダ内に置いてください。SAMI ファイルの URL は指定する必要はありません。-->
27  <param name="autoStart" value="false">
28  <param name="uiMode" value="full">
29  <param name="stretchToFit" value="true">
30  <!--以下は IE 以外のブラウザ用-->
31  <embed type="application/x-mplayer2"
32         pluginspage="http://www.microsoft.com/isapi/redir.dll?prd=windows&sbp=
33  mediaplayer&ar=Media&sba=Plugin&"
34         width="640" height="544"
35         src="http://www.dummy.com/Movie/Sample.mpg"
36         showcontrols="1"
37         autostart="0"/>
38  </object>
39  <!--以下はコメント-->
40  <span class="reg">
41  注 1：このファイルは、ウェブページに WMP を埋め込み、DirectVobSub を使って字幕を表示させるための
42  方法を記述しています。使用に当たっては、ローカルパソコンに DirectVobSub がインストールされている
43  ことを前提とします。DirectVobSub については染谷 (2008a) を参照してください。</span>
--  （中略）
75  <hr width="100%">
76  <table BORDER=0 CELLSPACING=0 CELLPADDING=0 WIDTH="100%" >
77    <tr>
78      <td class="small">This page was created by Yasumasa Someya, June 5, 2008
79  (HTML_Template5.txt)</td>
80    </tr>
81  </table></blockquote>
82  </body>
83  </html>
```

サンプルソースコード５のウェブページイメージ

WMPを埋め込み、DirectVobSubを使って字幕をオーバーレイ表示させる

VIDEO DirectVobSub がインストールされていることを確認してください。

▶Source info.: 320x240, 500kbps, MPEG1, 30fps, 4.75MB
▶同期がうまくいかない場合は、いったん一時停止し、ファイルの読み込みが終了してから再生してください(注6)。
▶字幕のフォント設定はローカルの DirectVobSub 上で行います(注4)。

解説付きのソースコードを開く | SAMIファイルを開く

注1:このファイルは、ウェブページに WMP を埋め込み、DirectVobSub を使って字幕を表示させるための方法を記述しています。使用に当たっては、ローカルパソコンに DirectVobSub がインストールされていることを前提とします。DirectVobSub については染谷 (2008a, p.2 および pp. 21-27) を参照してください。
注2:対応しているメディアファイルは原則として mpg (mpeg-1, 2) および avi 形式です。
注3:字幕ファイル (SAMIファイル)はメディアファイルと同じフォルダ内に置いてください(「URL SAMI構文」や SAMIFileName パラメタを使って別のフォルダから呼び出すことはできません)。
注4:メディアファイルが読み込まれると、DirectVobSub が自動的に起動します。画面右下に DirectVobSub のアイコンが表示されるのを確認してください。このアイコンをダブルクリックすると「DirectVobSub のプロパティ」設定画面が表示されます。ここで字幕の書式設定をすることができます。このサンプルでの推奨設定は以下のとおりです。

* DirectVobSub (auto-loading version) の「プロパティ」推奨設定
[Text Settings] で Outline, Shadow, Advanced Renderer をそれぞれ選択。
[フォント設定]
フォント名 = HGｾﾞﾝｺﾞｼｯｸM-PRO (または HG創英角ｺﾞｼｯｸUB)
スタイル = 太字
サイズ = 24
色 = 白
文字セット = 日本語

注5:全画面表示で閲覧する場合(非推奨)は、ローカルのWMPの字幕機能を「オフ」にしておいてください。ウェブページに埋め込んだままの状態で閲覧する場合はローカルWMPの字幕機能はオンでもオフでもかまいません。
注6:回線の状態やCPU性能、あるいは動画のエンコーディングの方法等の影響で、ファイルの読み込みが映像の表示に追いつかないことがあります(とくに初めてアクセスする場合)。この場合は、いったん再生を一時停止し、ファイルの読み込みが終了してから再生を開始してください。
注7:ローカルパソコンに QuickTime(またはその他のメディアプレーヤ)がインストールされている場合は mpg (mpeg) との関連付けをはずしておいてください(QuickTime は勝手に関連付けを戻してしてしまうことがあるので、できればアンインストールしておくほうがよい)。
注8:パソコンの「ハードウェアアクセラレーション」を最大 (Full) にすると最もきれいな字幕表示を得ることができます。詳しくは染谷 (2000a, p.32) を参照してください。

HTML_Template1.html (単一言語クラス)
HTML_Template2.html (言語クラスを選択) > Click to see sample from The Wizard of Oz
HTML_Template2B.html (言語クラスと文字クラスを選択)
HTML_Template3.html (プレーヤの横に字幕フレームを開く)
HTML_Template3B.html (パワーポイントスライドを動画化したスライドショー)
HTML_Template4.html (縦書き表示)
* HTML_Template5.html (DirectVobSub を使って字幕を表示)
HTML_Template6.html (フォントサイズを相対指定)
HTML_Template7.html (「ハードサブ」動画の配信)

関連論文
染谷 (2008a)「動画への字幕の挿入について: 英語教員のための Do-It-Yourself マニュアル (1)」
染谷 (2008b)「字幕付き動画のオンライン配信について: 英語教員のための Do-It-Yourself マニュアル (2)」
染谷 (2008b, Appx)「パワーポイントスライドの動画化について」(染谷2008b 付録)
染谷 (2008c)「ハードサブ動画の作成について: 英語教員のための Do-It-Yourself マニュアル (3)」
染谷 (2008d)「字幕付き動画作成のための SAMI ファイルの作成・編集について: 英語教員のための Do-It-Yourself マニュアル (4)」
染谷 (2008e, Appx)「Windows Media ファイルエディタを使った動画ファイルの編集について」(染谷2008e 付録)

This page was created by Yasumasa Someya, June 5, 2008 (HTML_Template5.html)

資料 6（図 15a） サンプルソースコード 6（フォントサイズを相対指定する）<HTML_Template6.txt>

```
01  <html>
02  <head>
03  <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
04  <title>サンプルソースコード 6: フォントサイズを相対指定し、ユーザによる変更を許す</title>
05  <style type="text/css"><!--
06  .big {line-height:150%; font-size:1.5em; font-family: Verdana,Arial,
07   Helvetica, 'ＭＳ ゴシック', sans-serif; color:blue;}
08  .reg {line-height:150%; font-size:1em; letter-spacing:1px; font-family:
09   Verdana,Arial,Helvetica,'ＭＳ ゴシック', sans-serif; color:black;}
10  .small {line-height:150%; font-size:0.9em; font-family: Verdana,Arial,
11   Helvetica,'ＭＳ ゴシック', sans-serif; color:black;}
12   -->
13  </style>
14  <!--Internet Explorer 以外のブラウザに対するアラート：オプション -->
15  <script language="JavaScript">
16  <!--
17  var b=navigator.appName.toUpperCase();
18  if(b.indexOf("EXPLORER")>=0){void(0);}else{alert("このページは Internet Explorer
18 での閲覧を推奨しています。他のブラウザでは適切に動作しないことがあります。");}
20  //-->
21  </script>
22  </head>
23  <body text="#000000" bgcolor="#FFFFFF">
24  <blockquote>
25  <br><p class="big">フォントサイズを相対指定し、ユーザによるサイズ変更を許す</p>
26  <hr width="100%">
27  <span class="small">このページは Internet Explorer (Ver.6 or above)で閲覧してください。
28  </span><br><br>
29  <span class="small"><font color="red"><u>字幕文字サイズについて</u></font>ブラウザ
30  メニューの「表示」>「文字のサイズ」から任意の表示サイズを選択することができます。通常は「小」または
31  「中」にして閲覧してください。プレーヤをフルスクリーン表示にすると字幕テクストのサイズも拡大して表示
32  されます（注参照）。
33  </span><br><br>
34  <!-- 映像表示部：映像表示サイズの指定、および映像ファイルと字幕ファイルの場所の指定 -->
35  <table border="0" style="width:650; height:20; background:black;">
36    <tr><td></td></tr>
37  </table>
38  <object id="Player" title="Windows Media Player"
39          classid="CLSID:6BF52A52-394A-11d3-B153-00C04F79FAA6"
40          width="650" height="400">
41  <param name="URL" value="http://www.dummy.com/Movie/Sample6.wmv?sami=
42  http://www.dummy.com/Subtitle/Sample6.smi">
43  <!-- この URL はダミーです。実際の URL と入れ替えてください。SAMI ファイルがメディアファイルと
44  同じフォルダ内にある場合は ?sami= 以下の指定なしで SAMI ファイルが自動的に読み込まれます。別のフォ
45  ルダにある場合は上記のようにその場所 (URL) をフルパスで指定します。ローカルフォルダ内にあるメディア
46  ファイにアクセスする場合は相対パスを使います。この場合、SAMI ファイルはメディアファイルと同じフォル
47  ダ内に置いてください。-->
48  <param name="autoStart" value="false">
49  <param name="uiMode" value="full">
50  <param name="stretchToFit" value="true">
51  <param name="captioningID" value="captions">
52  </object>
```

(2/2) に続く

資料 6（図 15a）<HTML_Template6.txt>　(2/2)

```
53  <!--字幕表示部：表示領域のサイズと画面背景色の指定-->
54  <div id="captions" title="Caption Window"
55       style="width:650px; height:90px; background:black;">
56  </div>
57  <!--以下、フルスクリーン表示用ボタン-->
58  <input type="button" value="フルスクリーン" name="FullScrnBTN"
59         title="フルスクリーンボタンは再生中のみ有効です。左ダブルクリックで元に戻ります。"
60         onClick="if (Player.playState==3) Player.fullScreen='true'">
61  <span class="small">     Source Info.: 640x480,
62  521kbps, WMV9, 2.98MB</span>
63  <p class="reg">
64  <a href="http://www.dummy.com/Public/HTML_Template6.txt" target="_blank"
65  title="txt ファイルが html ページとして開いてしまう場合は、ブラウザのメニューから「ツール」→「イン
66  ターネットオプション」→「セキュリティ」→「レベルのカスタマイズ」を選択し、「セキュリティ設定」の「その
67  他」（IE のアイコンが先頭にあるもの）の項目にある「拡張子ではなく、内容によってファイルを開く」を「無効
68  にする」にしてから再度ブラウザを立ち上げてください。">解説付きのソースコードを開く</a> | <a href=
69  "http://www.dummy.com/Public/Sample6.smi" target="_blank" title="SAMI ファイルが
70  他のアプリケーションに関連付けられていると、そのアプリケーションが立ち上がってしまうことがあります。その
71  場合はマウスを右クリックし、「対象をファイルに保存」を選択してください。">SAMI ファイルを開く</a>
72  </p>
73  <!--以下はプレーヤを使用するに当たっての注意書き（コメント）および補足説明 -->
74  <p class="reg">
75  注１：プレーヤを全画面表示させるには「フルスクリーン」ボタンをクリックします。フルスクリーンボタンは
76  再生中のみ有効です。左ダブルクリック（または [ESC] キー）で元に戻ります。<br>
77  注２：フルスクリーン表示で閲覧する場合は、ローカルの WMP の字幕機能を「オン」にしておいてください。
78  フルスクリーン表示の際の字幕文字サイズはおよそ 20 ポイント（固定サイズ）になります。ただし、画面の解像
79  度は荒くなります。<br>
80  注３：対応するメディア形式は原則として wmv のみになります。mpg (mpeg-1, 2) および avi でも動きますが、動
81  作が不安定になることがあります（この場合は、いったん動画ファイルのダウンロードが終了してから再生して
82  ください）。<br>
83  注４：DirectVobSub などの字幕用プラグインがインストールしてある場合は、あらかじめそれを「オフ」にして
84  おいてください（mpg および avi を再生する場合）。<br>
85  <p class="small">
--  （中略＝書誌情報その他のコメント省略）
101 <hr width="100%">
102 <table BORDER=0 CELLSPACING=0 CELLPADDING=0 WIDTH="100%" >
103   <tr>
104     <td class="small">This page was created by Yasumasa Someya, June 5, 2008
105 (HTML_Template6.txt) </td>
106   </tr>
107 </table>
108 </blockquote>
109 </body>
110 </html>
```

※外部スタイルシートの使用について＝巻末**注 28** 参照。

サンプルソースコード６のウェブページイメージ

フォントサイズを相対指定し、ユーザによるサイズ変更を許す

VIDEO このページは Internet Explorer (Ver. 6 or above) で閲覧してください。

字幕文字サイズについて: ブラウザメニューの「表示」>「文字のサイズ」から任意の表示サイズを選択することができます。通常は「小」または「中」にして閲覧してください。プレーヤをフルスクリーン表示にすると字幕テクストのサイズも拡大して表示されます(注参照)。

フルスクリーン　Source Info.: 640x480, 521kbps, WMV9, 2.98MB

解説付きのソースコードを開く | SAMIファイルを開く

注1:プレーヤを全画面表示させるには、「フルスクリーン」ボタンをクリックします。、フルスクリーンボタンは再生中のみ有効です。左ダブルクリック(または[ESC] キー)で元に戻ります。
注2:フルスクリーン表示で閲覧する場合は、ローカルのWMPの字幕機能を「オン」にしておいてください。フルスクリーン表示の際の字幕文字サイズはおよそ20ポイント(固定サイズ)になります。ただし、画面の解像度は荒くなります。
注3:対応するメディア形式は原則として wmv のみになります。mpg (mpeg-1, 2) および avi でも動きますが、動作が不安定になることがあります(この場合は、いったん動画ファイルのダウンロードが終了してから再生してください)。
注4:DirectVobSub などの字幕用プラグインがインストールしてある場合は、あらかじめそれを「オフ」にしておいてください(mpg および avi を再生する場合)。

HTML_Template1.html (単一言語クラス)
HTML_Template2.html (言語クラスを選択) > Click to see sample from *The Wizard of Oz*
HTML_Template2B.html (言語クラスと文字クラスを選択)
HTML_Template3.html (プレーヤの横に字幕フレームを開く)
HTML_Template3B.html (パワーポイントスライドを動画化したスライドショー)
HTML_Template4.html (縦書き表示)
HTML_Template5.html (DirectVobSub を使って字幕を表示)
> HTML_Template6.html (フォントサイズを相対指定)
HTML_Template7.html (「ハードサブ」動画の配信)

関連論文
染谷 (2008a) 「動画への字幕の挿入について: 英語教員のための Do-It-Yourself マニュアル (1)」
染谷 (2008b) 「字幕付き動画のオンライン配信について: 英語教員のための Do-It-Yourself マニュアル (2) 」
染谷 (2008b, Appx) 「パワーポイントスライドの動画化について」 (染谷2008b 付録)
染谷 (2008c) 「ハードサブ動画の作成について: 英語教員のための Do-It-Yourself マニュアル (3)」
染谷 (2008d) 「字幕付き動画作成のための SAMI ファイルの作成・編集について: 英語教員のための Do-It-Yourself マニュアル (4)」
染谷 (2008d, Appx) 「Windows Media ファイルエディタを使った動画ファイルの編集について」 (染谷2008d 付録)

This page was created by Yasumasa Someya, June 5, 2008 (HTML_Template6.html)

【注】

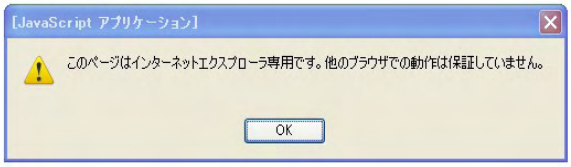
1) 本稿はもともと染谷 (2008a)、染谷 (2008c) および染谷 (2008d) と合わせてひとつの論文として書かれたものである。分割するに当たり、可能な限りそれぞれ独立して読めるように配慮したものの、記述の重複を避けるために、それぞれ他稿で扱った内容を所与の前提として記述されているところがある。本稿を読むに当たっては是非とも他の原稿と併せて読んでいただくようお願いしたい。
2) 授業で使う映像素材の著作権（とくに米国映画をめぐるいわゆる「53 年問題」）については染谷 (2008a) の**注 32** を参照されたい。
3) 大学教員（非常勤を含む）の場合は勤務校の情報センター（情報処理センター、メディアセンター、その他、名称は大学によって異なる）に行き、学内サーバ上に個人ホームページのスペースをもらうように申請する。データのアップロードおよびそれに必要な FTP ソフトウェアの使い方等については、通常、教員向けのマニュアルが整備されているはずなので、それを参照すればよい。
4) QiuickTime や RealPlayer もウェブページに埋め込むこと（および字幕を表示させること）ができるが、字幕形式にそれぞれ独自の規格を採用していること、および正規の製品は有料であることから、本稿では扱わない。なお、これらのプレーヤがすでにパソコン上にインストールされている場合は、あらかじめファイルの関連付け（特に mpg [mpeg] および avi との関連付け）を外しておくことをお勧めする。使用する動画のファイルタイプがこれらのプレーヤに関連付けられていると、動画を立ち上げたときにこれらのプレーヤが勝手に起動してしまい、円滑な作業ができなくなるからである。また、これに伴って次の手順で WMP でのファイル関連付けを確認（または再設定）しておくこと。
   - WMP を立ち上げ、メニューから[ツール] →[オプション] を選択し、[ファイルの種類] タブをクリック。表示される画面から [すべて選択] を選び [OK] ボタンをクリックしてプレーヤを閉じる（これで WMP がデフォルトのプレーヤになるが、前述のとおりしばしば他のプレーヤによってこの設定が強制的に変更されてしまうことがある。その場合は、同様の手順でファイルの関連付けを再設定する必要がある）。
5) <object> タグは Internet Explorer (IE) 以外のブラウザでは無効になる。例えば Safari や Firefox は <object> タグをサポートしておらず、<object> タグで埋め込まれた WMP 等のアプリケーションは立ち上がらない。IE 以外のブラウザで WMP を立ち上げるためには <embed> タグを使って WMP を埋め込めばよいのだが、この方法だと本稿で解説したような方法で SAMI ファイルを読み込み、動画と同期させて字幕を表示させることができない。したがって、字幕を表示させるためには、結局のところ、<object> タグを使って WMP を埋め込み、これを IE で立ち上げる以外に方法がないということになる。ただし、DirectVobSub を使って字幕を画面上にオーバーレイ表示させる場合は基本的にブラウザの種類を問わない―例えば Safari や Firefox でも WMP 対応のプラグインがインストールされていて、かつ <embed> タグを使って WMP が埋め込まれていれば (pp. 31-33)、ブラウザ上の WMP 上で字幕を同期表示させることができる。（→なお、本稿執筆後の調査で、限定つきながら <embed> タグを使った字幕表示が可能なことが判明した。詳しくは本稿末尾 (p. 67) の「追記」参照のこと）。

   以上のような理由で、本稿では IE を使うことを前提としているが、IE 以外のブラウザを使っているユーザのために、「このページは Internet Explorer 専用です」といったメッセージ（アラート）が、必要なときに適切なタイミングでポップアップ表示されるようになっていると都合がよい。これは以下のような JavaScript コードを HTML ページの `<head>`〜`</head>` の間に挿入することで実現できる。

```
<script language="JavaScript">
<!--
```

```
var b=navigator.appName.toUpperCase();
if(b.indexOf("EXPLORER")>=0){void(0);}else{alert("このページは Internet Explorer 専用で
す。他のブラウザでの動作は保証していません。");}
//-->
</script>
```

アラートメッセージの例（左＝Safari; 右＝Firefox）



なお、上記スクリプトの末尾に次のようなコマンド（history オブジェクの back メソッド）を加えると、ブラウザで表示した履歴の 1 つ前の画面に強制的に戻すことができる（したがって、 Internet Explorer 以外のブラウザでは当該ページへのアクセスができなくなる）。

この部分を追加。

```
す。他のブラウザでの動作は保証していません。"); history.back();}
```

6) 染谷 (2008a) では、WMP の字幕表示領域のことを「字幕ペイン (closed caption pane)」と呼んできたが、本稿での解説では多くの場合、字幕表示領域はウェブページ上のフレームとして開かれるため、以降、そのような字幕表示領域を「字幕フレーム」と呼ぶ。

7) ピクセルは「画素」という意味で、ディスプレイ表示や印刷での小さな色の点 (dot) のことを指す。1 ピクセルが実際にどの位の長さになるかはディスプレイやプリンタの解像度によって変化するが（したがって相対単位となる）、一般的な Windows パソコン用の液晶ディスプレイの解像度を約 120 dpi (dots per inch) とすると、1 dot は（1 インチ＝25.4 ミリ を 120 で割って）約 0.212 ミリ となり、これが 1 ピクセルに相当することになる。したがって、width="370" height="340" という指定は、それぞれ実測で 370×0.212　78.44 ミリおよび 340×0.212　72.08 ミリとなる。なお、実際に長さを決めるときは、width="320" height="300"（ここでは height=300 のうち 60 は字幕フレームの高さとして計算しているので映像自体は width="320" height="240" になる）のようにとりあえず基準となる映像サイズを指定し、画面上の表示を見ながらサイズを微調整していくのがよい。基準となるサイズについては染谷 (2008c, p. 23, pp. 26-27) 参照。

8) したがって、HTML ファイルと同じフォルダ内にメディアファイル (Sample.wmv) および字幕ファイル (Sample.smi) が保存されている場合は `<param name="URL" value="Sample.wmv">` という指定だけでよいことになる。ファイルをサーバにあげる前にローカルでテストをする場合もこのように指定する。参考までに、ローカルフォルダ内のメディアファイルにアクセスする場合の URL パラメタ指定の例を示す。

1. HTML ファイルと同じフォルダ（例：[MyMovies]）内にメディアファイルと sami ファイルがある場合

例：　[MyMovies]
　　├ HTML_Sample.html
　　├ Sample.wmv
　　└ Sample.smi

※ URL パラメタ指定： `<param name="URL" value="Sample.wmv">`

→ これは `value="./Sample.wmv"` と書いても同じ（`./` は「現在フォルダ」を指す。なお、

「現在フォルダのひとつ上のフォルダ」を参照する場合は `../` となり、「現在フォルダのひとつ上のフォルダ内にある MyData というサブフォルダ」を参照する場合は `../MyData/` となる）。

2. HTML ファイルと異なるフォルダ（例：[MyFiles] のすぐ下の階層にある [MyMovies]）内にメディアファイルと sami ファイルがある場合

例：　[MyFiles]
　　├ HTML_Sample.html
　　└ [MyMovies]
　　　├ Sample.wmv
　　　└ Sample.smi

※ URL パラメタ指定：`<param name="URL" value="MyMovies/Sample.wmv">`

→ これは `value="./MyMovies/Sample.wmv"` と書くこともできる。

→関連事項として本文 pp. 7-9 も参照。

9) このほか、invisible という値を指定するとプレーヤそのものを隠すことができるが、これはおもにバックグラウンド音楽などを流すときに使われるもので、本稿のように動画の表示を目的とする場合には使用する可能性はないと思われる。

10) **注 8** 参照。なお、外部サーバ上にメディアファイルと字幕ファイルを置く場合、マイクロソフト社は「SAMI URL 構文」を使って一括して呼び出す方法を推奨しているが、時としてブラウザからのアクセスがうまくいかないことがある。このような場合は、以下に示すとおり、URL パラメタにメディアファイルの URL を、SAMIFileName パラメタに字幕ファイルの URL をそれぞれ記入するようにするとうまくつながることがある（以下の URL は架空のもの）。

`<param name="URL" value="http://www.dummy.com/MyMovie/Sample.wmv">`
`<param name="SAMIFileName" value="http://www.dummy.com/MySubtitle/Sample.smi">`

11) DirectVobSub は字幕表示用のプラグイン。詳しくは染谷（2008a, p. 2 および pp. 21-26）参照。

12) 字幕の「2 重表示問題」については染谷（2008a, pp. 25-27）参照。

13) DirectVobSub の起動および初期設定（プロパティ設定）については染谷（2008a, pp. 23-25）を、「オフ」にする方法については同 p. 28 をそれぞれ参照。

14) JScript とはマイクロソフト社が開発したスクリプト言語の 1 つ。ネットスケープ･コミュニケーションズ社とサン･マイクロシステムズ社が共同開発したスクリプト言語である JavaScript をベースに、マイクロソフト社が独自の技術を追加して拡張したもの。JavaScript と互換性があり Windows 上で動作する。このサンプルの場合も <script language="JavaScript" . . . > という指定でも問題なく動く。なお、通常、Web ページ内に追加する拡張スクリプトは <SCRIPT Language="JScript"> のような形式で、使用する言語の指定をするが、この指定が省略された場合は自動的に JScript により記述とみなすようになっている（Internet Explorer を使って HTML ページを開いた場合のデフォルト値）。なお、JScript については MSDN ライブラリに収録されている「JScript について」(Microsoft Corp., ND/2008c) と題する文書に詳しい解説が掲載されている。

15) ウェブページ上のリンクを経由してメディアファイルを読み込む場合、しばしば SAMI ファイルの自動取得に失敗することがある（ただし、通常は、メディアファイルを再読み込みすることで解決する）。これは既知の問題であるが、マイクロソフト社ではこの問題を回避するために SAMI ファイルへのパスを「SAMI URL 構文」を使って明示的に指定することを推奨している。

16) JavaScript や JScript あるいは cgi プログラムによく使われる Perl などのスクリプト言語では、例えば、ピリ

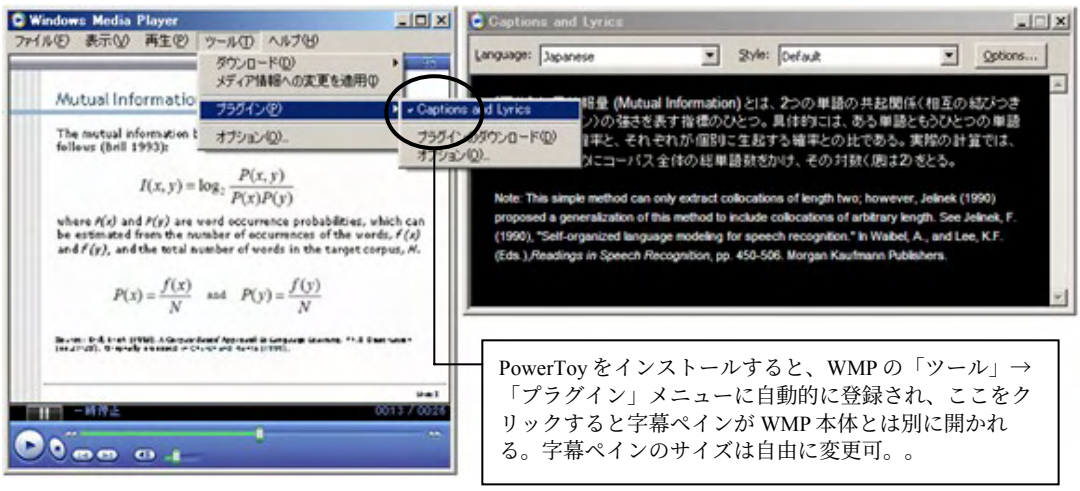
オド (.) が正規表現検索記号として「任意の 1 文字」を意味するなど、一定の文字や記号がプログラム上特殊な意味を持つことがある。これらの文字･記号（通常、「メタ文字」と呼ばれる）を、その文字通りの（本来の）意味で使いたい場合には、その文字･記号の直前に円記号（\）を置き、\. のように表記することになっている。この円記号 (\) 記号を「エスケープ文字」と呼ぶ。スクリプト上で \ 記号をフォルダの区切り記号として使う場合も、その直前にエスケープ文字を添えて \\ のように表記する（英文キーボードの場合、円記号 (\) はバックスラッシュ (\ ) になる）。

17) ここで字幕フレームの背景色 (background) を他の色、例えば blue とすると、青色の画面上に白い文字の字幕テクストが表示されるが、文字列の背景色が黒になる。これは、SAMI ファイルのスタイル指定で文字背景色 (background-color) を黒に指定しているためである。

18) SAMI ファイル中での言語クラスの定義、および複数言語対応の SAMI ファイルの作成等に関しては、染谷 (2008a, pp. 7-11) を参照されたい。

19) DirectVobSub を使用する場合、あるいは GOM Player や ViPlay のような独自の字幕エンジンを搭載したメディアプレーヤを使ってローカルに動画ファイルを立ち上げる場合は、ユーザが自分のパソコン上で自由に字幕の文字サイズを設定することができるため、基本的には問題ない。なお、WMP を使う場合でも、SAMI ファイル内のスタイル設定でフォントサイズを「相対単位指定」（染谷 2008a, p. 6）にすれば、ブラウザ上でのテクストの表示サイズ（ブラウザに埋め込まれた WMP での字幕表示も含む）をユーザが選択できるようになる。これについては、本稿の pp. 14-19 参照。

20) ただし、Captions and Lyrics PowerToy というプラグインをインストールすれば、映像画面と字幕ペインを切り離すことができる（字幕データには SAMI ファイルがそのまま使える）。参考までに、以下に WMP から PowerToy を開いた画面例を示す。



このプラグインは次のサイトから無料でダウンロードすることができる。ダウンロードサイト：http://www.wmplugins.com/ItemDetail.aspx?ItemID=345

21) ただし、厳密には発話開始と「同時」ではなく、実際の発話開始時間より 100～200ms（0.1～0.2 秒）ほど早く字幕表示を開始するのがよいとされている。音声情報に比べ、文字情報は文字[列]（およびその意味内容）の認識に、ごくわずかながら時間が必要なため、その時間をプラスすることで、視聴者はあたかも音声と字幕が正確に同期しているように感じるからである（染谷 2008d, p. 5 および**注 5** 参照）このあたりの問題は字幕認知に関する面白い研究テーマであるが、本稿では立ち入らない。

22) 米国の National Association of the Deaf が Described and Captioned Media Program (DCMP) と呼ばれるプロジェクトの一環として作成している聴覚障害者のための字幕作成ガイドライン *Captioning Keys: Guidelines and Preferred Techniques* (2007) では、1 行当たりの最大文字数を 32 文字とし、これを“ 32-characters-per-line requirement ”と呼んでいる。また、字幕の提示速度については１分当たりの語数（wpm = words per minute）を尺度として、教育用または子供向けのビデオ映像で 120 wpm、大人向けの教育用ビデオでは 150-160 wpm という基準を提示している。

なお、英語の字幕で「1 秒当たり 12 文字」ということは、仮に単語の平均文字数を 4 文字とした場合、180 wpm の速さで字幕を読むことを想定していることになる。これは母語話者にとっては楽に読める速度であるが、平均的な英語学習者にとってはかなり早い速度である。 テクストの提示速度が対象となる学習者の読解能力を大きく超えてしまうと、学習者は字幕を追うのに精一杯になり、多くの場合は読み終わらないうちに次の字幕が提示されてしまうことになる。したがって、語学学習という観点からすれば、表示する字幕の内容を学習者／視聴者のレベルに応じて編集した上で提示するのが望ましい。英語を日本語に翻訳した上で字幕表示する場合は、訳文を調整することで視聴者の能力に合わせた字幕提示レートを確保することができるが、英文をそのまま字幕化する場合には、例えば、It's time to stop talking about it, but we should act immediately before it's too late. という 4 秒／16 語 (240 wpm) ほどの原文を It's time to act before it's too late. (4 秒／8 語= 120 wpm) のように情報の等価性を保ちつつ必要最小限の語数に編集して字幕化することで視聴者の負担を軽くし、結果的に全体の理解を促進することができる。

ちなみに、上記の原文をそのまま和訳するとすれば、「（この問題について）議論をするのはもうやめるべきだ。手遅れにならないうちに、直ちに行動を起こすべきである」といった内容になるが、もちろんこれでは字幕として成立しない。したがって、字幕翻訳者は、この発話を「１秒当たり 4～5 文字、1 行当たり最大 13 文字で 2 行まで」という大原則を考慮に入れて、字幕として再構成することになる。結果的には、おそらく「手遅れにならないうちに／動いたほうがいい」といった内容になると思われるが、このような訳文を成立させる「原則」は、前記の英文の書き換え例を成立させている「原則」（例えば、発話記銘の初頭効果 (primacy effect) や最新効果 (recency effect) 、あるいは情報の焦点 (information focus) といった概念）と基本的には同じものである。字幕翻訳というと、通常、異言語間翻訳 (interlingual translation) を指すが、その前提となる語学力（ここでは英語）や総合的な言語運用能力を鍛えるための訓練として、英語による映画やドキュメンタリー、あるいはニュース放送をそのまま聞き取り、これを（必要に応じて）前記のような編集作業を加えた上で字幕として表示させるという作業を学生にさせることで、単なる聞き取り訓練だけでなく、英語の創造的産出能力を含む総合的な言語運用能力を同時に鍛えることができる。翻訳とはこのような編集プロセスを経て最終的な訳文の生成に至るものであり、同一言語内字幕化 (intra-lingual subtitling）の作業は、「字幕翻訳」訓練の 1 過程としても妥当かつ重要な訓練のひとつであると考えられる。

23) writing-mode 属性は、字幕フレーム内でのテクスト表示のほかに、HTML ページのテクスト表示にも適用することができる。ただし、writing-mode 属性は HTML の <body> 要素に対して直接適用することができないという制限があるため、ウェブページの本文全体を縦書きにするためには、以下に示すように本文全体をひとつの 要素で括り、その中でスタイル指定をすることになる（このほか、<head>～ </head> 内で div に対して一括してスタイル指定をしてもよい）。

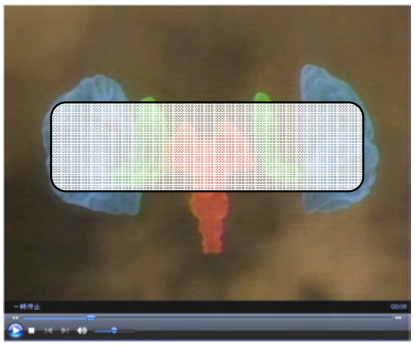
```
<body>
  <div style="writing-mode:tb-rl; font-family:'ＭＳ ゴシック'; font-size:10pt;
```

```
    line-height:150%; padding:5%;">
    ここに本文を入れる。この本文は HTML ページでは縦書き表示になる。テクストの表示方向は
    上 から下、右から左。ただし、半角英文字および洋数字は右に 180 度回転した形で縦方向に配置
    される。
    </div>
  </body>
```

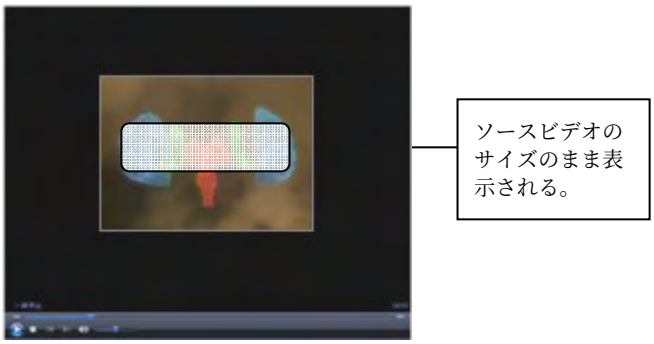
24) <ruby>タグは正式サポートの対象にはなっていないようであるが、IE Ver. 7 および WMP の Vers. 10/11 で問題なく動作することを確認している。ただし、WMP 以外のプレーヤおよび字幕プラグインでは未対応。

25) wmv 形式の動画は DirectVobSub では非対応。したがって、wmv 形式の動画を DirectVobSub を使って字幕表示させるためには、あらかじめファイル形式を mpg (mpeg-1, 2) に変換してから使用する。

26) マイクロソフトによると、stretchToFit 属性は、ビデオコントロールに定義された幅と高さまでビデオを引き伸ばすかどうかを示す値を指定する (http://msdn.microsoft.com/en-us/library/bb249368(VS.85).aspx)。ソースビデオが指定のサイズより小さい場合に有効で、拡大表示を許可する場合は“ true ”を、しない場合は“ false ”を指定。以下の例では左側が“ true ”右側が“ false ”（ソースビデオ＝320×240; 表示枠指定＝650×400 の場合）。ただし、mpeg ビデオの場合はこの指定の有無にかかわらず自動的に指定枠のサイズに拡大表示されることから、stretchToFit 属性の適用対象は wmv 形式のビデオに限定されるものと思われる。



Player.StretchToFit = True



Player.StretchToFit = False

27) ここでは「画像キャプチャーソフト」を使ってパワーポイントスライドをキャプチャーする方法を紹介しているが、このほか Windows のスクリーンキャプチャー機能を使うこともできる。この場合の方法は以下のとおり。

1. 目的のスライドを全画面表示（プレゼンテーション用の全画面表示）させる
2. [Fn] + [PrtSc] キーを押して表示画面をキャプチャー（キャプチャーされたデータはメモリ内に一時保存される）。
3. Windows のスタートメニューから [プログラム] → [アクセサリ] と進み、「ペイント」（Windows 付属の画像編集ソフト）を立ち上げる。
4. 「ペイント」のメニューから [編集] → [貼り付け] を選択。これで先ほどキャプチャーしたパワポ画面が「ペイント」のキャンバス上にコピーされる。
5. 同じく「ペイント」のメニューから [ファイル] → [名前を付けて保存] と進み、保存画面で JPEG 形式を選択した上で、適当な名称で画像ファイルを保存。

以上の要領で、必要なスライドをすべてキャプチャーして保存。その後、Windows ムービーメーカでの作業に移る。Windows のスクリーンキャプチャー機能は画面全体をキャプチャーするもので、通常の「画

像キャプチャーソフト」のように必要な箇所だけを切り取ることができないという問題点があるが、この例のように、全画面表示させたパワポの画面をそのままコピーする場合には便利な機能である。なお、キャプチャーした画像を拡大縮小する場合は、オリジナル画像の縦横サイズの比率を変えないようにすると比較的きれいなイメージを得ることができる。。

28) 本稿で紹介した html サンプルでは、資料 3 (HNTML_Template3.html) を除いていずれも html 内にスタイルシートを記述しているが、ソースコードをすっきりさせるために、html 内からスタイルシートの記述をなくし、外部ファイルとして読み込むようにすることができる。手順は以下のとおり。

1. html ファイルから以下のスタイルシート部分を切り取って、拡張子.css の任意のファイル名（ここでは仮に myStyle.css とする）のテキストファイルとして保存（ただし、スタイルタグとコメントタグは削除＝以下の例参照）。myStyle.css は html ファイルと同じフォルダ内に置く。

   ~~<style type="text/css"> <!~~ この行は不要

   ```
   .big {line-height 150%; font-size:12pt; font-family:Arial,Verdana,Helvetica,
   'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:blue;}
   .reg {line-height:150%; font-size:10pt; font-family:Arial,Verdana,Helvetica,
   'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:black;}
   .small {line-height:150%; font-size:7pt; font-family:Arial,Verdana,Helvetica,
   'ＭＳ ゴシック','ＭＳ Ｐゴシック',sans-serif; color:black;}
   ```

   ~~</style>~~ この行は不要

2. 次に、同じ html ファイルの <head>～</head> 内（＝先ほど切り取った部分）に、以下の行を挿入。

   ```
   <link rel="stylesheet" type="text/css" href="myStyle.css" />
   ```

3. 以上で設定は終了。インラインのスタイルシート指定（html の本文内で style 属性を使って個別に指定してあるもの）はそのままでよい。

［補足］前記 2 で挿入した行は link 要素と呼ばれ、html の head 要素の中で使われて html 文書をスタイルシートやスクリプト、代替文書などのさまざまなリソースと結びつける働きをする。通例、link 要素はリンクの関係を示す rel 属性およびリンク先を指定する href 属性を組み合わせて使われ、たとえば外部スタイルシートを指定するためには <link rel="stylesheet" type="text/css" href="*filename*.css" /> という形になる。rel 属性の値はリンクタイプと呼ばれ、リンク先のリソースとの関係を表す。今回の例のように、別のファイルとして用意したスタイルシートを使う場合にはリンクタイプ名を stylesheet とする。

【追記】（**注 5** 参照）

本稿執筆後の調査で、<embed> タグを使った字幕表示（したがって <object> タグに対応していないブラウザでの字幕表示）が、限定付きながらサポートされていることが判明した。具体的には、<embed> タグ内においても SAMI ファイルを読み込むための `SAMIFileName`, `SAMIStyle`, `captioningID` などの各種パラメタの使用が可能であり、さらに <object> タグの `URL` パラメタは <embed> タグの `src` パラメタで代用可能、ということになっている（マイクロソフトサポートオンライン掲載情報＝文書番号：436594、最終更新日：2003 年 11 月 12 日、リビジョン：1.0 [URL] http://support.microsoft.com/kb/436594/ja ＝による）。したがって、サンプルソースコード 5 の場合と同じく、以下のコードを <object> タグ内に埋め込むことで、Internet Explorer 以外のブラウザでも SAMI ファイルの読み込みと字幕の同期表示が可能になる（このうち、`width`, `height` および `src` の値には <objedct> タグ内で指定した画像サイズ、およびソース映像ファイルの取得先＋ファイル名を記入する）。

```
<!-- 以下は IE 以外のブラウザ用-->
<embed type="application/x-mplayer2" name="MediaPlayer"
       pluginspage="http://www.microsoft.com/isapi/redir.dll?prd=windows&sbp=
mediaplayer&ar=Media&sba=Plugin&"
       width="***" height="***"
       src="******"
       captioningID="captions"
       showcontrols="1"
       autostart="0"/>
```

ただし、IE 固有の機能および仕様にかかわるものについては他のブラウザでは対応しない。したがって、本稿で扱ったサンプルソースコードのうち、言語クラスおよび文字サイズの選択を含む HTML_Template2.html と HTML_Template2B.html 、および縦書き字幕を扱った HTML_Template4.html の 3 つについては <embed> タグによる記述の有無にかかわらず IE 以外のブラウザでは機能しない。以下に、対応関係を整理しておく。

サンプルソースコード	<object>	<embed>	関連付加機能
HTML_Template1.html	○	○	--
HTML_Template2.html	○	×	言語クラス選択
HTML_Template2B.html	○	×	言語クラス／文字サイズ選択
HTML_Template3.html	○	○	フレームを横に開く
HTML_Template3B.html	○	○	パワポ動画プレゼンテーション
HTML_Template4.html	○	×	縦書き字幕
HTML_Template5.html	○	○	DirectVobSub
HTML_Template6.html	○	○	字幕サイズ相対指定（フルスクリーンボタンは無効）
HTML_Template7.html	○	○	ハードサブ

なお、<embed> タグによる指定が有効なものについては、ソースコード中に <embed> タグを新たに追加した。ただし、ブラウザごとの内部仕様の違いのため、必ずしも IE と同等の表示が得られるわけではない。したがって、推奨ブラウザはあくまでも Internet Explorer であることに変わりはない。

【参考文献･資料※】

[1] DCMP (2007) "Captioning Key: Guidelines for Preferred Techniques." Described and Captioned Media Program (DCMP), The National Association of the Deaf, U.S.A. [Online] http://www.dcmp.org/About/Info/Articles.aspx

[2] Gottlieb, Henrik (1998). “Subtitling” in M. Baker (ed.) *Encyclopaedia of Translation Studies*. London and New York: Rutledge.

[3] Hatim, B. & Mason, I. (1997). *The Translator as Communicator.* London and New York: Rutledge.

[4] Microsoft Corp. (2000/2002) 「Web ページへの Windows Media の追加」（日本語版最終更新日 2002 年 3 月 20 日）[Online] http://www.microsoft.com/japan/msdn/windowsmedia/production/addwmwebpage.aspx

[5] ---------- (2000/2008a) 「Windows Media Player Active X コントロール」（MSDN ライブラリ > SDK ドキュメント > WMP 7 SDK) [Online] http://msdn.microsoft.com/ja-jp/library/cc392567.aspx

[6] ---------- (2000/2008b) 「Player オブジェクト」（MSDN ライブラリ > SDK ドキュメント> WMP 7 SDK > WMP Active X コントロール ）[Online] http://msdn.microsoft.com/ja-jp/library/cc410995.aspx

[7] ---------- (2000/2008c) 「ClosedCaption オブジェクト」（MSDN ライブラリ > SDK ドキュメント> WMP 7 SDK > WMP Active X コントロール ）[Online] http://msdn.microsoft.com/ja-jp/library/cc429725.aspx

[8] ---------- (2000/2008d)「Web ページ内のスクリプトの簡単な使用例」(MSDN ライブラリ > SDK ドキュメント > Windows Media Player 7 SDK) [Online] http://msdn.microsoft.com/ja-jp/library/cc364921.aspx

[9] ---------- (2000/2008e)「Internet Explorer 5.5 における縦書きレイアウトの使用例」（日本語版最終更新日 2000 年 11 月 2 日）[Online] http://www.microsoft.com/japan/msdn/web/ie/ie55/verticaltext.aspx

[10] ---------- (2001) "Adding Closed Captions to Digital Media" (MSDN Library, written by Kim Vum, Microsoft Corp.) [Online] http://msdn.microsoft.com/en-us/library/ms867679.aspx

[11] ---------- (2002) “SAMI 1.0 Demonstration Suite” (An Overview of SAMI with s sample media clip and scripts) [Online] http://download.microsoft.com/download/f/d/f/fdf0ba36-42cf-487c-ba4c-eb6f5e6e3715/SAMIOverview.exe

[12] ---------- (ND/2008a) “PARAM Elements in an OBJECT Element” (MSDN Library > WMP11 SDK > WMP Object Model > Using the WMP Control in a Web Page) [Online] http://msdn.microsoft.com/en-us/library/bb614511(VS.85).aspx)

[13] ---------- (ND/2008b) “ClosedCaption Object ” (MSDN Library > WMP11 SDK > WMP Object Model > WMP Object Model Reference) [Online] http://msdn.microsoft.com/en-us/library/bb262163(VS.85).aspx

[14] ---------- (ND/2008c) 「JScript について 」(MSDN ライブラリ > Visual Studio ドキュメント) [Online] http://msdn.microsoft.com/ja-jp/library/3bf5fs13(VS.80).aspx

[15] ---------- (ND/2008d) 「JScript 言語の紹介 」 (MSDN ライブラリ > Visual Studio ドキュメント) [Online] http://msdn.microsoft.com/ja-jp/library/t895bwkh(VS.80).aspx

[16] 香取淳子 (2007) 「高齢者にとっての TV 字幕の適切性についての研究」県立長崎シーボルト大学国際情報学部紀要第８号 (pp. 65-79)

[17] 染谷泰正 (2008a) 「動画への字幕の挿入について：英語教員のための Do-It-Yourself マニュアル (1) 」

[18] ---------- (2008c) 「『ハードサブ』動画の作成について： 英語教員のための Do-It- Yourself マニュアル (3)」

[19] ---------- (2008d) 「字幕付き動画作成のための SAMI ファイルの作成・編集について： 英語教員のための Do-It-Yourself マニュアル (4)」

[20] 田中武人 (2005) 「映像翻訳」 安西徹雄･井上健･小林章夫（編）『翻訳を学ぶ人のために』収録（第 5 章 pp. 135-151）世界思想社

[21] 宮本一郎、鈴木雄介、竹内晃一、長嶋祐二 (2006) 「手話と字幕の同時呈示に対する見やすさについて： 評価実験の実施」 電子情報通信学会技術研究報告 Vol.106, No.285 (pp. 89-94). 社団法人電子情報通信学会

※ ND [=Not Dated] は資料に初出の発行（または掲載）年月が明示されていないものを示す。URL はいずれも 2008 年 6 月 30 日現在のもの）

「ハードサブ」動画の作成について

～ 英語教員のための Do-It-Yourself マニュアル (3) ～

染谷 泰正
（青山学院大学）

はじめに

筆者の勤務校では、文学部英米文学科の専門科目（3, 4 年生対象）のひとつとして「翻訳 (3)」という授業を配置し、「映像翻訳」(Subtitle Translation) を扱っている [1)]。授業では、字幕作成用のソフトウェア（字幕エディタ）として Subtitle Workshop を使い、作成した字幕を Viplay というメディアプレーヤを使って映像と同期して表示させるという方法をとってきた [2)]。

上記ソフトウェア導入時の経緯もあって、本授業では当初から Viplay の使用を前提としており、字幕データもこれに対応した形式（vsf 形式）で作成・保存していた。そのため、Viplay 以外のメディアプレーヤでの視聴ができなかったが、字幕付き動画を、現在もっとも普及しているメディアプレーヤである Windows Media Player (WMP) でも視聴できるようにしたいという声が、学生のみならず、同様の授業実践に関心を持つ他校の先生方からも少なからず寄せられていた。また、苦労して作成した作品を（著作権上の問題がないことを条件として）インターネット上で配信したいという希望も多く寄せられている。このためには WMP に対応させることが必須要件のひとつになってくるが、その他にも、ユーザ（とりわけ国外のユーザ）側の環境によっては日本語の字幕が文字化けすることがあるなどの問題があった。

WMP にはすでに字幕表示機能が備わっており、これに対応した字幕データの作成も、実はごく簡単な方法で実現することができる [3)]。これについては、染谷 (2008a) および染谷 (2008b) ですでに解説したとおりである。ただし、ユーザ側の環境や字幕データの形式にかかわらず、安定した字幕付き動画を作成・配信するためには、従来の「ソフトサブ」方式（字幕データと映像をソフトウェアによって動的に制御し、同期させる方式）ではなく、いわゆる「ハードサブ」方式の字幕付き動画を作成する必要がある。

「ハードサブ」とは、字幕データを映像上に直接焼きこむ方式で、ソフトサブ方式と異なり、再生するメディアプレーヤの種類や字幕ファイルの形式を問わず、また字幕データが文字化けす

SOMEYA, Yasumasa (2008c). “Creating ‘Hard-Sub’ video clips.” (Part 3 of the monograph, *Adding subtitles to video clips and embedding them in your Web pages: A Do-It-Yourself manual for teachers of English wishing to introduce subtitle translation as part of their classroom activities*. 2008)
(c) 2008 Yasumasa Someya

るといった心配もない。ポッドキャスティングによる配信や、YouTube や Yahoo! Video などの動画サイトに字幕付きビデオを投稿をする場合も、ハードサブ方式によって字幕を直接、映像に焼き付けたものを使うことになる。

ハードサブ動画の作成は、字幕データの焼き込みという手間がかかることと、データの焼き込みにともなう技術的な問題のために、一般のユーザにはなかなか手の出しにくいものであったが、実際にはそう難しいものではない。本稿では、まず第 1 章で VirtualDub というソフトウェアを使ったハードサブ作成方法について解説する。VirtualDub はハードサブ動画作成の定番とされているソフトウェアで、使いこなすためには一定の予備知識を必要とするが、本稿では手順どおりに作業を進めていくだけで済むように、説明を必要最小限のポイントに絞り、できるだけわかりやすく、具体的に解説する。第 2 章では、DivXLand Media Subtitler というフリーウェアを使ったハードサブの作成について解説し、続いて第 3 章で TMPG Encoder、第 4 章で GOM Encoder という市販のエンコーダを使ったハードサブの作成方法について述べる。後者の 2 つのエンコーダは有料の市販品であるが、いずれもごく簡単な操作で高品質のハードサブの作成ができること、および試用版を業者のサイトから無料でダウンロードすることができることから、あえて本稿で紹介することとする。

図 1 にウェブページに埋め込んだハードサブ動画の例を示す。当然のことながら、ハードサブ動画は、特殊なコーデックを使って圧縮していない限り、WMP（あるいはその他の同等のメディアプレーヤ）を単独で立ち上げて再生することもできる。

図 1　ウェブページに埋め込んだハードサブ動画の例 ※

画面サイズ指定 (px)
width=640, height=544

[動作環境]
Windows XP (SP2)
WMP Ver. 7 and above

[ブラウザ]
Internet Explorer Ver. 7.0
Safari (3.1.2)
Mozilla Firefox (3.0.1)

※ <object> タグと <embed> タグを併用した WMP の埋め込みについては染谷 (2008b p. 56) の資料 5 (Lines 20-38) 参照。図 1 のソースコードは以下の URL からダウンロード可。
[Online] http://www.someya-net.com/83-SubtitleVideo/Public/HTML_Template7.txt

1. VirtualDub を使った「ハードサブ」動画の作成

VirtualDub はハードサブ動画を作成する際の定番とされているフリーウェアである。本節では、これを使ったハードサブ動画の作成について解説する。まず、作業に必要な 3 つのアプリケーションと、それぞれのダウンロード先を以下に示す。ダウンロードおよびインストールの方法については順次解説する。なお、当然のことながら、以下の作業のためには適当な動画ファイル（通常は avi または mpeg-1 形式）とこれに対応した字幕ファイルが必要である。[4)]

1.1　使用するアプリケーション

1) **VirtualDub 1.8.3** (32-bit version)

ダウンロード先： http://www.virtualdub.org/

日本語化パッチ： http://www.geocities.co.jp/SiliconValley-Cupertino/9994/

コメント：VirtualDub は映像ファイルに字幕データを焼き込むハードサブ作成用のソフトウェア。字幕データのフォント設定や映像ファイルの編集もできる（2008 年 6 月末日現在の最新バージョンは 1.8.3）。上記の日本語化パッチを当てることで、操作画面の表示を日本語に変換することができる。

対応動画形式：avi および mpeg -1 (wmv は非対応。出力は avi のみ)

対応字幕ファイル形式： srt, sub, ssa, ass, smi, psb

2) **VobSub 2.23**

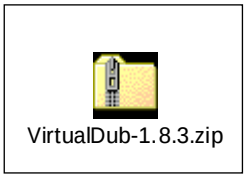
ダウンロード先： http://www.divx-digest.com/software/vobsub.html

日本語化パッチ： http://homepage2.nifty.com/sophia0/freesoft/VobSub/

コメント：VirtualDub 用の各種ユーティリティプログラムを含むプラグインセット。本稿での作業にはこのうちの **TextSub** (aka, TextSub for VirtualDub and AviSynth) というプラグインを使用（関連解説は染谷 2008a, p. 21 および**注 26** 参照）。

参考オンラインマニュアル（英語版）：“VobSub Subtitle Ripping, Playback and Permanent Subtitle Guide” (http://www.divx-digest.com/articles/article_vobsub_page1.html) 。

1.2　VirtualDub のダウンロードとインストール



1) 前記のダウンロード先から VirtualDub をデスクトップ上にダウンロード。ダウンロードが終わると右のようなアイコンがデスクトップ上に表示される。圧縮ファイル（zip ファイル）になっているので、これをダブルクリックして解凍処理。解凍が終わると（とくに場所を指定しない限り）デスクトップ上に新たなフォルダ [VirtualDub-1.8.3] が作成されるので、これを c ドライブの [Program Files] の中にそのまま移動する。

2) [オプション] 日本語化パッチを当てる場合は、パッチファイル (virtualDub165_jp01.exe) を前記のサイトからダウンロードし、これを先ほどの [VirtualDub-1.8.3] フォルダに移動した上でパッチファイルを実行（実行後は VirtualDub の操作パネル表示が日本語化される）。

1.3 Vobsub のダウンロードとインストール

1) 次に Vobsub (Vobsub_2.23.exe) を前記のサイトからデスクトップ上にダウンロード。ダウンロードが終わると右のようなアイコンがデスクトップ上に表示されるので、これをダブルクリック。

2) 画面上の指示に従ってインストールを進める。[Choose Components] の画面（**図 2**）で、[Plugins] 内の次の 3 つの項目にチェックマークを入れて Next をクリック（[Utilities] のところは初期設定のままでよい。ただし、字幕ファイルの編集に秀丸や TeraPad などのテキストエディタを使う場合はこの中の「notepad への関連付け」を外す）。

☑ VobSub for DirectShow

☑ VobSub for VirtualDub

☑ Textsub for VirtualDub and Avisynth ← 今回使用するのはこのプラグイン

なお、すでに染谷 (2008a) の指示にしたがって VobSub for DirectShow (=DirectVobSub) をインストールしてある場合は、ここで VobSub for DirectShow（および Utilities）を再インストール（上書きインストール）することになる。

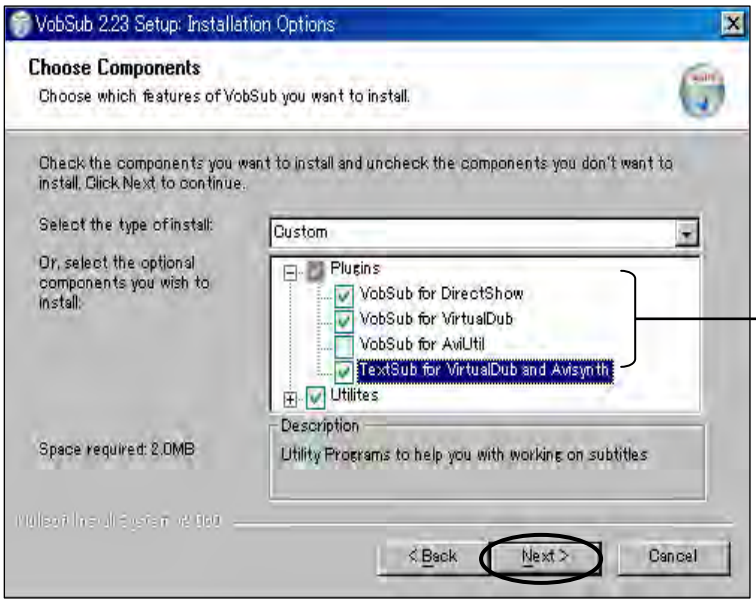


図 2 VobSub のインストール画面（インストールするコンポーネントの選択）

☑ VobSub for DirectShow
☑ VobSub for VirtualDub
☑ Textsub for VirtualDub and Avisynth
の 3 つの項目にチェックマークを入れて Next をクリック。

3) 上記画面で Next をクリックすると、次にインストールする場所 (Destination Folder) を尋ねてくるので、入力ボックス内に c:¥Program File¥Gabes¥VobSub と入力（ただし、通常は自動的にこれが表示されるので、それを確認）した上で、Install をクリック。

4) インストレーションの途中で VirtualDub が保存されている場所を尋ねてくるので（**図 3**）、先ほど保存したフォルダ (c:¥Program File¥VirtualDub-1.8.3) を指定し、Next をクリック。
→すでに VirtualDub が適切にインストールされている場合には自動的に必要なファイルが読み込まれるので、このダイアログが省略されることがある。

5) **図 3** で Next をクリックすると、しばらくして [Installation Complete] というウィンドウが表

示される。これでインストレーションは終了。画面上に表示されているウィンドウを手動で閉じる。ただし、場合によってはウィンドウが閉じないケースがあることが報告されている。その場合は Ctrl+Alt+Del キーを押して Windows のタスクマネージャを立ち上げ、アプリケーションを強制終了させてかまわない。

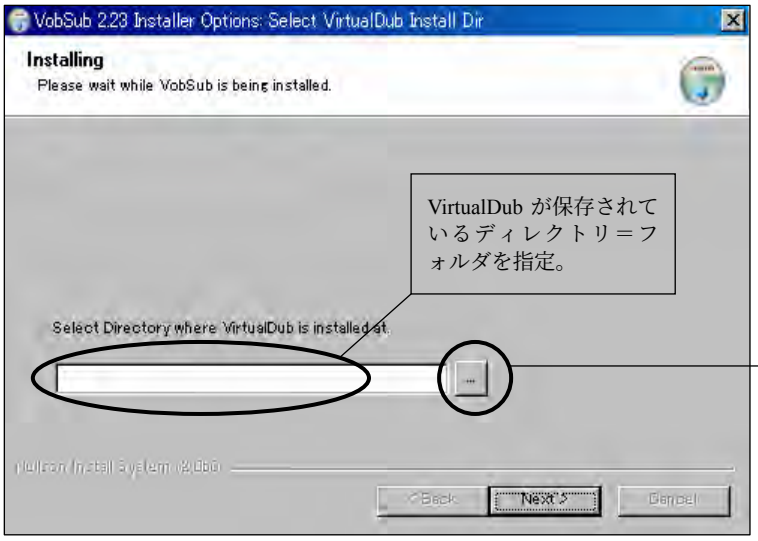
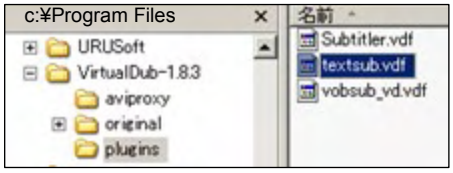


図 3　VobSub のインストール画面（VirtualDub フォルダの指定）

ここをクリックして、先ほど c ドライブの Program File の中に保存した VirtualDub-1.8.3 フォルダを探す。見つかったら、フォルダをクリックして確定。（→VirtualDub がインストールされている場合はこのダイアログが省略されて自動的に処理されることがある）。

6) なお、インストールが正しく行われていれば右図に示すように、c ドライブの [Program Files] にある [VirtualDub-1.8.3] のフォルダの中の [plugins]フォルダに **textsub.vdf** というファイルが読み込まれているはず。このファイルがない場合は再度インストール作業をやり直す。[5)]



以上で必要なソフトウェアのインストールは終了。以下、VirtualDub を使った字幕データの焼き込み作業に移る。なお、この作業に先立って DirectVobsub をオフにしておくこと。[6)]

1.4　VirtualDub の起動およびハードサブ作成の手順

1) 前記 1.2 でインストールした VirtualDub (VirtualDub.exe) を起動（c ドライブの Program Files に移動した [VirtualDub-1.8.3] というフォルダの中に右のようなアイコン (VirtualDub.exe) があるので、これをダブルクリックしてプログラムを起動（同じようなアイコンが他にもあるので注意！）。

2) VirtualDub.exe を起動したら、メニューから [File] → [Open video file] を選択し（**図 4**）、映像ファイルを読み込む。なお、読み込み可能な動画形式は avi, mpg (mpeg-1), divx で、wmv には非対応（したがって、wmv 動画を使いたい場合はいったん mpeg-1 形式に変換する）。

図 4　VirtualDub の初期画面（部分）からビデオファイルを読み込む

3)　映像ファイルが読み込まれたら、メニューから [Video] →[Filters] を選択（**図 5**）。

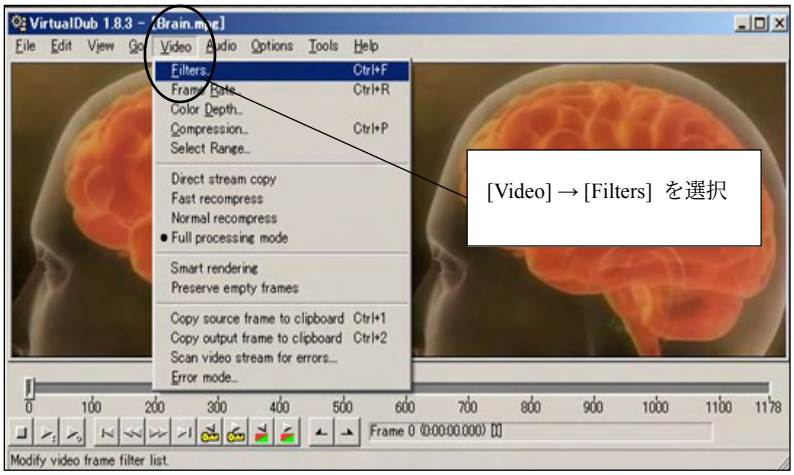


図 5　映像を読み込み後、[Video] →[Filters] を選択

4)　[Filters] というウィンドウ（この時点では空白）が出るので、右側ペインにある [Add] をクリック（**図 6**）。

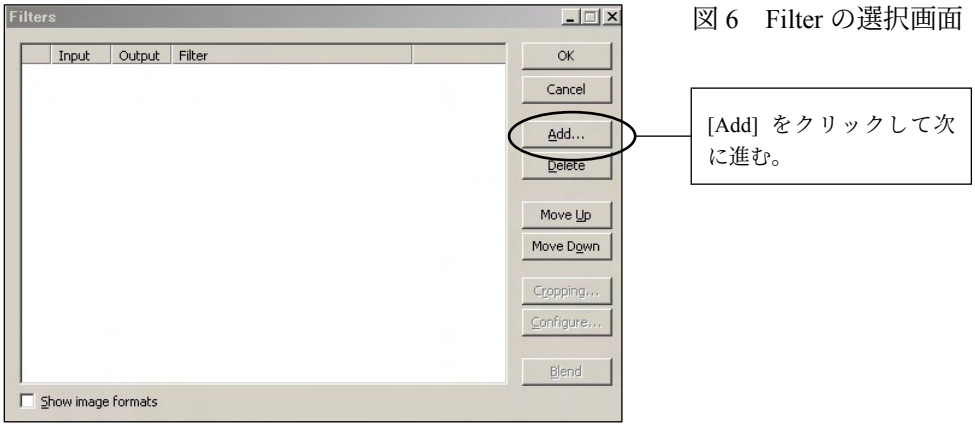


図 6　Filter の選択画面

5)　次に [Add Filters]（フィルタの追加）というウィンドウ（**図 7**）が出るので、そこで [**TextSub2.23**] を選択し、[OK]ボタンを押す。[7)]

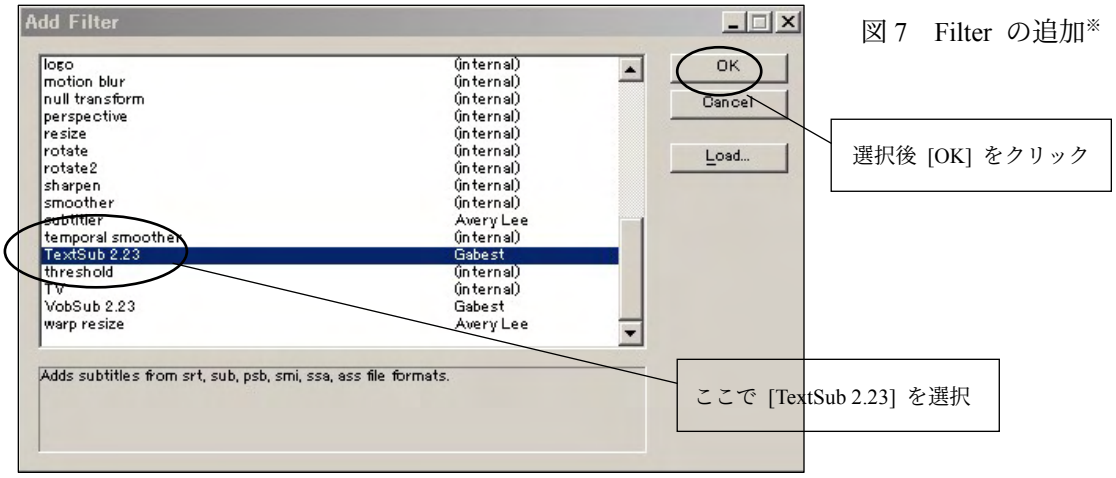


図 7　Filter の追加※

※ここでは一覧表の中から必要に応じて任意のフィルタを複数追加することできる。
pp.11-13 では、このうち映像サイズを変更するための resize というフィルタを
追加した例を示した。

6) 前記画面で [OK]ボタンをクリックすると、次に [TextSub settings] というウィンドウが表示される（**図 8**）。このウィンドウの [OPEN] ボタンを押すと字幕ファイルの選択画面になるので、あらかじめ用意しておいた字幕ファイルを選択して [OK] ボタンを押す。

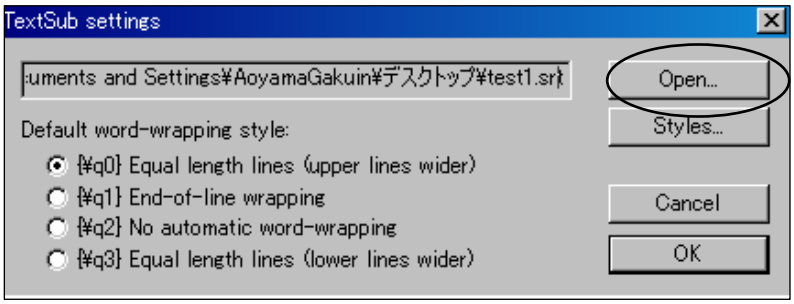


図 8　字幕ファイルの読み込み※

※字幕のファイル形式は smi, srt, ssa, sub, ass, psb がサポートされている。
したがって SAMI 形式（smi）で字幕ファイルが作成してあれば通常はそのまま読み
込むことができる。[8)]

7) 次に上記画面から [Styles] ボタンをクリック。次ページに示した Subtitle Style Editor 画面（**図 9a**）が表示されるので、ここで字幕のスタイルを設定する。設定する内容は、字幕表示位置、左右上下マージン、テクストの文字色、文字枠と影である。なお、日本語を使う場合はフォント（通常はＭＳゴシック系 →フォントについては p. 20, 図 23 参照）を指定した上で SHIFT_JIS を選択すること。

8) スタイルとフォントの指定がおわったら、それぞれ [OK] ボタンをクリックして**図 8** の画面に戻る。ここで再度、画面右下の [OK] ボタンをクリックすると、先程の [Filters] というウィンドウが再度表示される（**図 9**）。この中に [**Textsub**] の項目が追加されている事を確認したら画面右上の [OK] ボタンをクリック。

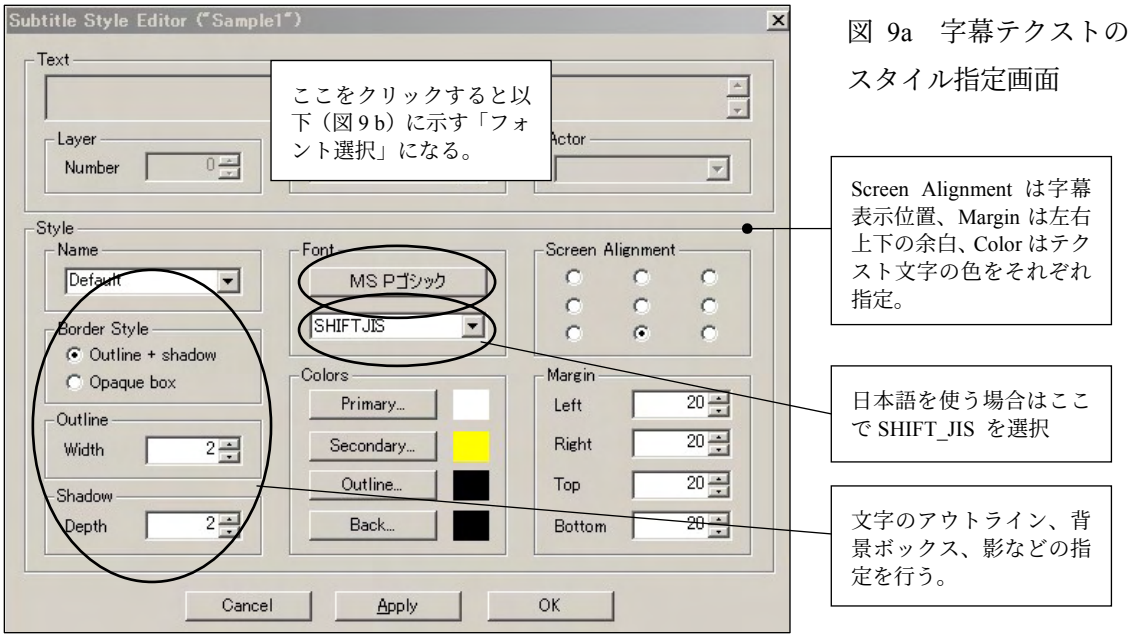


図 9a　字幕テクストのスタイル指定画面

図 9b　字幕テクストのフォント指定画面

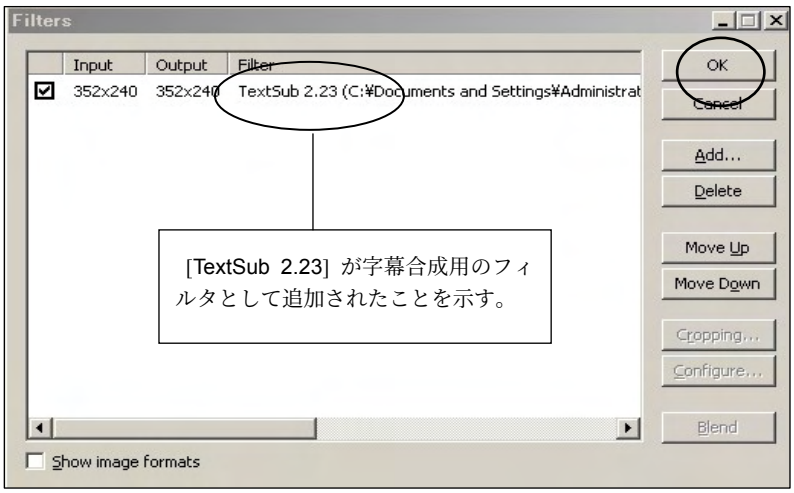


図 9　Textsub フィルタ追加の確認

9) 以上で字幕合成用の Textsub フィルタの読み込みが終了。次に映像にきちんと字幕が載っているかを確認。VirtualDab の画面には前記手順 2 で読み込んだビデオが表示されているので、ここから Output 再生ボタンを押して、出力画面に字幕が付いていることを確認（**図 10**）。

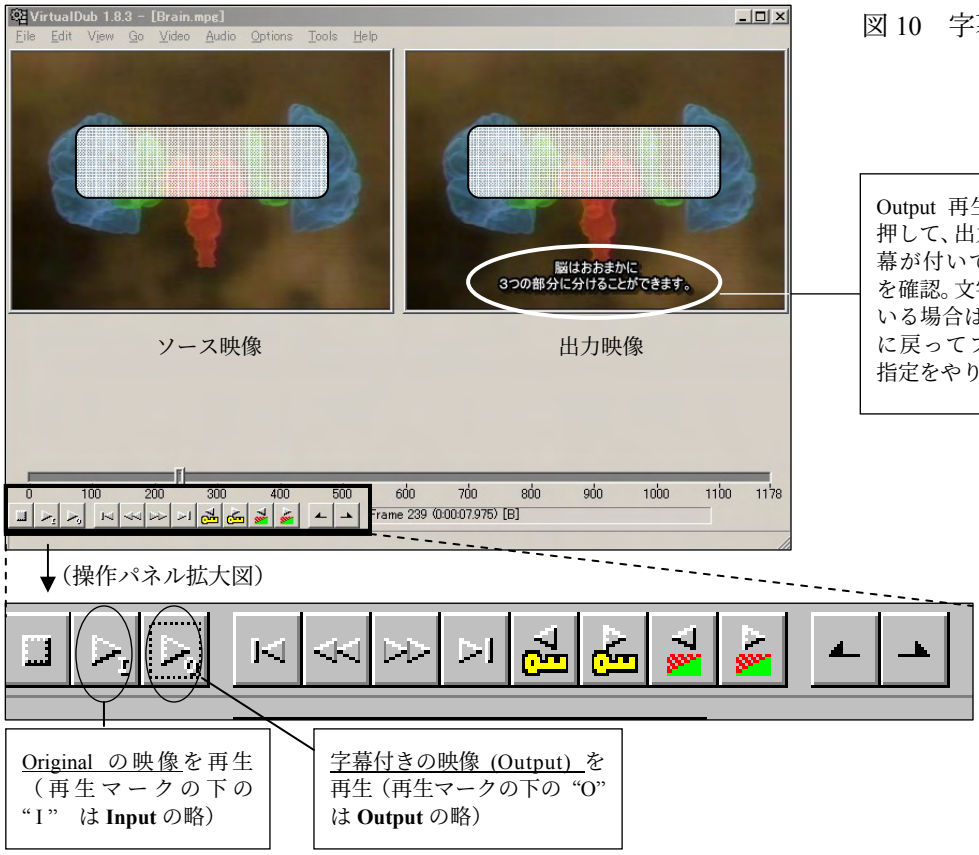


図 10　字幕表示の確認

Output 再生ボタンを押して、出力画面に字幕が付いていることを確認。文字化けしている場合は前記図 9a に戻ってフォントの指定をやり直す。

なお、図 10 の画面で左側に入力画面、右側に出力画面が出ていない場合はメニューから [View] を選択し、[Input video pane] と [Output video pane] にチェックマークが入っているかどうかを確認。入っていなければチェックを入れる。

10) 次に、メニューから [Video] → [Compression] を選択(**図 11**)。この際、その下の欄にある [Full Processing Mode] （完全処理モード）という項目にチェックが入っていることを確認。チェックがなければチェックを入れておく。

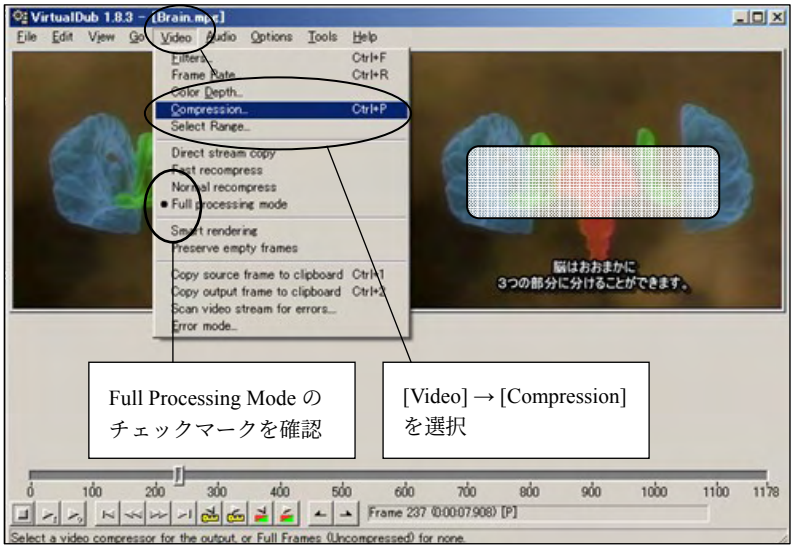


図 11　Compression モードの選択

11) 続いて [Select video compression] （映像圧縮方法の選択）という画面（**図 12**）になり、選択可能な Codec の一覧表が表示される。この中から最適と思われる Codec を選択して[OK] ボタンを押す (注：表示される Codec の内容はそれぞれの PC 環境により異なる。また、どの Codec を使うのが最適かは一概には言えないので、実際にいろいろ試してみていただきたい。ここではとりあえず Microsoft MPEG-4 Video Codec v3 [9] を選択しておく)。

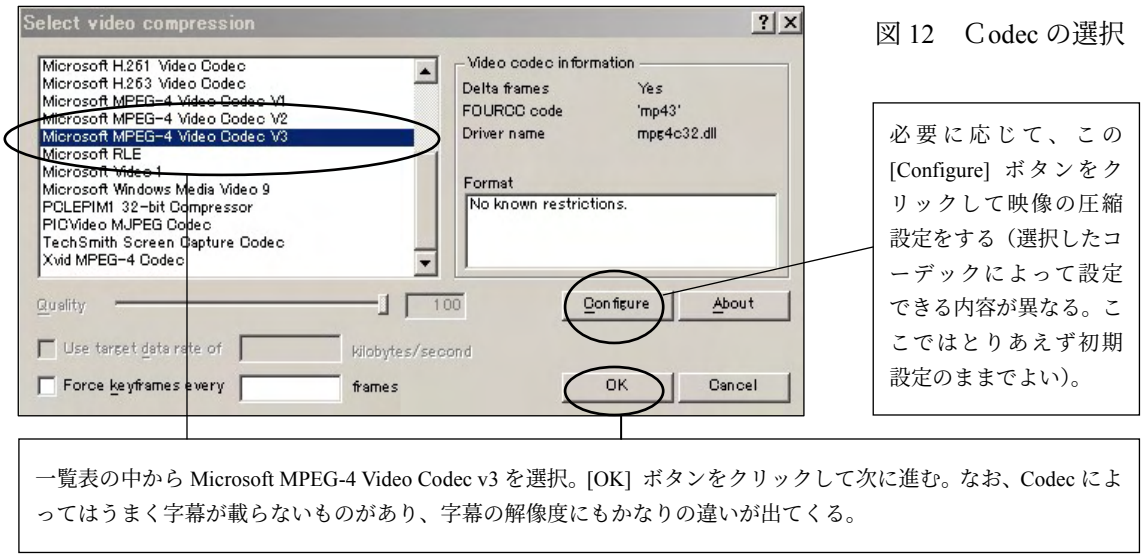


図 12　Codec の選択

12) 最後にメニューから [File] → [**Save as AVI**]（日本語パッチを当てている場合は「名前をつけて保存」という表示になる）を選択。保存場所を指定するウィンドウが開くので、ここで保存場所と保存ファイル名を指定して [保存] をクリックすると、映像と字幕をミックスした映像ファイルの作成が始まり、**図 13** のような作業画面が表示される。

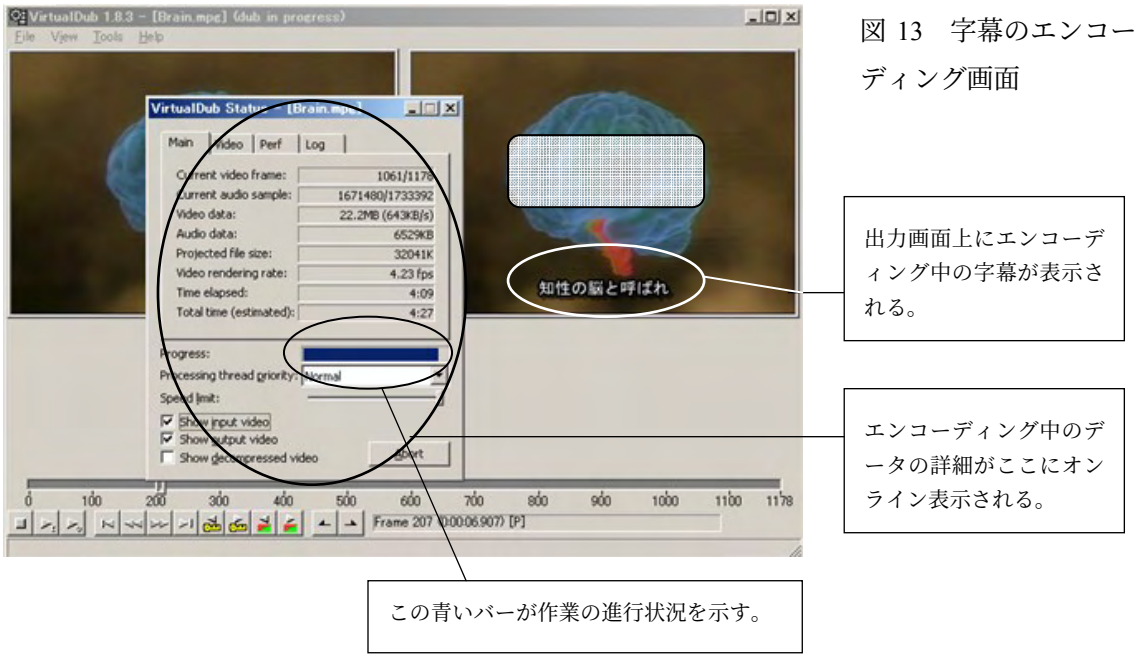


図 13　字幕のエンコーディング画面

13) エンコーディングが完了したら（完了のメッセージはとくに表示されず、作業進行画面が自動的に閉じる）、メニューから [File]→ [Quit] を選択して VirtualDab を終了。
14) 作成されたハードサブ動画は前記手順 12 で指定した場所に保存されているので、これを確認。字幕の品質に問題がある場合は、前記手順 11（**図 12**）に戻り、Codec を変えた上で再度、実行する（あるいは、次項の解説を参照して映像の出力サイズを変更する）。

1.5　映像サイズと字幕の品質について

一般に、ハードサブ字幕の品質は使用するコーデック、圧縮設定（とくにビットレートと映像のサイズ）、ハードウェア・アクセラレーション設定（染谷 2008a, p. 32 参照）、および使用するパソコンの CPU 性能等に左右されるが、基本的には指定した映像のサイズ（解像度）に合わせて字幕データがエンコードされるため、再生する際に映像を拡大表示させると字幕の質が大きく低下する。これを避けるためには最初から再生時のサイズに合わせた大きさにして字幕をエンコードする必要がある。

前出の**図 9** (p. 8) で見たとおり、本稿の解説に使用したソース映像は Input/Output ともに解像度 352×240 となっている。これは、ビデオ CD [NTSC] 規格の既定値であるが、字幕を表示させるにはやや小さすぎる。したがって、これを 640×480 程度のサイズに変更した上でエンコーディングするのが好ましい。VirtualDub には映像サイズを変更するためのフィルタがあらかじめ用意されており、これを追加することで、サイズの変更をしながら同時に字幕をエンコードすることができる。以下にその手順を示す。

1) 前記、**図 9** (p. 8) の状態から [Add] ボタンをクリック。
2) フィルタ一覧の中から [resize] を選択して [OK] をクリック。

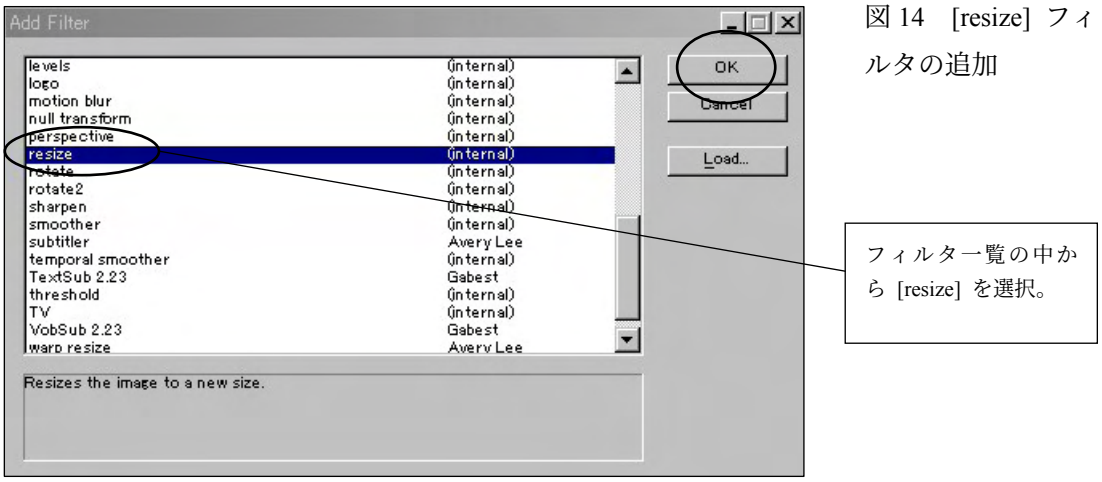


図 14　[resize] フィルタの追加

3) 以下のような画面が表示されるので、ここで必要な設定を行う。ここではとりあえず [Absolute (pixles)] をチェックして 640×480 と入力し、さらに [Aspect ratio] で [Compute height from ratio] をチェックした上で 4:3 と入力。他の項目は既定値のままでよい。

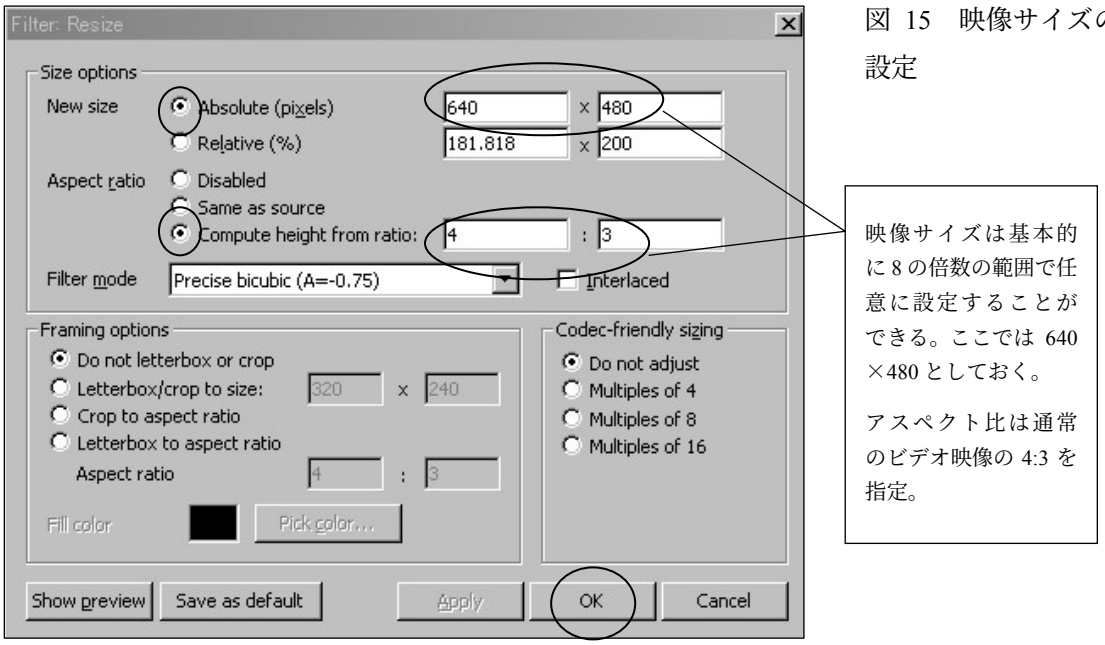


図 15　映像サイズの設定

[用語解説]

Letterbox: 映像フレームの上端と下端を黒く塗りつぶすこと。これによって、VHS テープ等からキャプチャしたフレームの端にしばしば現れる不要なライン（ノイズ）を取り除くことができる。

Crop: 映像フレームの余分なピクセルを切り落とすこと。

Aspect ratio: 映像の横縦サイズの比率。**注 12** の解説参照。

Codec-friendly sizing: コーデックによって画像のサイズが 4, 8, 16 のいずれかの倍数でないと適切に処理されないことがある。わからない場合は Do not adjust を選択しておく。

4)　設定後、[OK] をクリックすると、**図 16** のような画面になり、TextSub のほかに resize フィルタが追加され、映像サイズが 352×240 から 640×480 に変更されているのが確認できる。

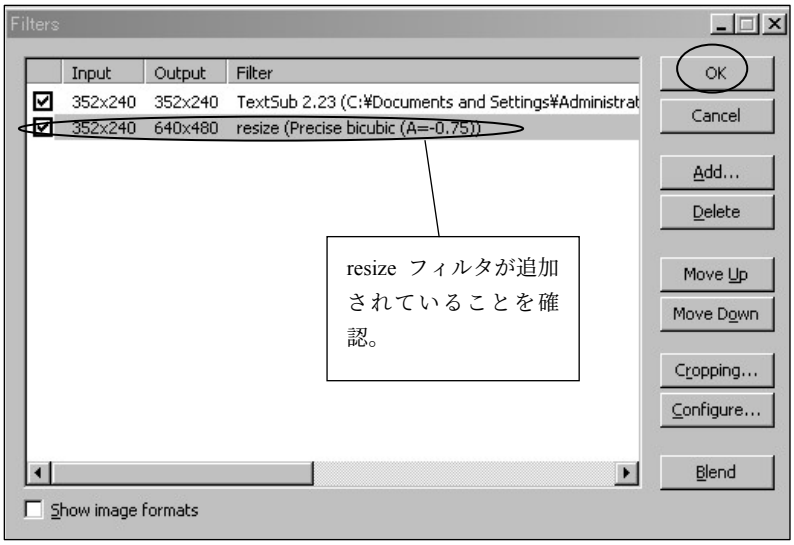


図 16　フィルタの読み込み確認画面

5)　確認後、[OK] ボタンをクリックしてパネルを閉じ、前出の**図 10** (p. 9) に戻る。ただし、今回は出力側の画像サイズを変更しているので、実際の表示は以下（**図 17**）のようになる。字幕がきちんと入っているかどうかは、操作パネルの Output ボタンをクリックすることで確認することができる。

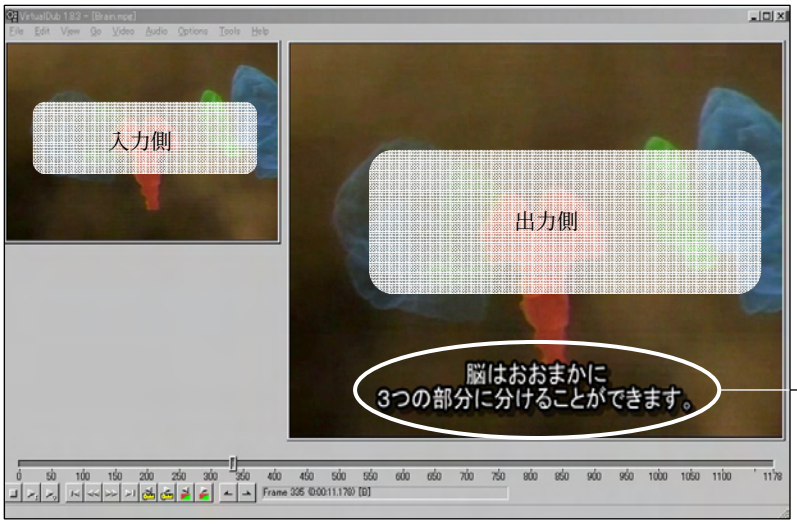


図 17　出力サイズ変更後のエンコード画面

映像サイズの拡大に伴って、字幕もより鮮明に出力される。

6)　字幕が載っていることを確認したら、あとは前述の手順 10 (p. 9) に戻り、順次作業を進めれば解像度 640×480 に変更されたハードサブ映像が作成される。

2. DivXLand Media Subtitler を使った「ハードサブ」動画の作成

DivXLand Media Subtitler（以下、Media Subtitler）は、フリーウェアの字幕エディタで、2007年1月にリリースされた最新版（ver 2.0.5）では、通常の字幕編集のほかに、ハードサブの作成（動画への字幕データの埋め込み）や動画からの音声の抽出等の機能がサポートされている。なお、ハードサブの作成は VirtualDab の機能を使って行うため、Media Subtitler を使うに当たってはあらかじめ VirtualDab をインストールしておく必要がある。

2.1 使用するアプリケーション

1) DivXLand Media Subtitler 2.0.5

ダウンロード先： http://www.divxland.org/subtitler.php

コメント：本来の「字幕エディタ」としても使い勝手がよく、かつ、音声波形を表示しながら字幕を編集することができるため、動画のみを見ながら作業する場合と比べてより厳密なタイミングの指定が可能 。ハードサブの作成は VirtualDab を利用した追加機能としてサポーされているものであるが、VirtualDab から直接ハードサブを作成するよりも操作が簡単で、同じ条件であればいくらか質の高い字幕を得ることができる。

対応動画形式：wmv, asf, mpg, (mpeg-1, 2), mp4, avi, divx, xvid, mkv, ogm 対応（※ただし、ハードサブの作成に関しては入力ファイルは avi と mpeg-1 形式、出力は avi 形式のみ）。

字幕ファイル形式： smi, srt, sub, ssa 等、およそ 30 種類の字幕形式に対応（※ただし、ハードサブを出力する際は ssa [= SubStation Alpha] 形式に変換してからエンコードする）。

2) VirtualDub 1.8.3 (32-bit version)

→セクション 1.1 (p.3) 参照（以下、VirtualDub がすでにインストールされているものとする）。

コメント：字幕の焼き付けは VirtualDab を使って行うが、VirtualDab そのものを別途立ち上げる必要はなく、表面的には Media Subtitler 内部のみでハードサブの作成ができる。

3) Subtitler.vdf (ハードサブ用のプラグイン)

ダウンロード先： http://www.virtualdub.org/virtualdub_filters.html

コメント： subtitler.vdf は Media Subtitler 用の VirtualDab Plugin。Media Subtitler をダウンロードした後、ファイルを解凍すると [subtitler-2_4] というフォルダができるので、この中にある subtitler.vdf を VirtualDab フォルダ (c:¥Program File¥VirtualDub-1.8.3) の中にある [plugins] フォルダに入れておく（VirtualDab のインストールについてはセクション 1.1 で既述）。なお、このプラグインは ssa 形式の字幕ファイルを処理するためのものであるが、ssa 形式への変換および保存は Media Subtitler 上で直接行うことができる。

4) Xvid codec 1.1.3 Stable

ダウンロード先： http://www.divx-digest.com/software/xvid.html

コメント：Subtitler がハードサブの作成に使用する MPEG-4 ビデオコーデック。ダウンロードしたファイルを開くとインストーラーが起動するので、指示に従ってインストールしておけば、Media Subtitler を起動後、必要に応じて自動的に読み込まれる。なお、Xvid はオープンソース（ソースコードが公開されていて、基本的にフリーウェア）でありな

がら、非常に高い画質で映像を圧縮することができるコーデックとして定評がある。似た名前のコーデックに DivX があるが、こちらは有料（Xvid は DivX を開発したスタッフが DivX の有料化に反対して独自開発したもので、画質は DivX とほぼ同等）。

5) **BeSweet (Audio Encoder) v.1.5b31**

ダウンロード先： http://dspguru.doom9.org/

コメント：映像から音声を抽出し、波形表示させる際に必要なオーディオコーデック (Version 1.5 のベータ版 v1.5b31 をダウンロード)。音声は wav 形式で保存されるので、一般的な音声編集ソフトや（音声波形表示に対応している）字幕エディタで再利用することができる。

2.2　Media Subtitler のダウンロードとインストール

1) 前記のダウンロード先から Media Subtitler をデスクトップ上にダウンロード。ダウンロードが終わると右のようなアイコンがデスクトップ上に表示されるので、これをダブルクリックして実行。

2) 画面上の指示に従ってインストールを進める（特に問題がない限り各画面で Next をクリックしていけばよい）。

3) インストールが終了すると、デスクトップ上に右のようなショートカットアイコンが配置されるとともに、c ドライブの[Program Files] フォルダの中に [DivXLand] というフォルダが作成される。プログラムはその下の [Media Subtitler] というフォルダの中に解凍されているので、これを確認。

2.3　Subtitler.vdf のダウンロードとインストール

1) 前記のダウンロード先にアクセスすると、VirtualDab フィルタのリストが表示されるので、ここから **subtitler-2_4.zip** をデスクトップ上にダウンロード。ダウンロードしたファイルをダブルクリックして解凍すると、[subtitler-2_4] という名称のフォルダがデスクトップ上に表示されるので、その中に **Subtitler.vdf** というファイルが含まれていることを確認。

2) 次に [subtitler-2_4] フォルダを、先ほどインストールした [Media Subtitler] フォルダ（c:¥Program Files¥DivXLand¥Media Subtitler）の中にそのまま移動（**図 18**）。

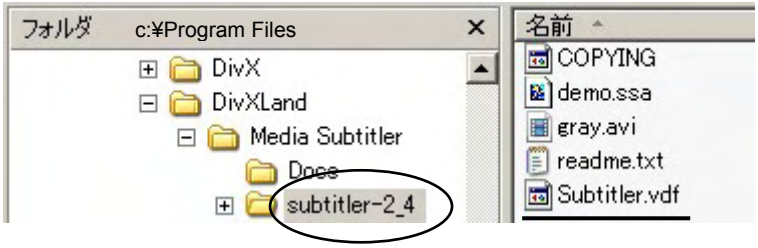


図 18　c:¥Program Files > DivXLand > Media Subtitler の中に[subtitler-2_4] フォルダを移動

3) [subtitler-2_4] フォルダの中にある **Subtitler.vdf** を VirtualDub フォルダ (c¥Program Files ¥DivXLand ¥VirtualDub-1.8.3) の中の [plugins] フォルダにコピーして貼り付け（**図 19**）。

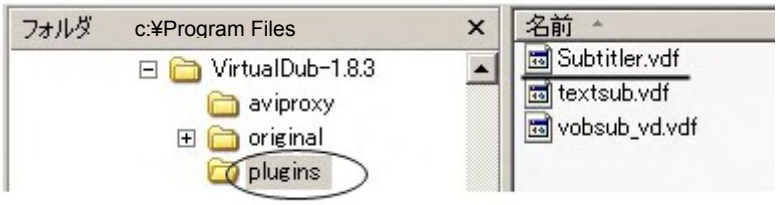


図 19 c:¥Program Files > VirtualDub-1.8.3 > plugins の中に Subtitler.vdf を移動 (copy & paste)

2.4 Xvid codec のダウンロードとインストール

1) 前記のダウンロード先にアクセスすると、Xvid codec のダウンロード用インターフェイスが表示されるので、1.1.3 Stable というバージョンであることを確認して [Download] という表示をクリックしてデスクトップ上にダウンロード。
2) ダウンロード後、デスクトップ上に表示される [Xvid-1.1.3-27042008] というアイコをダブルクリックして実行。
3) セットアップウイザードの指示にしたがって、すべて初期設定のまま [Next] をクリック。最後に完了のメッセージが出たら [Finish] をクリックしてインストレーション終了（最後にに c ドライブの [Program Files] の中に [Xvid] というフォルダがあることを確認）。

2.5 BeSweet のダウンロードとインストール

1) 前記のダウンロード先にアクセスし、タイトルバナーの下にあるメニューから [beta] をクリック。ダウンロード可能なバージョンの一覧が表示されるので、一番上にある[BeSweet v.1.5b31] をクリックしてデスクトップ上にダウンロード。
2) ダウンロード後、デスクトップ上に表示される [BeSweet v.1.5b31.zip] というアイコンをダブルクリックしてファイルを解凍。
3) 解凍が終わるとデスクトップ上に同名のフォルダが作成されるので、これを c ドライブの [Program Files] の中に手動で移動 (drag & drop) して作業終了（**Media Subtitle**r との関連付けは後で行うので、ここではこれで作業を終了してよい）。

2.6 Media Subtitler の起動およびハードサブ作成のための準備作業

デスクトップ上のショートカットアイコンをクリックして（または Windows の [スタート] から [プログラム]→[DivXLand]→[Media Subtitler] を選択して）Media Subtitler を起動。プログラムが起動すると、**図 20** のような初期画面が表示される。以下、この画面に映像ファイルと字幕ファイルを読み込み、ハードサブの作成を行うための手順を示す。なお、この作業に先立って DirectVobsub をオフにしておくこと（**注 6** 参照）。

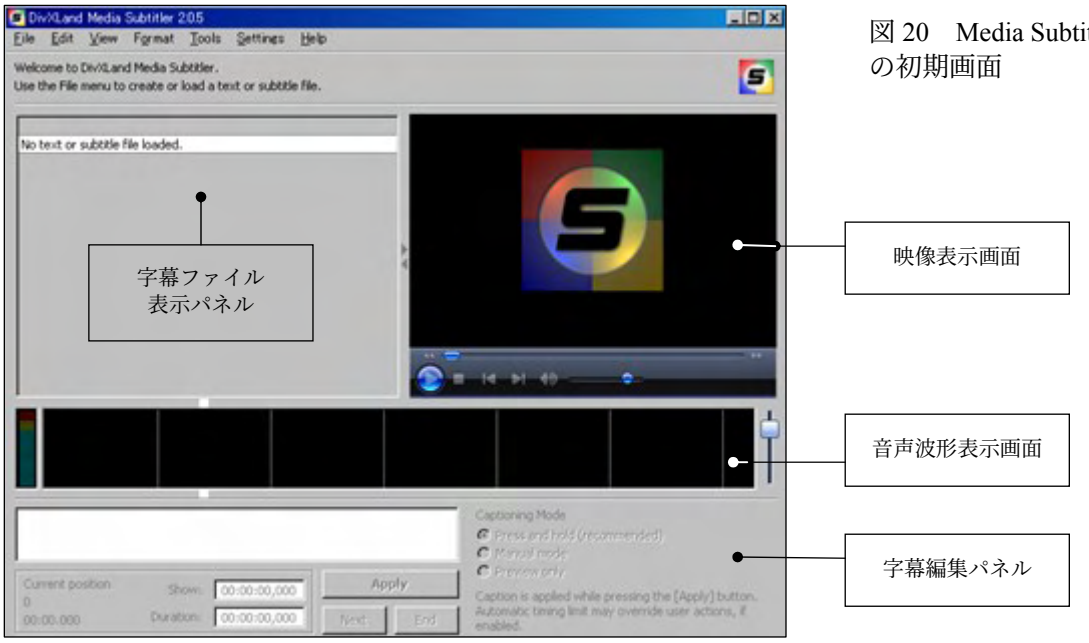


図 20　Media Subtitler の初期画面

1) まず、**図 20** の画面から、[File]→[Open Video] を選択し、映像ファイルを読み込む。対応しているファイル形式については p. 14 参照。
2) 次に、[File]→[Open Text or Subtitle] を選択し、字幕ファイルを読み込む（Media Subtitler を字幕エディタとして使う場合は、字幕データのみを書き出したテキストファイルを読み込み、これに各行ごとにタイミング＝開始時間と終了時間＝を加えていくが、ここではすでに時間データが付与された編集済みの字幕ファイルを読み込むものと仮定する。字幕フォーマットの変換については**注 8** 参照）。
3) 映像ファイルと字幕ファイルの読み込みが終わると、**図 21** のような画面になる（ただし、この段階では音声波形は表示されない）。
4) **図 21** で、字幕と映像を同期させて「プレビュー」するためには、画面右下にある [Captioning Mode] のところで、一番下の [Preview Only] にチェックを入れてプレーヤを再生。これで、左側ペインに表示されている各字幕行が映像画面上に順次同期表示される。プレビュー時の字幕の位置設定およびフォント設定の方法は以下のとおり。

<u>プレビュー時の字幕の位置設定</u>

メニューから[Settings]→[General Settings]→[Subtitle Preview Settings] を選択。[Subtitle preview placement] のスライドバーを 90-100%の範囲に設定（数字を少なくすると上に移動）。

<u>プレビュー用のフォント設定</u>

メニューから[Settings]→[General Settings]→[Subtitle Preview Settings] を選択。[Font] ボタンをクリックしてフォントのサイズ等を指定（プレビュー用なので 10～12 ポイント程度のサイズを指定）。さらに [Color] ボタンをクリックしてフォントの色を指定。

図 21 Media Subtitle のプレビュー画面

プレビューモードにすると、各字幕行が同期して順次画面上に表示される。

音声波形の始点と終点はマウスでドラッグすることで、任意に指定・再指定が可能、

字幕のプレビュー時にはこの "Preview Only" にチェックマークを入れる。

編集モードにすると、ここで各字幕行の編集および同期時間の (再) 設定を行うことができる。

5) プレビューの後、字幕を部分的に再編集したい場合には [Captioning Mode] を [Press and Hold (Auto)] か [Manual] にした上で、必要な編集作業を行う。字幕の内容や同期タイミングに問題がなければ、そのまま字幕を映像に焼き込む作業に移ることができる。

6) ただし、その前に、今後の作業に必要な外部フィルタ読み込み用の設定をあらかじめ行っておく（この作業は一度だけ行っておけば、次からは設定済みの画面が表示される）。

<u>外部フィルタの設定と音声データの抽出</u>

1. まず、**図 21** の状態（ただし音声波形表示はなし）から、メニューの[Tools]→[Audio extraction wizard]（音声抽出ウィザード）を選択。
2. Target format の選択画面で、[WAV (PCM)] を選択し、[Load extraction file on the audio bar]（音声データの読み込み）にチェックマークを加えて [Next] をクリック。
3. 次の Sample rate と Bitrate は初期値（"Use original"）のまま [Next] をクリック。
4. 外部プログラム (external programs) の設定画面（**図 22**）になるので、ここで **VirtualDub** フォルダと **BeSweet** フォルダの所在場所をそれぞれ指定。これまでの指示に従って作業を行っている場合はそれぞれ以下のような指定先となる。

   VirtualDub's folder: C:¥Program Files¥VirtualDub-1.8.3¥

   BeSweet Encoder's folder: C:¥Program Files¥BeSweetv1.5b31¥

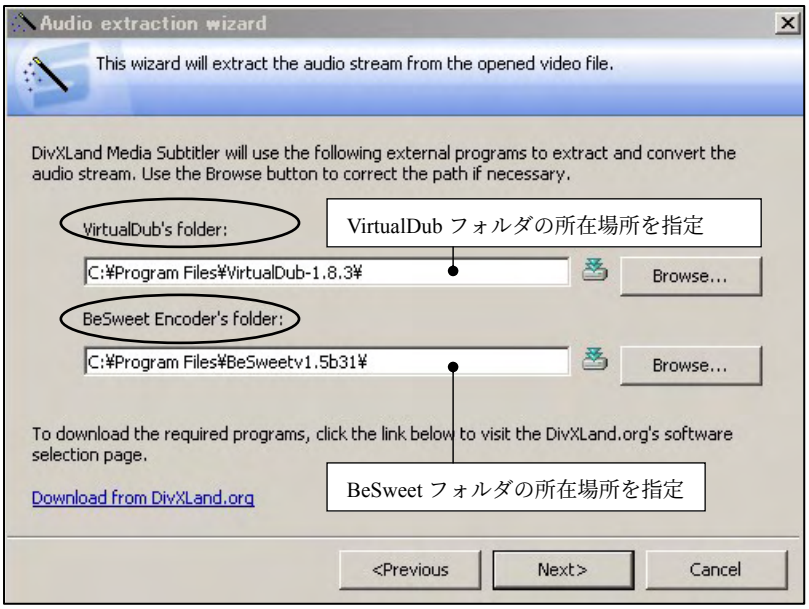


図 22　外部フィルタの所在場所の指定 ※

※ハードサブ作成用と WAVE 音声抽出用の外部フィルタの指定

5. 指定後、[Next] をクリックすると保存フォルダの指定画面になるので、[Save as] をクリックし、保存フォルダと保存ファイル名を指定して [Next] をクリック。
6. Audio extraction completed! という画面が出たら[Finish] をクリックして作業終了。
7. 終了と同時に自動的に**図 21** の画面に戻り、音声波形表示画面上に先ほど抽出した音声の波形が表示される（ただし、メニューの [View]→[Audio bar] にチェックが入っていないと音声波形表示画面そのものが表示されない）。波形が表示されていない場合にはプレーヤを再生させると波形が表示されるようになる。なお、作成した音声ファイルは指定したフォルダに保存され、プログラム終了後もそのまま残される。

以上で、音声データの抽出と波形表示が可能になるとともに、VirtualDub の機能を使ってハードサブを作成するための準備ができたことになる。ここで再度、**図 21** の画面に戻り、ここからハードサブの作成に入る。

ハードサブの作成

1) **図 21** の状態（＝字幕が動画にきちんと同期しており、これ以上の編集が必要ない状態）で、メニューから[Settings]→[Format Settings] を選択。
2) [Format Settings] の画面で左側ペインに表示されている字幕フォーマットのうちから **Sub Station Alpha [SSA]** 形式を選択した上で、[Font], [Color], [Style], [Alignment], [Margin] 等を順次指定。このうち [Alignment] では字幕の表示位置を設定する（通常は center と bottom を選択）。[Font] はゴシック系のフォント（和文の場合は MS ゴシック、MS P ゴシック、HGP 創英角ゴシック UB、HG 丸ゴシック M-PRO など。英文の場合は Arial, Verdana, Tahoma など＝次ページ**図 23** のサンプル参照）を指定。フォントサイズは画面サイズに対して可能かつ適切な範囲でできるだけ大きいほうがよい。日本語を使う場合はフォントの [文字セッ

ト] を「日本語」にしておくこと。なお、フォント設定等を変更した場合は、ここでいったんオリジナルの SSA ファイルを上書き保存（[File]→[Save]）してからエンコーディングすること（保存しないと、ここでの指定や変更内容がハードサブに反映されない）。

図 23　ゴシック系のフォントサンプル（12 ポイント）

これは "MS **ゴシック**" **のサンプルです。（仮名漢字）**

これは "MS P ゴシック" のサンプルです。（仮名漢字）

これは "HGP **創英角ゴシック** UB" **のサンプルです。（仮名漢字）**

これは "HG 丸ゴシック M-PRO" のサンプルです。（仮名漢字）

This is how "Arial" looks like, and this is **Bold Face**

This is how "Verdana" looks like, and this is **Bold Face**.

This is how "Tahoma" looks like, and this is **Bold Face**.

3) 書式の設定が終わったら [OK] をクリックして [Format Settings] の画面を終了。
4) 次に、メニューから[File]→[Embed subtitles] を選択すると**図 24** のような出力ファイルの設定画面が表示される。この画面での設定項目および標準設定値は以下のとおり。

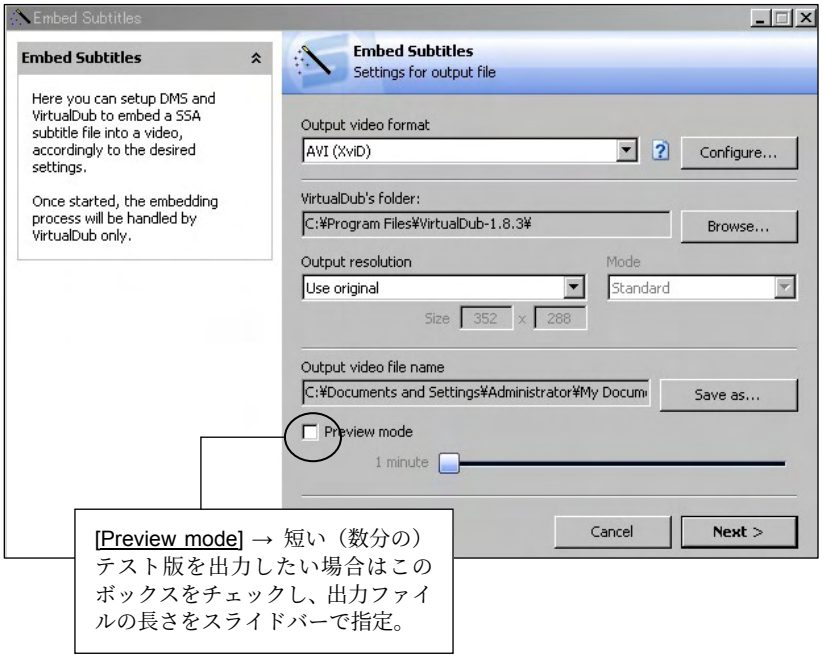


図 24　出力ファイルの設定画面

[Output video format]
→ 既定の AVI (XviD) 使用。

[Configure] →ここの設定はすべて初期値のままでよい。

[VirtulDub's Folder]
→ 設定済み。

[Output resolution] → 出力ファイルの解像度（画面サイズ）を指定。とくに変更する必要がなければ既定値（"Use original"）のままでよい。

5) 必要な設定が終わったら、[Output video name] の項目の [Save as] をクリックし、保存場所と保存ファイル名を指定した上で [保存] をクリック。

6) **図 21** の画面に戻るので、ここで [Next] をクリックし、さらに次の画面で [OK] をクリックするとエンコーディングが開始され、次のようなウインドウ（**図 25**）が表示される。

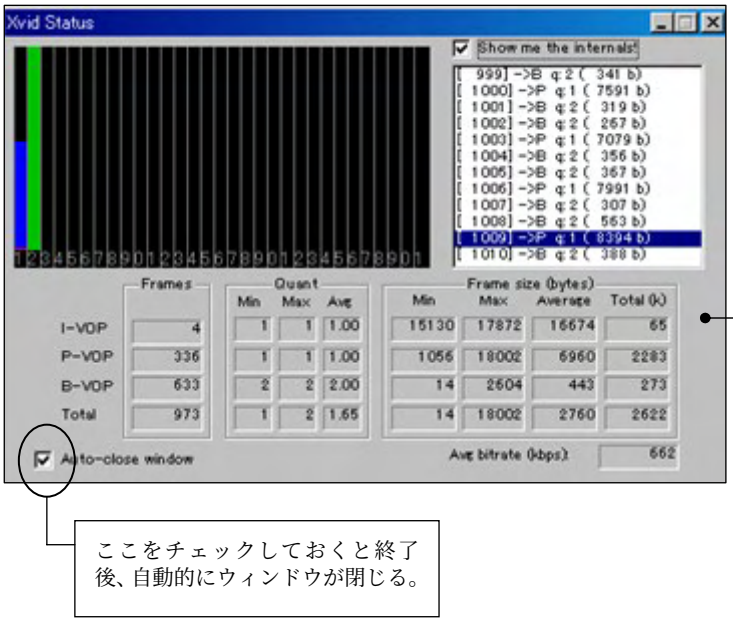


図 25　エンコーディングの進行状況ウィンドウ (Xvid Status)

パネルの表示内容は、図 19 での XviD Configuration の設定内容に応じて変化するが、通常はまったく気にする必要はなく、単に作業の進行状況を示すインジケータとしてみておけばよい。

ここをチェックしておくと終了後、自動的にウィンドウが閉じる。

7) エンコーディングが終了すると Media Subtitler は自動的に初期画面（**図 20**）に戻り、次の新規作業のためのスタンバイ状態になる。字幕が焼きこまれた「ハードサブ」動画は前記 5 で指定した保存場所に保存されているので、これを再生してうまくできているかどうかをチェック。コーデックは Xvid に固定されているため、最終的な映像の質はもっぱらオリジナル映像の質やパソコンの CPU 性能に左右される。ただし、字幕の焼き込み自体は VirtualDub の機能を使って行うため、第 1 章で述べた VirtualDub によるハードサブ作成時の字幕品質を大きく上回ることはない。

3. TMPG Encoder を使った「ハードサブ」動画の作成

TMPG Encoder はその名のとおり mpeg エンコーダで、たいへん質の高い mpeg 画像が得られるエンコーダとして高い評価を得ている。無償版と有料版があり、無償版では mpeg-1 へのエンコードが無制限で可能になっているが、mpeg-2 は使用期間が 2 週間に限定されている。有料版は TMPGEnco Plus 2.5 という名称で製品化されており（税込価格 2,940 円)、mpeg-1 のほかに mpeg-2 も恒常的に使いたいという場合は製品版を購入する必要がある。いずれもソース映像として avi, mpeg, mov, asf, wmv 等の動画形式をサポートしている（出力は mpeg のみ)。なお、このうち wmv は公式な対応形式一覧には含まれていないが、問題なく読み込むことができる。[10)]

TMPG Encoder を使った「ハードサブ」動画の作成は、開発者が本来想定していたものではなく、いわば非公式の裏技であるが、パソコンに DirectVobSub がインストールされていれば、その機能を利用してハードサブ動画を出力することができる。以下、その具体的な方法を解説する。

3.1 使用するアプリケーション

1) **TMPGEnc version 2.525**（無償版）

   ダウンロード先： http://www.tmpgenc.net/ja/j_download.html

2) **DirectVobSub**

   DirectVobSub については染谷 (2008a, pp. 21-28) 参照。以下、本稿では DirectVobSub がインストールされ、自動起動可能な状態に設定されているものとする。

3) **対応する動画および字幕ファイルの形式**

   動画：wmv 形式

   字幕ファイル： SAMI 形式の字幕ファイル（ただし、wmv 形式の動画ファイルと同じ名称（拡張子は.smi）で同一フォルダ内に保存されていること)。

   コメント：TMPG Encoder で wmv 形式の動画を読み込んだ後、mpg 形式（mpeg-1 または mpeg-2）での保存を指定すると DirectVobSub が自動的に起動し、画面上に字幕データを取り込みながらエンコーディングすることができる。ただし、映像ソースが初めから mpg または avi 形式になっている場合は DirectVobSub は起動しない [11)] 。なお、字幕の書式は基本的には DirectVobSub での書式設定（図 30, p. 27 および染谷 2008a, p. 24 参照）が適応されるが、SAMI ファイルの字幕本文中での HTML タグによる書式指定はそのまま反映される。

3.2 TMPG Encoder のインストールとプログラムの起動

1) まず、上記のダウンロード先から TMPG Encoder（無償版）をデスクトップ上にダウンロード。
2) ダウンロードが終わると、デスクトップ上に [TMPGEnco-2.525.64.184-JA.zip] という圧縮ファイルのアイコンができているので、これをダブルクリックして解凍。
3) 解凍が終わると、デスクトップ上に [TMPGEnco-2.525.64.184-JA] というフォルダが作成されるので、これを c ドライブの [Program Files] 内に移動。
4) [TMPGEnco-2.525.64.184-JA] フォルダの中にある、右のようなアイコンをダブルクリックしてプログラムを起動。

3.3　「プロジェクトウィザード」を使ったエンコーディング ― ハードサブ動画の出力

5) プログラムが起動すると、初期画面とともに「プロジェクトウィザード」画面（**図 26**）が開くので、ここで、これから作成する映像のフォーマットを選択する（出力ファイルのフォーマットを手動で細かく設定することもできるが、今回はウィザードを使用する）。既定の選択肢は①ビデオ CD [NTSC] （解像度 352×240, フレームレート 29.97 fps の MPEG-1 ファイル）、②スーパービデオ CD [NTSC] （解像度 480×480, フレームレート 29.97 fps の MPEG-2 ファイル）、③DVD ビデオ [NTSC 4:3] （解像度 720×480 / 352×480, フレームレート 29.97 fps の MPEG-2 ファイル） または④DVD ビデオ [NTSC 16:9]（解像度 720×480, フレームレート 29.97 fps のワイド規格 MPEG-2 ファイル）のいずれか。今回はとりあえず①のビデオ CD [NTSC] 規格に準じた MPEG-1 ファイル（CBR 標準フォーマット）を選択。

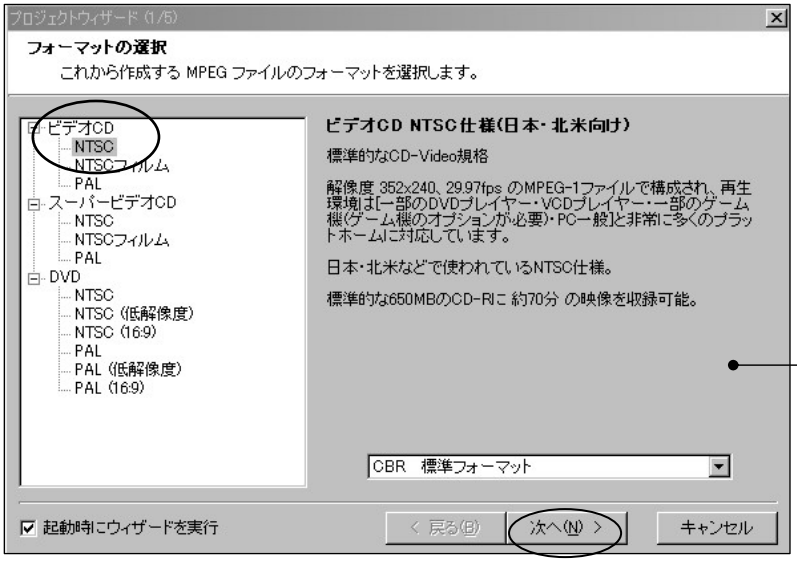


図 26　TMPG Encoder のプロジェクトウィザード画面

フォーマットの選択：①ビデオ CD [NTSC]　②スーパービデオ CD [NTSC] ③DVD ビデオ[NTSC 4:3] または④DVD ワイドビデオ [NTSC 16:9] のいずれか適当なものを選択して「次へ」をクリック。

6) 上記画面から「次へ」をクリックすると「ソースファイルの選択」画面（**図 27**）になるので、ここで「参照」をクリックして目的のフォルダに移動。「ファイルの種類」を「すべてのファイル」にして、字幕を付けたい **wmv** ファイルを指定する。

7) ソースファイルを指定したのち「次へ」をクリックすると「フィルタの設定」画面（**図 28a**）に移動するので、ここで「その他の設定」をクリックして、「.MPEG の詳細設定」画面（**図 28b**）に進み、既定の映像サイズを変更する。

注）前記の手順 5＝**図 26** で選択したビデオ CD [NTSC] 規格の MPEG-1 ファイルの既定サイズは 352×240 だが、このサイズだと字幕表示には小さすぎるので、ここではこれを 640×480 または 720×480 に変更した上でエンコーディングする。なお、サイズは縦横ともに 8 の倍数でなければならない。通例、映像サイズを大きくすると画質は落ちるが字幕は（小さな画面を拡大表示した場合に比べて）よりきれいに表示される。

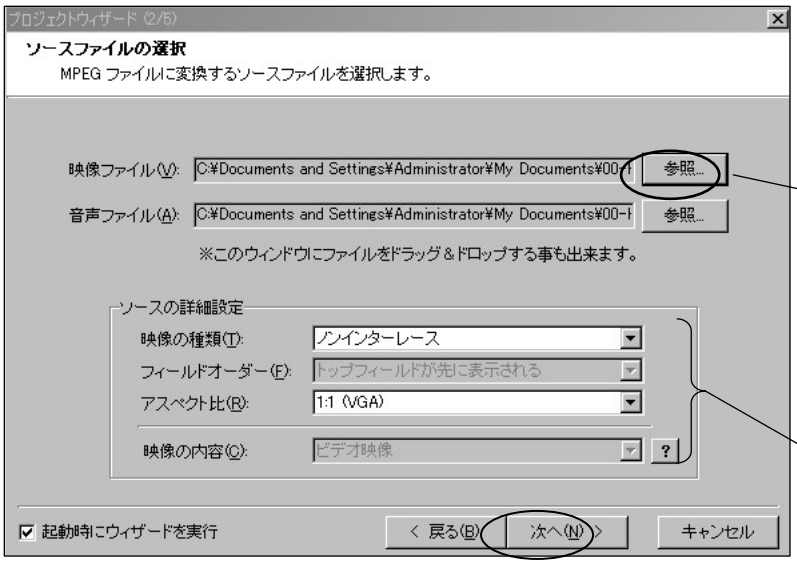


図 27　ソースファイル選択画面

ソース映像の指定：「参照」をクリックして目的のフォルダに移動。「ファイルの種類」を「すべてのファイル」にして字幕を付けたい wmv ファイルを指定する。

ソース映像の詳細：ソース映像のデータが表示される。ここは通常は変更の必要なし。

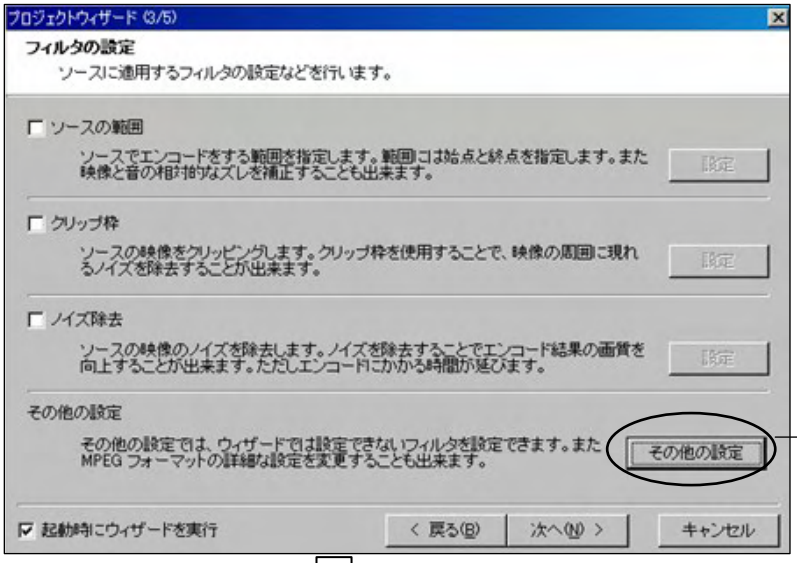


図 28a　フィルタの設定画面

既定の画面サイズその他を変更するには、ここをクリックして「MPEG の設定」画面に移行。

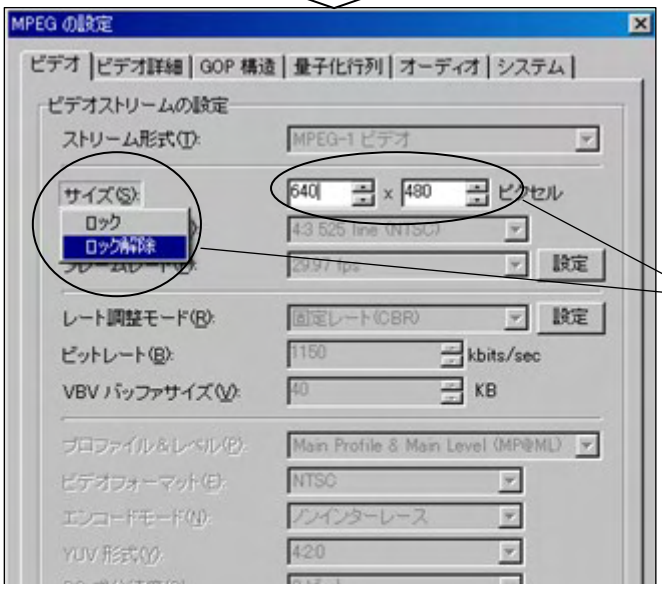


図 28b　MPEG の詳細設定画面 (画面サイズの変更)

各項目は既定の設定にロックされているが、項目名をクリックすると左図に示すように「ロック解除」という選択肢が表示され、ボックスが入力（または選択）可能な状態になるので、適宜、必要なデータを入力（または選択）する。ここでは画像サイズを変更しておく。

8)　上記画面で「OK」をクリックすると、**図 28a** の画面に戻るので、ここで「次へ」をクリックすると、「ビットレートの設定」という画面に移行する(画面イメージは省略)。ここでは TMPG Encoder がソース映像に合ったビットレートを自動的に設定してくれるので、原則として設定値を変更せずにそのまま「次へ」をクリックする（通例、ビットレートが高いほど画質はよくなるが、ビデオ CD の規格では 1150 bps（CBR 固定）が規格上の上限値となっているため、プログラムの既定値を変更せずにそのまま使用する）。

9)　前記の「ビットレートの設定」画面で「次へ」をクリックすると「プロジェクトウィザード」の最後の画面 (5/5) になるので、ここで出力ファイル名を指定した上で「直ちにエンコードを開始する」をチェック。最後に「OK」をクリックするとエンコードが開始され、以下のような画面（**図 29**）が表示される。

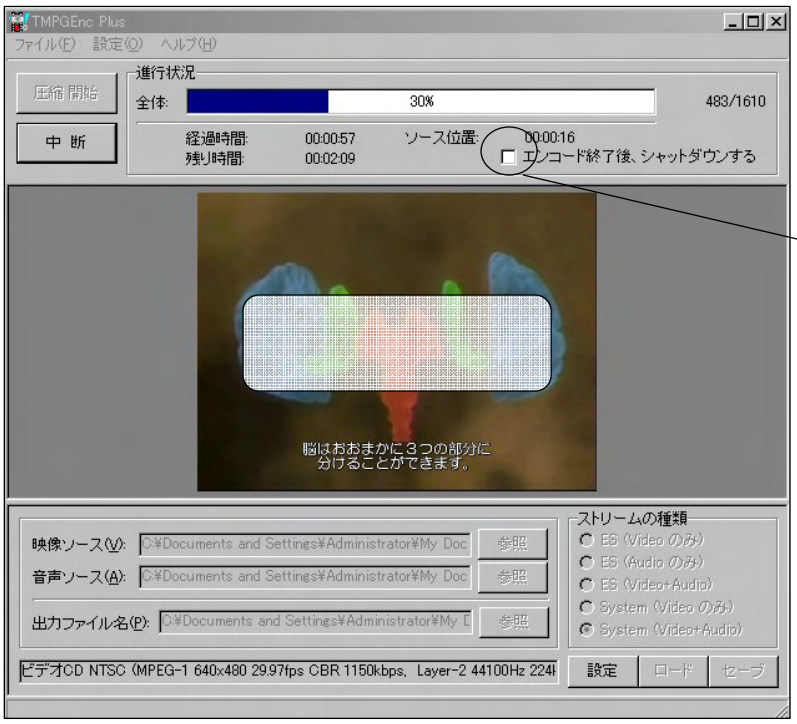
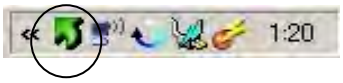


図 29　エンコード画面

大きなサイズの映像をエンコードする場合はこのチェックボックスを ON にして、エンコード終了後、自動的にシャットダウンさせる。

なお、エンコーディングの開始とともに DirectVobSub が自動的に起動するが、正しく起動している場合には Windows のタスクバー右側の通知領域に以下のような矢印のアイコンが表示される。これが表示されていない場合はエンコーディングをいったん中止し、DirectVobSub を適切に設定し直した後、前記 5) 以降の作業を繰り返す。



エンコードにかかる時間はソース映像のサイズや出力設定によるが、通例、かなりの時間を要する。したがって、大きなサイズの映像をエンコードする場合は、上記画面（**図 29**）の「進行状況」の枠内にある「エンコード終了後、シャットダウンする」のチェックボックスを ON にした上で、就寝中や外出中の時間を利用してエンコード作業をさせておくのがよい。

なお、エンコードを開始しても画面上に字幕が表示されない場合は、前述の DirectVobSub が適切にインストールされ、自動起動可能な状態になっているかどうかのほかに、ソース映像が wmv ファイルかどうか、および SAMI 形式の字幕ファイルがソース映像ファイルと同じ名称（拡張子は .smi）で同一フォルダ内に保存されているかどうかをチェックすること。**図 37b** (p. 32) に TMPG Encoder を使ったハードサブのサンプルイメージを示す。

3.4 MPEG-1 と MPEG-2 の違い、および映像のサイズ設定について

前記の「プロジェクトウィザード」画面（**図 26**）では、既定の選択肢としてビデオ CD、スーパービデオ CD、および DVD ビデオの 3 種類があったが、このうちビデオ CD は **MPEG-1**、スーパービデオ CD と DVD ビデオは **MPEG-2** の形式でそれぞれエンコードされる。**MPEG-1** は汎用性が高く、ほとんどのパソコン上で再生が可能だが、画質は MPEG-2 より劣る。これに対して、**MPEG-2** は高品質のエンコーディングができ、字幕もきれいに出すことができるが、MPEG-2 に対応したコーデックがインストールされていない PC 環境では再生できないことがある。

映像のサイズ（解像度）は原則として 8 の倍数という制限内で任意のものを指定することができるが、とくにこれといった目標サイズがない場合はソース映像と同じサイズにしておくのがよい。一般的な映像のサイズは、ビデオ CD が 352×240 ピクセル、DVD ビデオが 720×480（または 704×480）ピクセルで、VGA 規格では 640×480 ピクセルになる。一般に、映像サイズを大きくすると画質は落ちるが字幕はよりきれいにエンコードされる。

参考までに、以下にケース 1 からケース 3 まで 3 つの異なった条件で同じソース映像を TMPG Encoder を使ってハードサブにエンコードした場合の結果を示す。ソース映像は、ファイル形式＝wmv、ファイルサイズ＝3.1 MB、映像サイズ＝640×480、ビットレート＝CBR 531kbps で、アスペクト比 1:1 (VGA) [12] のノンインターレース映像 [13] である。

ケース 1

<u>エンコード条件</u>：Video-CD (NTSC), MPEG-1, 352×240, 29.9fps, CBR 1150kbps, 4:3 525 line (NTSC)

<u>結果</u>：字幕の品質は VirtualDab と同等。ただし、既定サイズ 352×240 では画面が小さすぎて字幕がきれいに出ない。ソース映像と同じ 640×480 にしてエンコードすると画面がやや縦長の感じになるが、720×480 にするとほぼ正しい横縦比率になり、字幕の品質も向上する。出力ファイルサイズ＝9.2 MB。

ケース 2

<u>エンコード条件</u>：Super Video-CD (NTSC), MPEG-2, 480×480, 29.9fps, CBR 2520kbps, 4:3 Display

<u>結果</u>：ソース映像が 640×480 なので、既定値の 480×480 でエンコードすると縦にゆがんだ映像になる。ただし、出力サイズを 640×480 に変更してエンコードすると映像・字幕ともきれいに出る。MPEG-1 よりも品質がよい。出力ファイルサイズ＝18.2 MB。

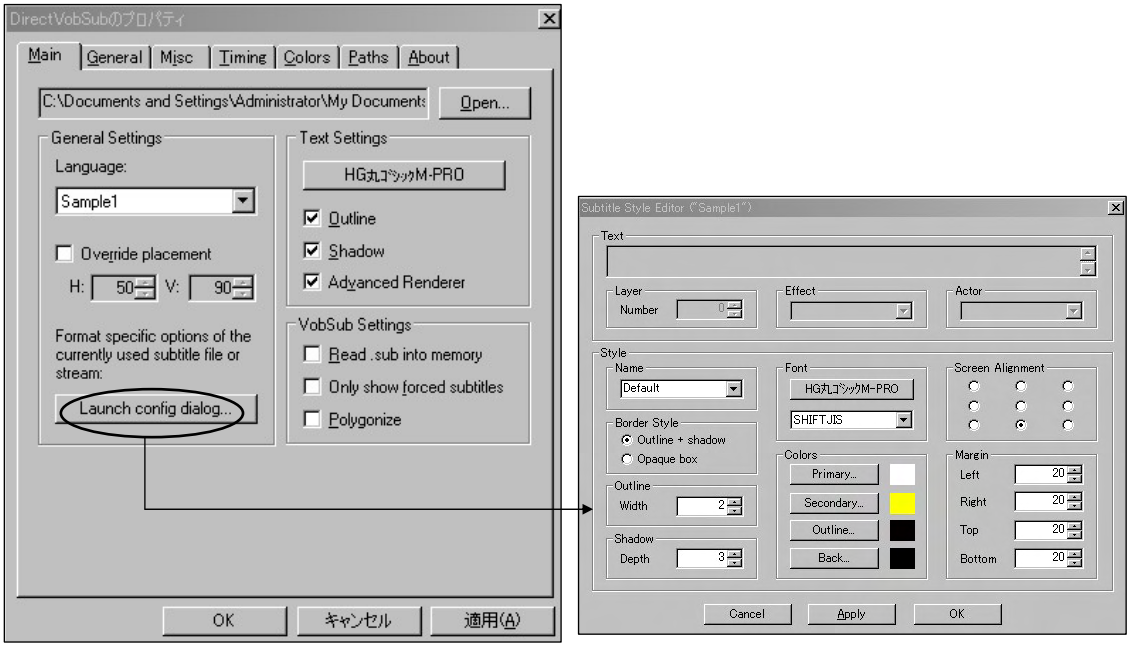
ケース 3

エンコード条件：DVD (NTSC), MPEG-2, 720×480, 29.9fps, CBR 7992kbps, 4:3 Display

結果：ソース映像が 640×480 なので、既定値の 720×480 でエンコードするとやや横に広がった映像になるが、字幕はきれいに出る。ソース映像と同じ 640×480 にしてエンコードすると映像・字幕とも高品質の出力が得られる。ただし、ファイルサイズが大きくなる。出力ファイルサイズ＝55.7 MB。

出力結果はソース映像のサイズや品質のほか、エンコーディング時の設定によってもかなりの違いが出る（パソコンの「ハードウェア・アクセラレーション」の設定も字幕の質に影響する。染谷 (2008a) の巻末資料 p. 31 参照）。したがって、ソース映像に応じた最適のエンコーディング設定を得るためにマニュアル設定でいろいろと試してみるのがよい。この場合、最初の 1〜2 分のみでエンコーディング作業を「中断」し、中断した時点までのファイル（中断時点で自動的に保存される）を再生してエンコーディングの状態を確認するようにすると時間を節約することができる。

[参考]　DirectVobSub での字幕書式設定例※（図 30）



[Main] タブでの書式設定例　　[Subtitle Style Editor] での書式設定画面

※Windows の[スタート]ボタンから[プログラム]→[VobSub]→[DirectVobSub Configure] で起動。

4. GOM Encoder を使った「ハードサブ」動画の作成

GOM Encoder はすでに染谷 (2008a, pp.17-20) で紹介した高機能メディアプレーヤ GOM Player を無償配布しているグレテック社が 2007 年 2 月に発表したエンコーダである。基本的には各種動画のフォーマット変換用ソフトウェアであるが、本格的なハードサブ作成の機能も備えており、作成したハードサブ字幕付き動画はどのプレーヤでも再生可能である。現在は機能が限定された無料のベータ版（試用版）がリリースされており、以下、これを使ったハードサブ動画の作成について解説する。

4.1 使用するアプリケーション

1) **GOM Encoder**（試用版）

ダウンロード先： http://www.gomplayer.jp/

インストールおよび設定解説： http://www.gomplayer.jp/encoder/

2) 対応する動画および字幕ファイルの形式

入力：AVI, MPEG-1/2, WMV, FLV, MOV, QT, RM, MKV, OGM, MP4 等。

出力：WMV, AVI, SWF, FLV, OGM, MP4 (H.264 対応)。

字幕ファイル： SAMI 形式の字幕ファイル（ただし、対応する動画ファイルと同じ名称で同一フォルダ内に保存されていること）。このほか、`srt` と `ssa` 形式についても動作確認済み。

コメント：字幕の書式設定は GOM Encoder の内部機能を使って行うため、字幕ファイルでの書式設定は基本的には無視される。なお、GOM Encoder の字幕書式設定パネルは、GOM Player の字幕書式設定パネル（染谷 2008a, 図 10a, p. 19) と基本的には同じものである。

4.2 GOM Encoder のインストールとプログラムの起動

1) まず、上記のダウンロード先から **GOM Encoder**（試用版）をデスクトップ上にダウンロード。

2) プログラムのインストールと初期設定については上記の URL に詳しい解説が掲載されているため、ここでは省略する。セットアップが終了すると、Windows のプログラムグループ内に GOM Encoder というフォルダが作成されるとともに、デスクトップに右のようなショートカットアイコンが作成される。

3) 上記のアイコンをクリックするとログインメッセージに続いてプログラムが起動し、**図 31** のような初期画面が表示される（ただし、この段階では**図 31** に見られるソース映像の情報は未表示）。

4.3 入出力映像の設定、および字幕の書式設定

4) この初期画面から、まずはソース映像を読み込む。以下の図の中央左側（コメント①参照）にある「追加」ボタンをクリックすると、「ファイルを開く」画面になるので、目的のフォルダに移動し、ソース映像を選択して「開く」をクリック。

5) ソース映像が読み込まれ、下部左側の「入力情報」ペインにソース映像の詳細データが表示される。また、対応する字幕ファイルがソース映像と同じフォルダ内にあれば、上部ペインの最初に行の「字幕」欄に「あり」と表示される（コメント②参照）。

図 31　GOM Encoder 初期画面

6) この状態から、次に画面上部右側にある「オプション」ボタンをクリック（コメント②参照）して、「オプション」の「字幕」設定に進む（**図 32**）。

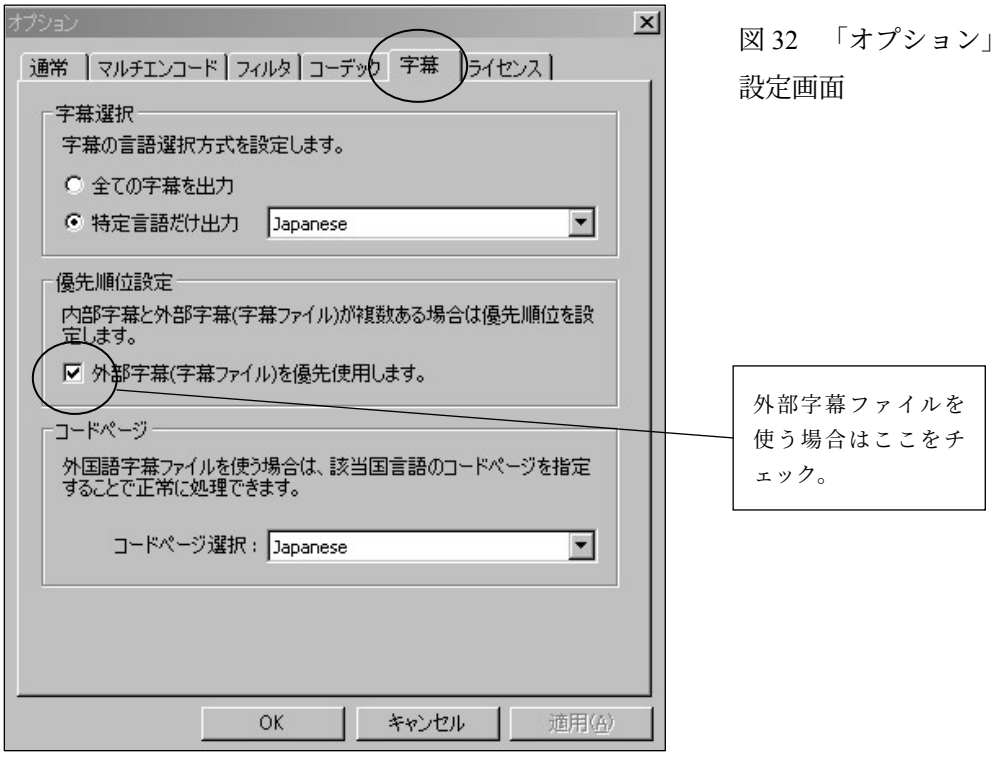
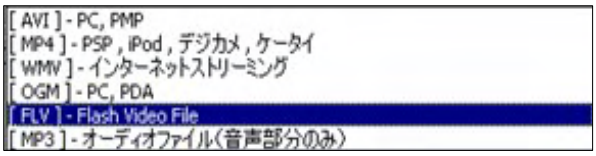


図 32　「オプション」→「字幕」設定画面

7) 「オプション」画面には以下の図に見られるように 6 つのタブが用意されているが、ここではこのうちの「字幕」タブを選択して字幕の言語、優先順位、コードページの 3 つをそれぞれ指定（他のタブはとりあえず初期設定のまま使う）。指定後、「OK」ボタンをクリックして「オプション」画面を閉じる。

8) 次に、**図 31** の画面から、画面右下にある「出力設定」ボタンをクリック（コメント④参照）。

9) 以下のような画面（**図 33**）が表示されるので、ここで出力形式を設定する。選択可能な映像出力形式は以下のように 5 種類あるので、ここから目的のフォーマットを選択する。



10) 出力形式を選択すると、それに応じたビデオおよび音声コーデックの設定が自動的に行われるので（**図 33** 下部参照）、通常はそのまま次に進む。ただし、必要に応じて適宜変更してもよい。

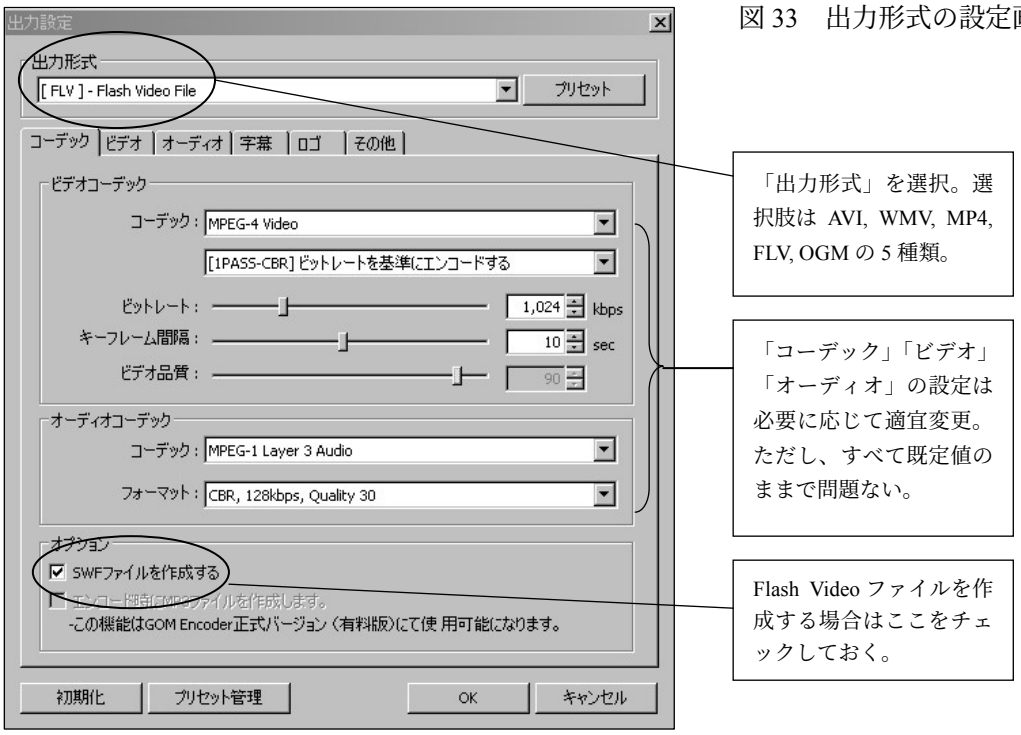


図 33　出力形式の設定画面

11) 次に、「字幕」タブをクリックして字幕書式の設定を行う（**図 34**）。映像上にエンコードされる字幕はここで設定された書式にしたがって行われるため、字幕ファイル内での書式設定は基本的には無視されることになる（ただし、字幕ファイル内の書式情報が書き換えられることはない）。

図 34　字幕書式の設定

12)　字幕書式の設定が終わったら「OK」ボタンをクリックして「出力設定」画面を閉じる。

13)　再度、初期画面に戻る。ただし、ここで表示されているのは、入出力ファイルの設定が完了してエンコード開始可能になった状態の画面である（**図 35**）。

図 35　入出力ファイルの設定が完了してエンコード開始可能になった画面

14) この状態から、画面下部右側にある「開始」ボタンをクリックするとエンコードが開始される。**図 36** に示すように、字幕ファイルがきちんと読み込まれていれば、プレビュー画面上でこれを確認することができる。なお、今回使った「試用版」の GOM Encoder では、最初の 10 秒間のみ画面右上にグレテック社のロゴマークが挿入される。

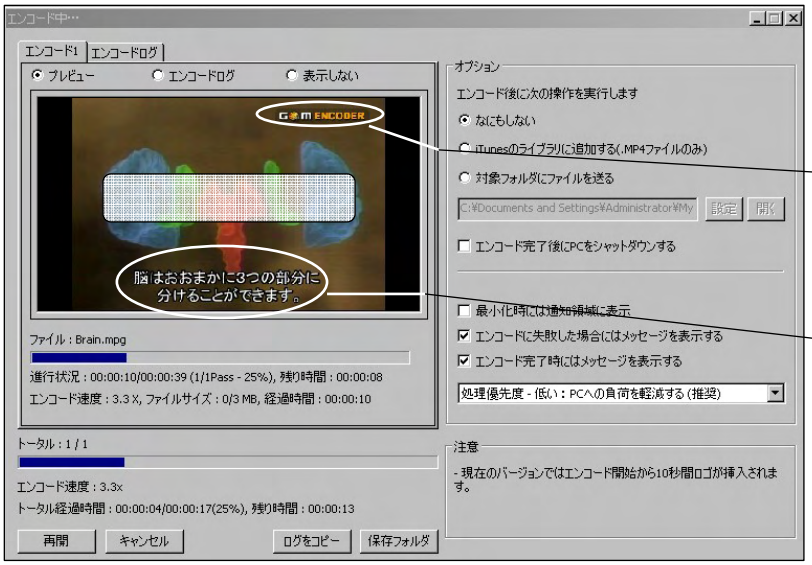


図 36 エンコード画面

「試用版」には最初の 10 秒間のみここにグレテック社のロゴマークが挿入される。

字幕がエンコードされているのをプレビュー画面で確認することができる。

15) エンコードが終了すると「エンコードが完了しました」というメッセージが表示されるので、「OK」をクリックした上で、開いている画面をすべて閉じて作業終了。**図 37** に字幕エンコード結果のイメージを比較したものを示す。左側の**図 37a** が GOM Encoder（試用版）による出力例（ただし、表示されているのは wmv 形式へのエンコード例）、右側の**図 37b** は第 3 章で解説した TMPG Encoder（無料版）による出力例（mpeg-2 へのエンコード例）である。いずれもほぼ同等の高品質の字幕が表示されている。

図 37 字幕エンコード結果のイメージ比較

図 37a GOM Encoder（試用版）による出力

図 37b TMPG Encoder（無料版）による出力

5. まとめ

以上、本稿では、第 1 章で VirtualDub、第 2 章で Media Subtitler、第 3 章と第 4 章ではそれぞれ TMPG Encoder および GOM Encoder という市販のエンコーダの無償試用版を使ったハードサブの作成方法について解説した。**表 1** は、本稿で取り上げた４つのソフトウェアの性能その他を比較した一覧表である。

表１　本稿で取り上げた４つのソフトウェアの比較表

ソフトウェア	対応メディア形式 ※1		対応字幕ファイル形式	操作性	字幕品質	有償・無償の別
	入力	出力				
VirtualDub	avi, mpeg -1	avi	srt, sub, ssa, ass, smi, psb	△	○	フリーウェア
Media Subtitler	avi, mpeg -1	avi	smi, srt, sub, ssa, etc. ※2	○	○	フリーウェア
TMPG Encoder＋DirectVobSub	wmv	mpeg-1 & 2	smi	○	◎	市販品（ただし無償の試用版あり）
GOM Encoder	wmv, avi, mpeg-1 & 2, FLV, mp4,etc.	wmv, avi, swf, FLV, mp4, etc.	smi, srt, ssa	○	◎	市販品（ただし無償の試用版あり）

※1　ハードサブの作成に関して対応しているメディア形式。

※2　ただし ssa 形式に変換した上で使用（フォーマットの変換は Media Subtitler 上で直接行うことができる）。

この４つのソフトウェアのいずれを使ってもハードサブを作成することができるが、字幕の品質および操作性という点では TMPG Encoder と GOM Encoder に分があるという結論になる。問題は、いずれも有償の市販品であるという点だが、無償の試用版が使えることから、授業ではとりあえず試用版をダウンロードして使うということで間に合わせることができる。

ただし、VirtualDub でも適切なコーデックの選択や出力サイズの設定、あるいはハードウェア・アクセラレーションの調整など、条件さえ整えれば十分に実用に耐える字幕品質を得ることができる。操作性という点でやや難があるように思われるが、これは要するに慣れの問題であり、とくに大きな問題はない。また、VirtualDub はハードサブの作成以外にもさまざまなフィルタが用意されており、拡張性が高いことから、ぜひとも使い方を覚えておくように勧めたい。一方、Media Subtitler は VirtualDub の機能を使ってハードサブを作成するため、字幕の品質は VirtualDub 単独で使用した場合と同等であるが、その名のとおり本来の字幕エディタとして優れた機能と操作性を持っており、こちらもぜひともインストールしておきたい。

【謝辞】本稿のうち、VirtualDub を使ったハードサブの作成方法については、青山学院大学外国語ラボラトリー嘱託職員として勤務しておられた益子亮氏によるメモを参考にした。ここに記して感謝の意を表したい。

補足資料：Windows のデータ実行防止 (DEP) の設定の変更

AVI ファイルを起動しようとすると、Windows に組み込まれているデータ実行防止 (DEP = Data Execution Prevention) 機能が働いて、ファイルの閲覧が阻止されてしまうことがある。サムネイル・プレビュー (thumbnail preview) 機能がオンになっている場合は、AVI ファイルのあるフォルダにアクセスするだけでもアクセスがブロックされる。これは、ほとんどの場合 DivX エンジン呼び出しのブロック、という問題で、DivX コーデックによって圧縮された AVI ファイルに対して問題を起こすということらしい [1]。

また、しばしば「アプリケーションエラー "0x00ae415b" の命令が "0x00000000" のメモリを参照しました。メモリが "written" になることはできませんでした」というエラー表示が出ることがあるが、これも（メモリーオーバーフローにかかわる現象ということで）同様の理由による問題と思われる。これは、DivX に特有の問題というわけではなく、古いバージョンのプログラムやデバイスドライバなどでは、それ自体にはとくに問題がなくても、この DEP 機能によって実行が強制的に中断させられてしまうことがある。

このような場合は、下記の方法によって DEP 機能を無効にすることで事態が改善されることがある[2]。この方法は、プロセッサのセキュリティ機能が無効になるために若干セキュリティレベルが低下するが、Windows 上で通常のセキュリティ設定をした上で、ウイルスソフトが入っていれば大きな問題はない。

<u>DEP 機能の解除</u>

1. 「スタート」→「設定」→「コントロールパネル」と進み、「システム」フォルダを開く。
2. 「システムのプロパティ」画面から「詳細設定」タブを選択し、「起動と回復」の「設定(T)」ボタンをクリック。
3. 「起動と回復」画面から「起動システム」の「編集(E)」ボタンをクリック。
4. 次のような内容のテキストファイル (boot.ini) が表示されるので、末尾にある `NoExecute=`**`OptOut`** を `NoExecute=`**`AlwaysOff`** に書き換える。

5.　変更後、ファイルをそのまま再保存したのち、パソコンを再起動。

DEP 機能を解除したくない場合の対策としては、DivX を最新の Version 6.5+ に置き換えることや、Windows Vista を使っている場合は最新の Vista Codec Package にアップデートする等の方法が推奨されている[3]。なお、コーデックを新たに入れる場合は必ず Windows のシステム回復ポイントを新たに作成し、問題があった場合にいつでも導入前の状態に回復できるようにしておくこと。

このほか Adobe Acrobat Standard (Ver. 7) でも、一部の PDF ファイル（とくに古いバージョンで作成したファイル）を開くときに前述の「アプリケーションエラー "0x00ae415b" の命令が "0x00000000" のメモリを参照しました。メモリが "written" になることはできませんでした」というエラー表示が出て、作業が強制中断されることがある。この現象は DEP 設定の変更や最新のアップデートのインストールでも改善されない[4]。対策としては（最新版の Adobe Acrobat を新たに購入するオプションは除外するとして）、無料の Adobe Reader 9 を Adobe 社のサイトからダウンロードし、Adobe Acrobat では開けないファイルを読むときはこちらを使うようにすればよい。もちろん、PDF ファイルの作成にはこれまでどおり Adobe Acrobat を使うことができる。

参考サイト：

[1]　http://www.icholon.co.jp/pcsoft/support/DVD/vep_kaitei/FAQ_1.html

[2]　http://www.atmarkit.co.jp/fwin2k/win2ktips/501dep/dep.html

[3]　http://www.howtogeek.com/howto/windows-vista/fix-for-com-surrogate-has-stopped-working-error-in-vista/

[4]　http://www.tohdamikio.com/2008/04/post_0381.html

【注】

1) 主として英語から日本語の翻訳を中心に、稲生衣代准教授が指導に当たっている。本授業の実践報告については稲生 (2004) および稲生 (2007) 参照。ちなみに、「翻訳 (1)」では文芸翻訳を、「翻訳 (2)」では実務翻訳を中心にした翻訳の授業が行なわれている。いずれも通年科目である。

2) Subtitle Workshop および Viplay は、いずれも URU Soft 社（http://www.urusoft.net/）がフリーウェアとして公開しているものである。なお、最新の ViPlay (Ver. 3) には字幕表示に関するメニューが見当たらないことから、バージョンアップに伴って字幕表示機能が削除された可能性がある。

3) 字幕データを SAMI 形式で作成すればよい。SAMI (Synchronized Accessible Media Interchange) とはマイクロソフト社が公開している字幕表示技術で、WMP にもこの方式による字幕表示機能が実装されている。この技術を用いて字幕情報を映像と同期させて表示させるためには、SAMI の文法に従った字幕ファイル＝SAMI ファイル（.smi という拡張子を持つテキストファイル）を作成する必要があるが、Subtitle Workshop は SAMI 形式にも対応しており、SAMI 独自の文法をとくに意識せずに SAMI ファイルを作成することができるようになっている（基本的には保存時に .smi 形式を選択するだけでよい）。ただし、SAMI ファイル自体は一般の文書作成用エディタでも十分に作成可能であり、とくに専用のソフトウェアを必要としない。詳しくは染谷 (2008a) を参照されたい。

4) テスト用の動画ファイルと字幕ファイルが必要な場合は以下のサイトからダウンロード。
http://www.someya-net.com/83-SubtitleVideo/Public/TestSample.html

5) あるいは、C:¥Program File¥Gabes¥VobSub の中にある textsub.vdf を手動でコピーし、[VirtualDub-1.8.3] のフォルダの中の [plugins] フォルダに貼り付けてもよい。なお、.vdf という拡張子はこれが VirtualDub Filter であることを示す。

6) DirectVobsub (VobSub for DirectShow) はソフトサブ用のプラグインで、VobSub のインストールとともに自動的にインストールされる（図 2 参照）。DirectVobsub がオンになっていると、以下の作業で動画ファイルを読み込んだときに DirectVobsub が起動してしまうことがあるので、あらかじめこれをオフにしてから作業を行う。操作手順については染谷 (2008a, p. 28) 参照。

7) 今回は TextSub を使ってハードサブを作成するが、VirtualDub にはその他にもさまざまなフィルタが用意されている。例えば pp. 11-13 では、resize というフィルタを使って動画のサイズを変更しながら字幕をエンコーディングする例を示した。これらの各種フィルタについての解説は本稿の範囲を超えるため、ここでは触れないが、ぜひとも時間のあるときに試用してみることをお勧めしたい（各フィルタについての解説は VirtualDub のメニューから [Help]→[Contents]を選択し、[Video filter reference] の項目で参照することができる）。

8) SAMI 形式（拡張子.smi）の字幕ファイルは公式にサポートされており、筆者の PC 環境では問題なく読み込むことができるが、もし問題があるようならいったん SubRip 形式（拡張子.srt）または Sub Station Alpha 形式（拡張子.ssa）に変換してから使うとよい。ファイル形式の変換は Subtitle Workshop を使って以下の手順で行う。

   1. Subtitle Workshop を起動（Subtitle Workshop については染谷 (2008d, pp. 16-20) 参照）。
   2. メニューから [File]→[Load]→[Original] を選択（**図 38**）。「ファイルを開く」のページから、読み込みたい字幕ファイルが保存されているフォルダを表示して、当該のファイルを読み込む。

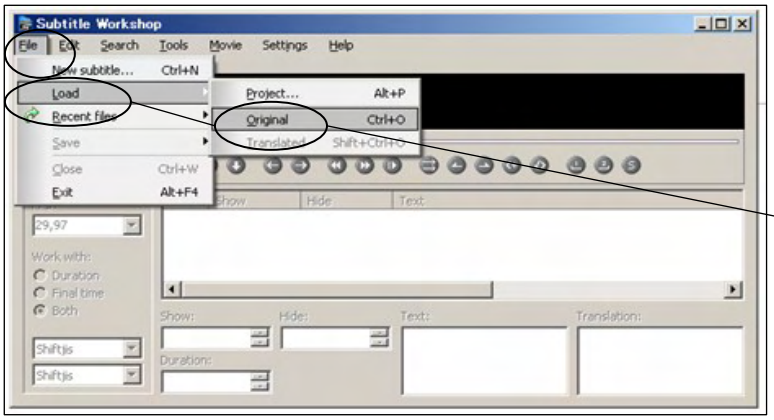


図 38　字幕ファイルの読み込みダイアログ画面

3. 字幕ファイルが読み込まれたら、今度はメニューから [File]→[Save]→[Original as] を選択し（**図 39a**）、保存したいフォーマット形式のダイアログ画面（**図 39b**）で [SubRip] または [SubStation Alpha] を選択してダブルクリック（一覧表が表示されていない場合は画面下部の「All formats（全てのフォーマット形式の表示）」にチェックを入れる）。
4. 「名前をつけて保存」の画面になるので、ここで任意の名称（拡張子は`.srt`）を入力して適切なフォルダ内にファイルを保存。保存を確認したら、Subtitle Workshop をそのまま終了。

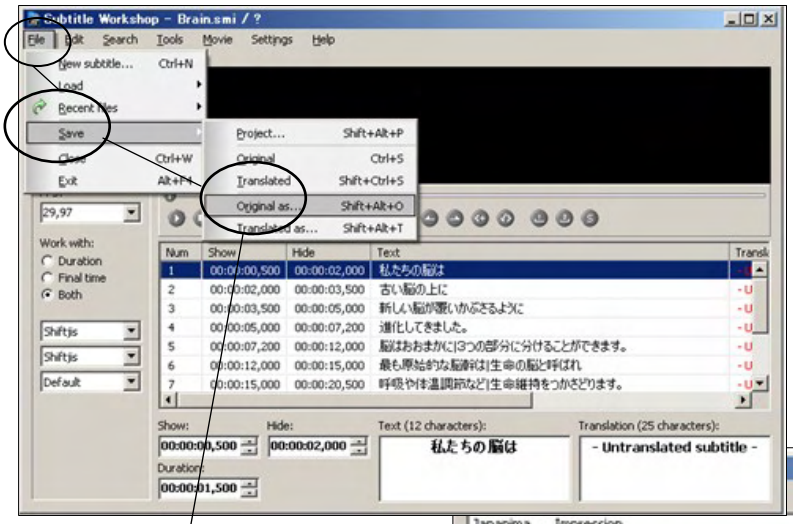


図 39a　字幕ファイルの保存ダイアログ画面

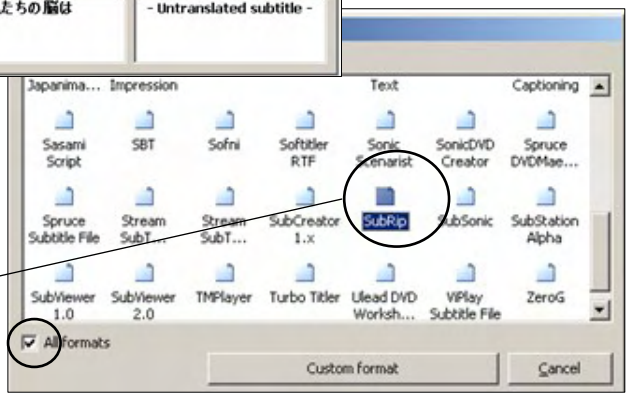


図 39b　字幕ファイルの保存形式の選択画面

9) このほか、筆者の PC 環境では Xvid MPEG-4 Video Codec や DivX 6.8.4 Codec でも比較的良好なエンコーディング結果が得られた（Xvid については p. 14 参照）。なお、Codec（コーデック）とは COmpression and DECompression (aka, COding and DECoding) を縮めた造語で、映像や音声データを圧縮・展開するた

めのプログラムの総称。映像や音声ファイルは何らかのコーデックを使って圧縮（＝エンコード）されており、これを再生するためには、同じコーデックを使ってデータを展開（＝デコード）する必要がある。いわゆるファイル形式とコーデックはまったく別物で、例えば同じ avi 形式の動画でも自分のパソコンに入っているコーデックによって、ある avi ファイルは再生できるのに別の avi ファイルは再生できない（あるいは再生できても画質やサイズなどが著しく異なる）といったことが起こる。普段はあまり気にする必要もないが、動画ファイルを扱う場合には、自分のパソコンにどのようなコーデックが入っているか（あるいは入っていないか）を、いちおうは知っておいたほうがよい。Windows XP (SP2) の場合は、次の手順でこれを調べることができる。

1. [スタート] ボタンから [設定]→[コントロールパネル] → [システム] → [ハードウェア] タブ→ [デバイスマネージャ] を選択。
2. ここから [サウンド、ビデオ、およびゲームコントローラ] をダブルクリックするとその中に [ビデオ CODEC] という項目があるので、これをクリックして、[プロパティ] タブを再度クリック。
3. これで、現時点でインストールされているコーデック一覧が表示される（サウンドのコーデックを調べたい場合は [ビデオ CODEC] の代わりに [オーディオ CODEC] をクリック）。

ただし、通常のインストール手順を経ずに導入したコーデックは上記の一覧表に登録されていないことがある。なお、ある動画／音声ファイルがどのようなコーデックで処理されているかを調べるには、すでに紹介した「真空波動研」というフリーウェアが役に立つ（染谷 2008a, **注 20**）。これを使うと、映像・音声合わせて 500 種類以上のコーデックを判別することができる。

10) TMPGEnco Plus 2.5 のマニュアルによると、「Directshow Multi Media Reader プラグインを使用することで (wmv の) 読み込みが可能となる場合がある」とある。このプラグインを使用する場合は「VFAPI (=Video File Application Program Interface) プラグイン」の優先度を変える。具体的には、TMPG Encoder のメニューから「設定」→「環境設定」→「VFAPI プラグイン」の順に開き、一覧表にある [Directshow Multi Media Reader] の優先順位を [AVI2 (OpenDML) File Reader] よりひとつ上げればよい（項目をマウスで選択→右クリックで「優先度を上げる」をクリックすると数字がひとつ上がる。現在は -1 または -2 になっていて優先度が下げられているので、これを 1 にする）。ただし、筆者の PC 環境では初期設定のままで問題なく wmv が読み込めている。もし、初期設定の状態では読み込めないようなら設定を変えるようにするとよい。ただし、動作は保証しないとのこと。

11) DirectVobSub は本来 mpeg または avi 形式の動画に対応したものであるが、TMPG Encoder を間に挟む場合に限って wmv 形式にのみ反応するようになる。その理由はよくわからない。「非公式の裏技」たる所以である。

12) 「アスペクト比」とは映像の横縦サイズの比率のことを指す。一般のテレビ放送 (NTSC＝北米・日本, PAL＝欧州) の場合は 4:3 で、DVD-Video では 16:9 または 4:3 の比率が用いられる。4:3 と 16:9 はいずれも DVD の規格に準拠しており、それぞれ再生時に指定の比率に合わせて画像が伸張される。以下に、ソース映像と推奨アスペクト比の対応一覧を掲げる。なお、アスペクト比 1:1 (VGA) はソース映像のピクセルサイズをそのまま画面上に表示する設定で、パソコン上での再生を目的とした場合に使用する。どれを選択すべきかよくわからない場合はこれを選んでおく。参考までに、以下に、ソース映像のサイズと推奨アスペクト比との関係を一覧表にしたものを引用しておく。

ソース映像	推奨アスペクト比
320×240, 640×480 ピクセル、または下記のいずれにも該当しないソース映像の場合	1:1 (VGA)
352×240 (Video-CD); 704×480, 720×480 (DV, DVD) などの NTSC 方式のソースの場合	4:3 525 line (NTSC)
352×240; 704×480, 720×480 (DV, DVD) などの NTSC 方式 16:9 ワイドソースの場合	16:9 525 line (NTSC)
縦横の画面比率が 4:3 で表示されるムービーがソースの場合	4:3 Display
縦横の画面比率が 16:9 で表示されるワイドムービーがソースの場合	16:9 Display

（出典：「TMPGEnco Plus 2.5 クイックマニュアル」2003 年 ㈱ペガシス）

13) 「インターレース」とは、従来のアナログテレビで使われている映像表示技術で、走査線を偶数ラインと奇数ラインに分けて飛び越し走査（interlace scan）して映像を表示するというものである。この方式は、1 画面を本来の半分の走査線数で表示するため、画質が高いとはいえなかった。これに対して、BS デジタル放送では、1 画面を上から順番に走査する「ノンインターレース」（プログレッシブ）方式を採用しているため、ちらつきのない高い画質の映像が得られるようになっている（以上、『ウィキペディア（Wikipedia)』参照)。コンピュータのディスプレイは基本的に「ノンインターレース」対応であるが、そのパソコン上で、「インターレース」方式の映像（アナログテレビ放送の録画映像や、家庭用テレビで鑑賞するための DVD-Video から取り込んだ動画等）を再生すると、横にライン状のノイズ（＝インターレース縞）が乗ることがある。これを取り除くことで、よりきれいな画像を得ることができる。これを「ノンインターレース化」または「プログレッシブ化」と呼ぶ。TMPG Encoder には MPEG-2 へのエンコーディングについて「インターレース縞」を取り除きながらエンコードできるようになっている（「MPEG の設定」→「エンコードモード」で選択。ソース映像がはじめからノンインターレース映像の場合でも「ノンインターレース」方式のエンコードモードでよい)。

【参考文献･資料※】

[01] Jmet (2005) “VobSub Subtitle Ripping, Playback and Permanent Subtitle Guide.” (Article posted on the Digital Digest website) [Online] http://www.divx-digest.com/articles/article_vobsub_page1.html

[02] Zsögön, Andrés (2005) “Getting Started With DivXLand Media Subtitler.” [Online] http://www.divxland.org/msubtitlerdocs/GettingStarted_eng.pdf

[03] 稲生衣代 (2004) 「 大学教育における『映像翻訳コース』の指導手法に関する研究」日本通訳学会発行『通訳研究』第 4 号 (pp. 83-101)

[04] 稲生衣代 (2007) 「共同学習を導入した映像翻訳教育に関する考察」日本通訳学会発行『通訳研究』第 7 号 (pp. 147-165)

[05] 染谷泰正 (2008a) 「動画への字幕の挿入について：英語教員のための Do-It-Yourself マニュアル (1)」

[06] ---------- (2008b) 「字幕付き動画のオンライン配信について：英語教員のための Do-It-Yourself マニュアル (2)」

[07] ---------- (2008d) 「字幕付き動画作成のための SAMI ファイルの作成・編集について：英語教員のための Do-It-Yourself マニュアル (4)」

[08] ㈱グレテック (2008) 「GOM Encoder インストールガイド」[Online] http://www.gomplayer.jp/encoder/

[09] ㈱ペガシス (2003) 「TMPGEnco Plus 2.5 クイックマニュアル」

※ URL はいずれも 2008 年 6 月 30 日現在のもの。

字幕付き動画作成のための SAMI ファイルの作成・編集について

～ 英語教員のための Do-It-Yourself マニュアル (4) ～

染谷 泰正
（青山学院大学）

字幕付き動画作成のために必要となる「字幕ファイル」には、現在、さまざまな形式のものが提案されている。染谷（2008a）では、そのうちの主要なフォーマットのひとつである SAMI (Synchronized Accessible Media Interchange) 形式について紹介し、その基本的な文法と書式について解説した。その上で、読者の便宜を考慮して、日本語用と日英 2 ヵ国語用の字幕ファイル「テンプレート」をそれぞれ用意し、このテンプレートを使った SAMI ファイルの作成手順について述べた (Ibid., pp. 13-15) 。

ここで作成した字幕ファイルは、とりあえずテンプレートに字幕データを流し込んだだけの「暫定版」あるいは「作業用ファイル」という性格のものである。この作業用ファイルができ上がれば、今度は実際に映像に字幕を同期表示させながら、それぞれの字幕のタイミングや書式設定、および字幕内容の調整等を必要に応じて行うことになる。字幕の同期タイミングをミリセコンド単位で正確に計測するためには、その目的のために開発された専用の「字幕エディタ」を使うのが望ましいが、何らかの事情でより簡便に作業を行いたい場合には、ごく一般的な文書作成用エディタを使うこともできる。

以下、本稿では、まず通常の文書作成用エディタを使った字幕ファイルの作成方法について述べ（第 1 章）、さらに、その補足として、ミリセコンド単位でのタイミング計測をするための簡易ツールを紹介する（第 2 章）。その後、第 3 章では、フリーウェアとして入手可能な各種「字幕エディタ」のうち、比較的扱いやすいものをいくつか紹介しながら、より正確な字幕ファイルの作成について述べ、最後の第 4 章で全体のまとめを行う。

1. 一般的な文書作成用エディタと WMP を使って字幕ファイルを編集する

テンプレートを使って暫定ファイル（作業ファイル）を作成したあと、その完成度を高めるためには、前述のとおり、作成した字幕を実際の映像に同期表示させ、これを見ながらそれぞれの字幕のタイミングや書式設定、内容等の調整を行う。そのためのもっとも簡単な方法は、

SOMEYA, Yasumasa (2008d), “Creating and Editing SAMI files” (Part 4 of the monograph, *Adding subtitles to video clips and embedding them in your Web pages: A Do-It-Yourself manual for teachers of English wishing to introduce subtitle translation as part of their classroom activities*, 2008.)
(c) 2008 Yasumasa Someya

WMP (Windows Media Player) の字幕表示機能を使って映像と字幕の同期タイミングをチェックしながら、同時に「TeraPad」（フリーウェア）や「秀丸」（シェアウェア）などのごく一般的な文書作成用エディタ（染谷 2008a, p. 13）を使って字幕ファイルに修正を加えていくというやり方である。これは、通常、以下のような手順で行う（以下、SAMI ファイルと映像ファイルが Sample.smi および Sample.wmv のように同じ名称で同一フォルダに保存してあるものとする）。

1. 編集したい SAMI ファイルをエディタで開く。（以下の**注 1** 参照）
2. WMP で動画ファイルを開く。（同**注 2** 参照）
3. **図 1** に示すように、映像と字幕ファイルを作業しやすいように適当なサイズにして並べる。
4. 映像を再生して字幕各行のタイミングや内容をチェック。修正が必要であれば SAMI ファイルに適宜修正を加えていく。

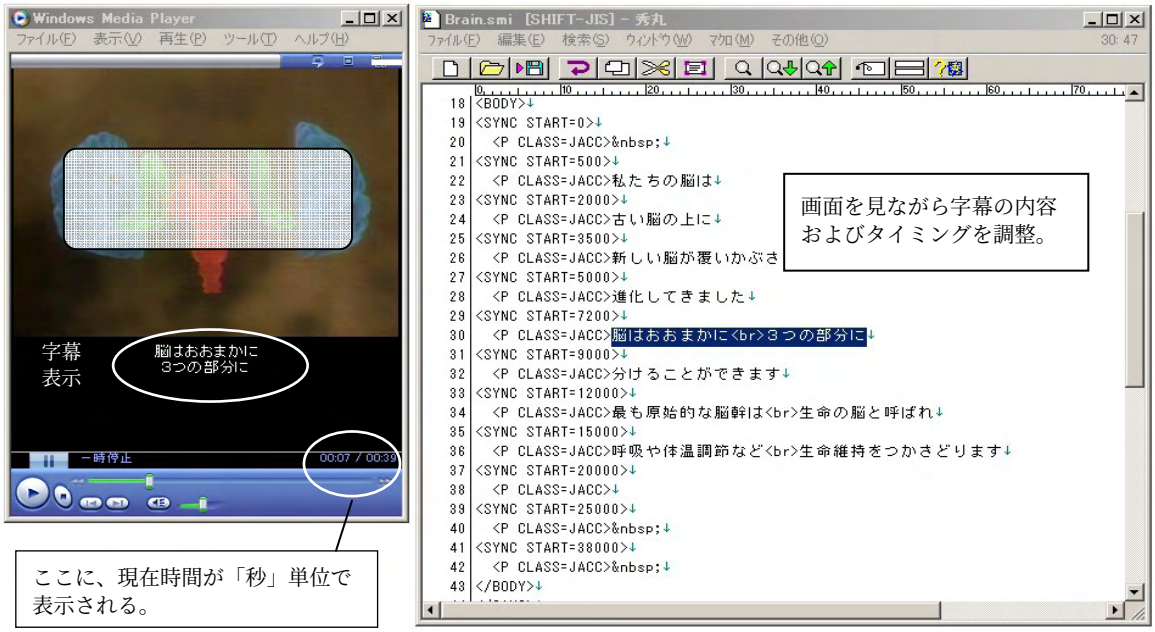


図 1　映像に字幕を同期表示させながら字幕ファイルに修正を加える

注 1）1 と 2 を逆にして最初に WMP で動画ファイルを開いた場合、字幕ファイルは（すでに動画ファイルに関連付けられているため）書き換え禁止の状態となり、映像を見ながら修正を加えることができなくなる。したがって、必ず最初に字幕ファイルを開いてから WMP で動画を立ち上げること。

注 2）作業に先立って、あらかじめ WMP のメニューから「再生」→「歌詞、キャプション、および字幕」を選び、「利用できる場合はオン」をチェックしておくこと（染谷 2008a, pp. 15-16 参照）。なお、動画ファイルが mpeg または avi 形式で、かつパソコン上に DirectVobSub がインストールしてある場合は、あらかじめ DirectVobSub をオフにしておくこと（染谷 2008a, p. 28 参照）。

5. 最後まで修正が完了したら（あるいは何らかの事情で途中で作業を中断したい場合は）、まず WMP を終了し、その後、字幕ファイルを「上書き保存」する。

6.　再度 WMP で動画ファイルを開き、修正した字幕のタイミングおよび内容を確認する。

この方法の利点は、特別なソフトウェアを使用することなしに手軽に字幕ファイルの編集ができるという点である。ただし、WMP での時間表示機能が「秒」単位に限定されているために、SAMI ファイル上の開始時間 (START time) をミリセコンド単位で厳密に指定することができないという欠点がある。WMP でのおおざっぱな計測で特に問題がない場合はよいとして、早いテンポで切り替わる会話シーンのように、字幕切り替えのタイミングが重要な場合はミリセコンド単位で正確にタイミングを測れるほうが都合がよい。

もちろん、そのためには専用の「字幕エディタ」を使えばよいわけだが、教室で使うことを想定した場合、これはこれで問題がないわけではない。「字幕エディタ」の使い方を学生に覚えさせなければならないという手間が加わるからである。そこで、次項ではとりあえずもう少し簡便な方法を紹介しておく。

2. 「Windows Media ファイルエディタ」を使って正確なタイミングを計測する

WMP に代わってミリセコンド単位で映像の表示経過時間を表示してくれるメディアプレーヤがあればよいのだが、現在のところそのようなプレーヤは見当たらない。通常のメディアプレーヤでは、もともとそのような機能は必要ないからであろう。ただし、幸いなことに、マイクロソフト社で無償提供している「Windows Media ファイルエディタ (Ver. 9)」というソフトウェアにはミリセコンド単位で映像の再生時間を表示する機能があり、これを利用することで、より厳密な字幕表示タイミングの計測が可能になる。

「Windows Media ファイルエディタ」（以下、WMEd）は、「Windows Media エンコーダ 9 シリーズ」に含まれるユーティリティツールのひとつで、もともとは長い動画ファイルに挿入する再生位置マーカーの作成や、指定された時間にウェブページを開いたりメッセージを表示させたりするためのスクリプトコマンドの挿入、あるいはコンテンツに関する表示情報を追加するといった各種編集作業をするためのソフトウェアである。WMEd を含む「Windows Media エンコーダ 9 シリーズ」は、マイクロソフト社のダウンロードセンター (MS Download Center) から無償でダウンロードすることができる [1)]。なお、「Windows Media エンコーダ」は「Windows ムービーメーカ」[2)] とともに、wmv 形式の動画ファイル作成・編集に必須のものであり、WMEd を使用するかどうかにかかわらずインストールしておくべきものである。

ダウンロード先：http://www.microsoft.com/japan/windows/windowsmedia/download/encode.aspx

2.1　「Windows Media エンコーダ 9」(WME9) のインストール

WME9 のインストールファイル (WMEncoder.exe, 9.55 MB) を上記のサイトからデスクトップ上にダウンロードしたあと、このファイルをダブルクリックしてインストールを開始。インストール作業は「セットアップウィザード」の指示にしたがって進めていくだけで、ごく簡単に行うことができる。

インストールが終わると、c:¥Program Files¥Windows Media Components の中に [Encoder] というフォルダが作成されるとともに、Windows のプログラムグループ（[スタート]→[プログラム]）にも [Windows Media] というフォルダが（ない場合は）新たに作成される。この [Windows Media] フォルダの中に「Windows Media エンコーダ (WME9)」本体が収録され、さらに [ユーティリティ] という名称のサブフォルダ内に、目的の「**Windows Media ファイルエディタ (WMEd)**」が収録されていることが確認できる（**図 2**）。[3)]

図 2　Windows メニューから「Windows Media ファイルエディタ」を確認

2.2　「Windows Media ファイルエディタ」の起動と字幕タイミングの計測

WMEd がインストールできたら、次にこれを立ち上げて目的の動画を読み込み、厳密なタイミングの測定が必要な箇所についてその発話開始時間（および必要に応じて終了時間）をチェックする。なお、WMEd では画面上に字幕を同期表示することができないため、基本的な字幕編集作業は前記の**図 1** に示したように WMP を使って行い、微妙なタイミングの調整が必要な箇所のみを WMEd を使って再計測する、というやり方が望ましい。WMEd を使って微調整した箇所が映像上で正しく同期しているかを確認するためには、再度、WMP に戻って確認することになる。

WMEd を起動するには、前記の**図 2** から「Windows Media ファイルエディタ」を選択する。起動後、メニューから「ファイル」→「開く」を選択して目的の動画ファイルを読み込んだ状態を**図 3** に示す。

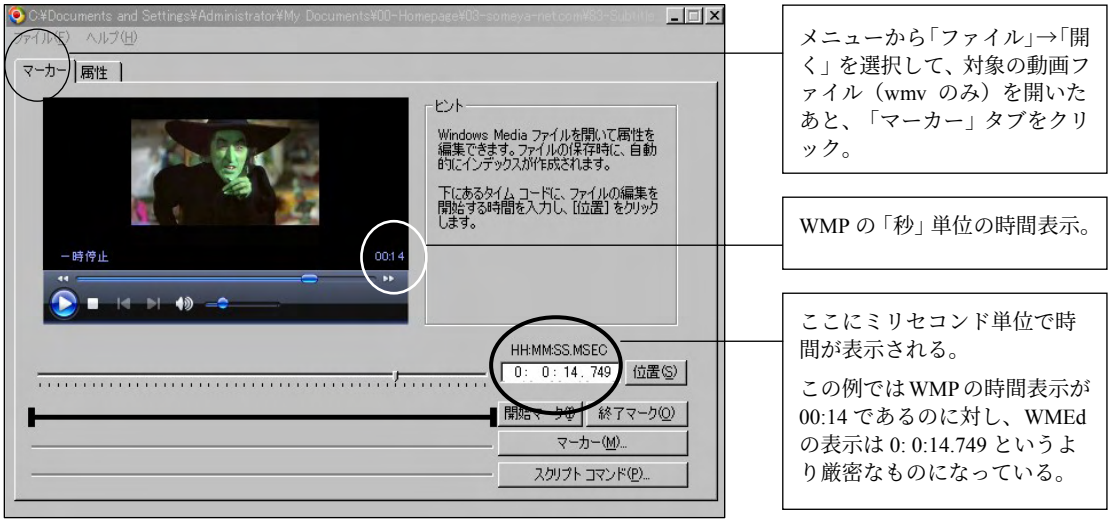


図 3　「Windows Media ファイルエディタ (Ver. 9)」の作業画面

画面上に表示される映像は、通常の WMP での操作と同じように再生、一時停止、巻き戻し等ができる。また、埋め込みプレーヤの映像フレーム右下部には、現在表示されている映像シーンの位置（経過時間）が「秒」単位で表示される。埋め込みプレーヤの下部にはプレーヤと連動した WMEd のシークバーがあり、その右横にミリセコンド単位で映像の現在位置が表示されている [4)]。たとえば、**図 3** の例では WMP の時間表示が 00:14（14 秒）であるのに対し、WMEd の時間表示は 00:00:14.749（14.749 秒）という、より厳密な表示になっていることがわかる。

もっとも、実際上は 100 ミリセコンド（0.1 秒）単位でタイミングを合わせておけば十分で、それ以上の端数にわたる厳密さが求められる場面はほとんどないように思われる。**図 3** の例でも、これを 00:00:14.700（14.7 秒）としても（あるいは、前後 0.1 秒ずつ増減しても）映像と字幕のズレを感じることはほとんどない。『人間工学基準数値数式便覧』（佐藤 1992,　p. 163）によれば、われわれの一般的な視覚刺激と聴覚刺激への単純反応時間はそれぞれ平均 194.25ms および 148.5ms となっており、刺激の提示から反応までおよそ 200-150ms 程度の遅れ（あるいは猶予）があることがわかっている [5)]。ようするに、字幕の同期タイミングの計測に当たって必要なのは感覚上の同期感であって、過度の機械的厳密さを求める必要はないということだろう。

なお、せっかく WMEd を使うならば、単なるタイマーとして使うだけでなく、WMEd 本来の動画ファイル編集機能の使い方も覚えておきたい。とくに、長いファイルの「切り詰め編集」や、クリックひとつで任意の箇所に移動できる「マーカー」の挿入は教育的見地からも便利な機能であり、この機会に是非とも覚えておくとよい。これについては詳しい解説資料を本稿の巻末付録 (pp. 22-32) として添付しておいた。併せて参照しておいていただきたい。

3.　専用の「字幕エディタ」を使って字幕ファイルを編集する

前節で述べたとおり、SAMI ファイルの作成は、基本的には一般的な文書作成用エディタと WMP を使って行い、微妙なタイミングの調整が必要な箇所についてのみ WMEd を使って再計測する、という方法でかなりの程度まで正確な作業を行うことができる。ただし、専用の「字幕エディタ」を使えば、これらの作業をより効率的に行うことができる。

どちらを選択するかは、それぞれの必要に応じて決めればよいことであるが、字幕の表示タイミングを全編にわたってミリセコンド単位でより厳密に設定したい場合や、何らかの理由で SAMI 形式以外のフォーマットで字幕ファイルを作成する必要がある場合、あるいは、本稿で意図しているような語学学習の一環としての字幕作成・字幕翻訳ではなく、（将来のプロの道も視野に入れた）本格的な「字幕翻訳」の授業という枠組みで指導ができる場合は、字幕作成専用のソフトウェア（以下、「字幕エディタ」）を導入することを勧めたい。

現在、「字幕エディタ」にはかなり高価な市販品から無償のフリーウェアまで数多くのものが出回っているが、本稿ではこれまでの既定方針（染谷 2008a, p. 1）にしたがって、フリーウェアとして入手可能なものについてのみ取り扱う [6)]。筆者が直接入手し、試用することができたフリーウェア（で、かつ紹介に足ると判断されるもの）としては SubCreator, Subbu, ChronoSub, DivXLand Media Subtitler, Sub Station Alpha, Subtitle Processor, Subtitle Workshop などがある。ひとくちに「字幕エディタ」といっても、機能を絞り込んで操作性を重視したものから、「プロ

仕様」の仕事ができるほど充実した機能を備えているものまで、さまざまであるが、以下、これらの「字幕エディタ」について、順を追って紹介していく。なお、ここではそれぞれの「字幕エディタ」の使用法を（操作マニュアルのように）細かく解説するのではなく、あくまでも概要の紹介という範囲にとどめることとする。

3.1 SubCreator (Ver.1.2)

前記のリスト中から、とにかくシンプルで使いやすいものという基準で選ぶとすれば、その筆頭は SubCreator (Ver.1.2) であろう。**図 4** に示すとおり、ユーザインターフェイスもシンプルで、機能が絞られているため、マニュアルなしでほぼ直感的に操作することができる。

基本的な画面構成は、上部に映像パネル、下部に字幕パネルとなっており、上部パネルの映像をみながら、字幕パネル上で直接データを編集する。なお、画面上にはプレーヤの操作ボタンがなく、基本的にはキーボードショートカットを使って操作を行う（例えば [Ctrl]＋[Space] バーで再生／一時停止をコントロール）。キーボードショートカットによる操作は、慣れればマウスを使った作業よりも格段に効率的な作業が可能になる。

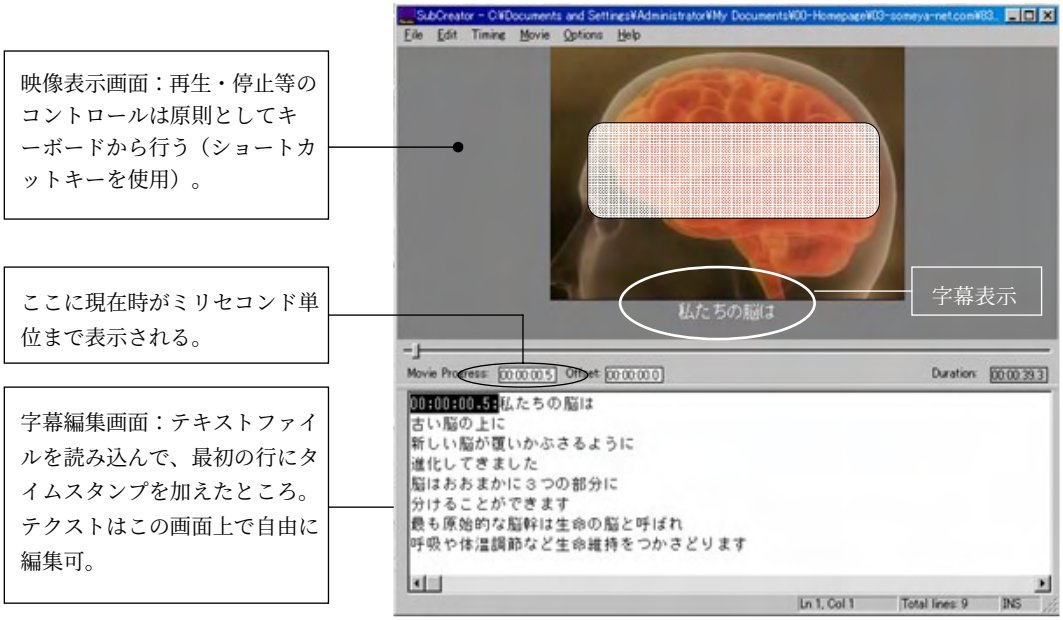


図 4　SubCreator (Ver.1.2) の作業画面

動画ファイルは wmv, mpeg, avi など標準的なファイル形式に対応し、字幕ファイルは独自の Simple Time Format (stf) のほか、srt と ssa に対応している（ssa 形式の場合は字幕の書式について詳しい設定が可能。stf 形式で作成したファイルは srt または ssa に変換可）。SAMI ファイルの読み込みも可能だが、スタイル設定は無視される。ただし、ファイルの読み込み時に、フォーマット変換をせずに読み込めば（読み込み時に [convert] か [cancel] かを聞いてくるので、後者を選択する）、元の SAMI ファイルのデータをそのまま生かすことができる。この場合、通常の文書エディタと同じように、手動でファイル中のデータを書き換えることになる（要す

るに「プレーヤ付きの文書エディタ」と考えればよい。これはこれで便利なツールであることに代わりはない）。

SubCreator を使った字幕ファイルのもっとも簡単な作成方法は次のとおり。①まず、字幕データをテキストファイルとして作成、②これをそのまま SubCreator に読み込み、③映像を見ながら「タイムスタンプ」（同期時間表示）を字幕各行に加えていく（**図 4** 参照）。タイムスタンプは [Ctrl]＋[A] で挿入。カーソルは自動的に次の行に移るので、映像の再生をショートカットキーでコントロールしながら、次々にタイムスタンプを加えていくことができる。④最後までタイムスタンプの挿入が終わったら、ファイルを srt または ssa 形式に “Export” し、保存して終了。srt または ssa 形式で作成した字幕ファイルを改めて SAMI 形式（拡張子 smi）に変換したい場合は、後述の Subtitle Workshop を使うとごく簡単な手順で変換作業を行うことができる（基本的には元のファイルを読み込み、SAMI 形式を指定して保存するだけでよい）[7]。

なお、プログラムを以下のサイトからダウンロードしてインストールしたあと、プログラムフォルダ内に作成される RICHED32.DLL という名称のファイル（アプリケーション拡張ファイル）を無効にしておくこと（ファイル名の前に×印を加えておく。ただし、削除しても問題ない。そのままにしておくと、漢字が一部文字化けするなどの問題が発生する）。

入手先※：

- http://www.radioactivepages.com/index.php?docid=subcreator&section=software

[Mirror Sites]

- http://www.free-codecs.com/download/SubCreator.htm
- http://www.softpedia.com/get/Multimedia/Video/Encoders-Converter-DIVX-Related/SubCreator.shml

※ソフトウェアの入手先最終確認日は 2008 年 10 月 30 日（以下同）。

3.2　Subbu (0.3.0)

字幕エディタとしての Subbu (0.3.0, 2005-11-27 release) の最大の特徴は、映像表示に加えて wav 音声の波形表示ができるようになっているという点で、ソフト設計の基本思想として、音声波形画面で正確な同期タイミングを測り、映像画面で字幕の位置確認や書式設定を行うという使い方が想定されている（**図 5a, 図 5b** 参照）。

対応している映像フォーマットは avi のみで、音声は wav 形式。字幕ファイルは srt, ssa, ass のいずれかを初期値として設定することができる。また、通常のテキストファイルとして作成した字幕データを読み込んで編集することも可能。ただし、文字コードの設定が適切に行われていないとうまく読み込めないことがあるため、日本語の字幕を作成する場合は、最初から Subbu 上で字幕データの入力を行うのが望ましい（SAMI 形式への変換は前述のとおり Subtitle Workshop を使えばよいので、フォーマットについてはとくに問題にならない）。

編集画面は比較的シンプルで扱いやすく、日本語化パッチが付属しているためメニュー表示を日本語化することもできる。ただし、操作法を詳しく解説したヘルプファイルが付属していないため、最初のうちは多少の遅行錯誤が必要になる。なお、筆者の記憶では、映像の読み込

みに伴って自動的に音声波形が抽出されるようになっていたはずであるが（記憶違いの可能性もある）、この稿を書いている現時点では、音声波形は wav ファイルを別個に読み込まないと表示されない。もっとも、映像ファイルからの音声の抽出はごく簡単な手順でできるので [8)]、いずれにせよとくに大きな問題はない。

図 5a　Subbu (0.3.0) の作業画面（波形表示）

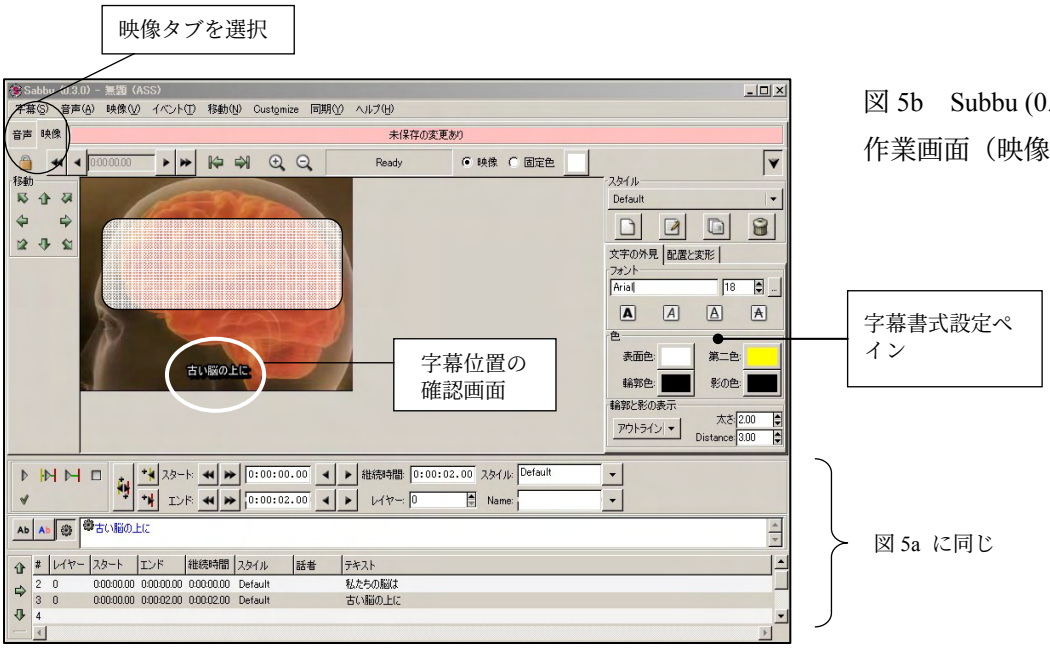


図 5b　Subbu (0.3.0) の作業画面（映像表示）

入手先：

- http://sourceforge.net/projects/sabbu/
- http://sourceforge.net/project/showfiles.php?group_id=123405

3.3　ChronoSub (7.7.1)

ChronoSub (7.1.1, 2005-6-25 release) は前出の Subbu と同じく、映像表示と音声波形表示の両方ができる字幕エディタである。ただし、**図 6** に示すとおり、こちらは一枚の画面に映像と音声波形および字幕編集ペインがコンパクトにまとめられており、新規ユーザにはよりフレンドリーな感じを与えている。ただし、これも操作法を解説したヘルプファイルが付いていないため、最初のうちは多少の遅行錯誤が必要になってくるものと思われる。

対応している映像フォーマットは avi, mp4, mkv [9] などで、音声は wav 形式のほかに mp3, aac, ac3 などの圧縮ファイルにも対応している（ただし、今回確認したのは avi と wav のみ）。字幕ファイルは srt, ssa, ass および usf (Universal Subtitle Format) という独自の形式に対応している。なお、usf 形式というのは XML (EXtended Markup Language) をベースにしたもので、通常の字幕ファイルに比べて冗長になり、字幕エンジン側での処理が難しくなるという欠点があるため、一般にはほとんど採用されていない形式である。

図 6a　ChronoSub (7.7.1) の作業画面 1

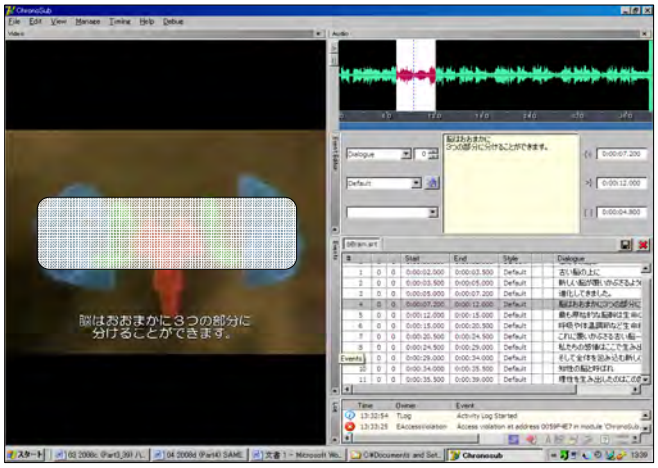


図 6b　ChronoSub の作業画面 2：映像パネルを大きく表示した状態

図 6a は動画ファイルを読み込んだときのデフォルトの作業画面を示す。動画は音声と連動して動くようになっており、音声波形パネルの左端に小さく表示されている > 記号をクリックすると選択箇所を再生し、|| 記号クリックすると一時停止する（ただし、ショートカットキーを独自にアサインして使うことができる）。映像および字幕は音声波形と同期して再生されるようになっているため、通常のメディアプレーヤのように映像だけを再生コントロールすることはできない。波形表示ペインと映像表示ペインはマウスを使ってドラッグすることで上下左右に自由に移動させることができるため、必要に応じて**図 6b** のように映像ペインを大きく表示することもできる。

なお、映像を見ながら字幕のタイミングを計測する場合、どうしてもだいたいの見当でタイミングを決めることになるが、この場合、「ここ！」と思ったところですばやく停止ボタンを押しても、100 ミリセコンド単位でずれが生じるのは避けられない（p. 5 参照）。一方、音声波形を見ながら字幕のタイミングを決める場合は、発話の開始点と終了点（および継続時間）がミリセコンド単位で正確に測定できるという利点がある（波形を見ればどの時点から発話が開始されたかが一目瞭然であり、映像だけの場合のようにズレが生じることもない）。

プログラムの入手先は以下のとおりであるが、ファイル（zip 形式の圧縮ファイル）をダウンロードしたあと、これを展開すると [ChronoSub_2060625] というフォルダが作成され、この中に ChronoSub.exe というプログラムファイルが展開される。wav 音声の抽出は同じく展開される WavWriter.dll という拡張プログラムを使って行われる。なお、[ChronoSub_2060625] フォルダは、展開後どこにおいてもかまわない。

入手先：

- http://sourceforge.net/project/showfiles.php?group_id=104310

3.4 DivXLand Media Subtitler (ver. 2.0.5)

DivXLand Media Subtitler（以下 Media Subtitler）は、すでに染谷 (2008c, pp. 14-21) でも取り上げたとおり、フリーウェアの字幕エディタとしてもっとも使い勝手のよいもののひとつで、2007 年 1 月にリリースされた最新版（ver. 2.0.5）では、通常の字幕編集のほかに、ハードサブの作成や、動画からの音声の抽出等の機能がサポートされている。使用に当たっては VirtualDab（動画編集用プラグイン）および BeSweet（音声抽出用プラグイン）をあらかじめインストールしておく必要があるが、これについては染谷（同上）を参照していただきたい。

対応している動画形式は wmv. asf, mpg (mepg1, mpeg2), mp4, avi, divx, xvid, mkv, ogm[10] と多様で、字幕ファイル形式も smi, srt, ssa, sub などおよそ 30 種類のフォーマットに対応している（ただし、ハードサブの作成に関しては入力時の動画ファイルは avi と mpeg-1 形式、出力は avi 形式のみ。字幕ファイルは ssa [= Sub Station Alpha] 形式に変換してからエンコードする）。

図 7 に作業画面例を示す。この画面でもわかるとおり、前述の ChronoSub と同じく、Media Subtitler では同一画面上で音声波形と映像を同時に表示しながら字幕を編集することができるため、映像のみを見ながら作業する場合と比べてより厳密なタイミングの指定が可能になって

いる。なお、ChronoSub では映像のみの再生コントロールはできなかったが、Media Subtitler の映像パネルには WMP がそのまま埋め込まれているため、WMP と同等の映像コントロールが可能になっている。したがって、WMP の画面から右クリックして表示されるメニューの機能（例えば全画面表示、ファイルマーカー、再生速度コントロール等）は、すべて通常どおり使うことができる。

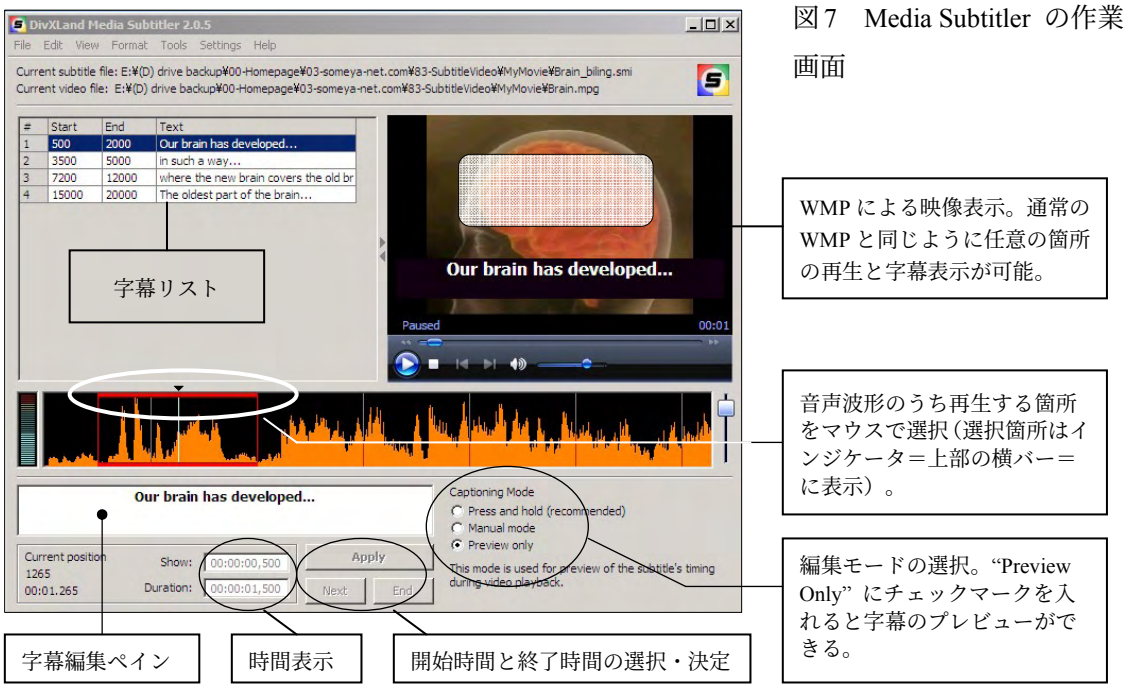


図 7　Media Subtitler の作業画面

字幕ファイル作成の基本的な操作は次のとおり。①まず、映像ファイルを読み込む。②次に、すでに用意した字幕データ（通常のテキストファイルでもよい）があればこれを読み込み ([File]→[Open Text or Subtitle])、③さらに別途抽出した音声ファイル（[Tool]→[Audio extraction wizard]）を読み込んだ上で映像を再生。④映像上に同期表示させながら、画面下部の編集パネル上でタイミングの調整や字幕内容の編集を行う。⑤編集作業が終了したらメニューから [File]→[Save As] を選択し、字幕のフォーマットを指定して保存（既存のファイルを上書き保存する場合は [File]→[Save] を選択）。上記②で、新たに字幕ファイルを作成する場合は、メニューから [File]→[New Subtitle] を選び、③に進む。なお、④では字幕編集モード (Captioning Mode) として自動と手動のいずれかを選択するが、詳しくは付属のマニュアル（[Help]→[View the User’s Guide]）を参照されたい。

入手先※：

- http://www.divxland.org/subtitler.php

※ 詳しくは染谷（2008c, p.14 参照）

3.5 SubStation Alpha (v4.08)

SubStation Alpha (v4.08) は音声波形を見ながら字幕編集ができる字幕エディタとして、いわば老舗のフリーウェアで、Windows 3.1 の時代から多くのユーザに愛用されてきたものである。映像表示機能はなく、もっぱら音声波形を見ながら字幕のタイミングを決めていくようになっているが（**図 8** 参照）、すでに述べたとおり、タイミングの測定には映像は必要なく、むしろ波形を見ながら決めていくほうがより正確な作業ができる。したがって、この点はとくに問題にはならない。

対応している字幕ファイル形式は ssa 形式のみ（ssa 形式はもともと SubStation Alpha 専用のデータ形式として開発されたもの）。ただし、あらかじめ Plain Text として作成した字幕データ（一行ごとに改行したもの）をそのまま読み込むことができる（[File]→[Open]→[Plain Text]）。日本語を使う場合は、初期状態のままだと文字化けをするので、メニューから [Options]→[Preferences]→[Display Select Grid Font] を選択し、「フォント」を MS ゴシックに、「文字セット」を日本語にそれぞれ設定しておく。フォントサイズは 10 ポイント程度でよい。

図 8 SubStation Alpha の作業画面

作業画面もシンプルで比較的使いやすく、詳細なマニュアル（英文）も付属しているため、初心者でもとくに大きな問題なく操作法を習得することができる。また、jpg, gif, bmp などの静止画像の挿入や、他の wav ファイルや avi ファイルへのリンクを “Non-text Events” として埋め込むことができたり、DirectDraw（グラフィックレンダリング用の DirectX API）を利用して

字幕をスクロール表示させたり、カラオケ風にハイライトの位置をずらしていくといったこともできる。こういった機能を実装している字幕エディタは数えるほどしかなく、SubStation Alpha の大きな特徴のひとつとなっている。ただし、SubStation Alpha を選択するに当たって、いくつか問題がないわけではない。

まず、最終的に作成される字幕ファイルの形式が ssa 形式に限られていることは、（本稿を含む一連の関連論文で SAMI 形式を前提として議論してきたという立場からすれば）大きな不満のひとつである。ssa 形式は、SAMI に比べて仕様が複雑で、一般的なテキストエディタで直接編集するのに適さないからである。もっとも、前述のとおり、Subtitle Workshop を使えば ssa 形式の字幕ファイルを簡単に SAMI 形式に変換できるため、この点はかろうじて見逃すことができる。

もうひとつの問題点は、SubStation Alpha では一般的な wav ファイルは使えず、“8 bit PCM” 形式のモノラル wav サウンドでなければならない、という制約がある点である。さらに、wav ファイルを読み込む際に末尾の部分が切れてしまうことがあるため、あらかじめファイルの末尾に数十秒から数分（ファイルの大きさによる）ほどの無音部分を挿入しておく必要がある。こうした作業は音声ファイル編集ソフトがあればごく簡単にできるものであるが [11)]、初心者には余計な負担が増えることになる。なを、このソフトウェアは現在は開発が中止されており、今後の更新やバグの修正は望めない。

字幕エディタとして優れた機能と操作性を備える SubStation Alpha ではあるが、導入に当たってはこれらの点を十分に考慮に入れておくべきであろう。

入手先：

- http://www.videohelp.com/tools/Sub_Station_Alpha

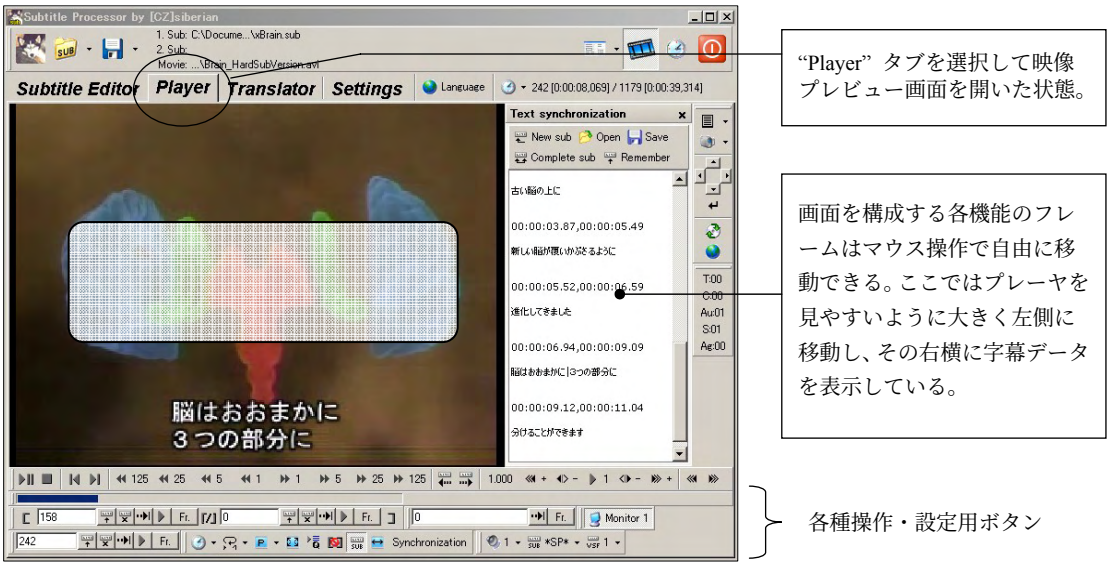
3.6　Subtitle Processor (7.7.1)

Subtitle Processor (7.7.1) は、後述の Subtitle Workshop と同じく、フリーウェアとしてはたいへん充実した機能を備えた字幕エディタである。ユーザインターフェイスは、**図 9a** および**図 9b** に見られるとおり、「字幕エディタ (Subtitle Editor)」「プレーヤ (Player)」「翻訳 (Translator)」および「プログラム設定 (Settings)」の４つのページからなり、それぞれのページは該当するタブをクリックすることで開くことができる。

「字幕エディタ」画面（**図 9a**）は、上段に字幕各行の開始時間と終了時間が入力された字幕リストがあり、中段に字幕編集ペイン、その右側に編集中の字幕のプレビューウィンドウ、さらに下段に字幕の拡張編集用エリアがある、という構成である。字幕の表示開始ポイントと終了ポイントは、フレームまたは時間単位のいずれか任意の方法で指定が可能。また、作業箇所の字幕は、右側のプレビュー画面でフォントなどの書式を確認しながら編集することができる。字幕の拡張編集用エリアにはさまざまなオプションが用意されており、たとえば字幕のプロパティ設定、テクストおよびタイミングの補正、スペルチェッカー設定（ただし辞書は別途追加）、字幕のトランジション（フェード効果やスクロールなどの字幕表示効果）設定などを行うことができる。

図 9a　Subtitle Processor の作業画面 1：字幕エディタ画面

図 9b　Subtitle Processor の作業画面 2：プレーヤ画面



「プレーヤ」画面（**図 9b**）では、作成した字幕を実際に映像に同期させて表示し、適切にタイミング設定が行われているかをチェックすることができる。なお、画面を構成する各機能のフレームは必要なものだけを表示することができ、またマウス操作で自由にその位置を移動することができる。**図 9a** では、プレーヤを見やすいように大きく左側に配置し、その右横に字幕データを表示している。動画形式は wmv, avi, mpeg (mpg), ogm, mkv などをサポートしている。字幕表示のタイミングは 100 ミリセコンド単位で指定することができる。ただし、音声波形表示機能がないため、正確なタイミングの計測にはやや技術を要する。

Subtitle Processor にはさらに「翻訳」用の画面があり、ここに第 1 字幕（オリジナル字幕）

と第 2 字幕（翻訳字幕）を読み込んで、両者を比較しながら翻訳作業をすることができる。なお、字幕データは第 1 字幕と第 2 字幕を連動してスクロールさせることができるため、効率的に作業を行うことができる。

「プログラム設定」ページは、文字通りプログラム使用に当たっての基本的な設定をするところである。ここでは、たとえばユーザインターフェイスおよびツールバーの設定、字幕ファイルとの関連付け、字幕フォーマットの各種オプション設定、VobSub との連動、時間表示のモード設定、映像表示に関する設定、ショートカットキー設定等々の諸設定を行うことができる。いくつかの項目については多少の専門的な知識が必要であり、必ずしも user-friendly なものにはなっていないが、オンラインでの解説資料が用意されており、これを解読する手間を惜しまなければ、それなりに対応することは可能である。

なお、字幕の編集が終わればこれを保存することになるが、Subtitle Processor で出力可能な字幕ファイル形式は sub, srt, ssa, ass, txt, dks, scr, mpl, pjs, psb となっている [12)]。

日本語化および初期画面からの操作について

インストール用ファイル（zip 形式の圧縮ファイル）を以下に示す入手先からダウンロードしたあと、これを解凍すると [SubtitleProcessor771] というフォルダがデスクトップに作成される（このフォルダは解凍後、任意の場所に置いておくことができる）。通常はこの中の [lang] というサブフォルダに日本語化用の言語ファイル (Japanese.lng) があらかじめ収録されている。したがって、プログラムを起動した後、メインタブ領域の右側にある [Language] ボタンをクリックし、一覧表されるリストの中から Japanese.lng を選択すれば、そのまま（再立ち上げせずに）画面表示全体が日本語化される。もし、上記フォルダ内に日本語化用言語ファイルがない場合には、以下に示すサイトからダウンロードすることができる。

なお、プログラムの立ち上げは前記の [SubtitleProcessor771] フォルダ内にある Subtitle Processor.exe というプログラムファイルをダブルクリックすればよいが、初めて立ち上げたときに、どこから開始すればよいか迷うことがある。プログラムに慣れるにはとにかくまずは動かしてみることが大事だが、入り口でつまずくことがないよう、参考までに「メインツールバー」と「メインタブ」領域、およびここから映像ファイルと字幕ファイルを読み込むための手順を**図 10** に示す。

以上、Subtitle Processor についてごく簡単に解説した。実に多彩な機能を搭載するエディタであることは間違いないが、難を言えば、高機能であるがゆえに、使い方をきちんと覚えるのにかなり手間隙がかかるという点だろう。直感的な操作だけでは対応できないところも多く、これを使いこなすためには、前述のとおり必ずしも読みやすくはないオンライン資料（マニュアル）をきちんと読むなど、それなりの努力と忍耐を必要とする。授業で使用することを考えると、この点はかなり重要である。

図 10　Subtitle Processor の「メインツールバー」と「メインタブ」領域

入手先：

- http://sourceforge.net/projects/subtitleproc/
- http://sourceforge.net/project/showfiles.php?group_id=158915

［オンライン資料（マニュアル）］

- http://subtitleproc.sourceforge.net/help/_index.htm

［日本語化用の言語ファイル］

- http://sourceforge.net/project/showfiles.php?group_id=158915&package_id=179432

3.7　Subtitle Workshop (Ver. 2.51)

Subtitle Workshop は、すでに前稿（染谷 2008a, p. 20; 染谷 2008c, 注 2, 注 8）でもたびたび言及してきたとおり、高機能字幕エディタとして定評のあるフリーウェアで、現在、筆者の勤務校での授業でも使用されているものである。前出の Subtitle Processor や Media Subtitler と同じように、動画をプレビュー表示しながらリアルタイムに字幕テクストおよびタイミングを編集画面上に入力していくため、効率的な字幕作成ができるようになっている。

対応する動画形式は avi, mpg, wmv, mp4 など主要なものをカバーし、字幕形式についても 50 種類を超えるフォーマットに対応している。これは、現在公表されている字幕フォーマットのほぼすべてを網羅するものである。また、英語、日本語のほか、フランス語や中国語などさまざまな言語に対応したローカリゼーション用言語ファイルが付属しているため、メニュー表示を必要に応じてほぼ任意の言語に変更することができる（日本の場合はメニューから「設定」→「言語」→「Japanese」を選択）。

図 11 に Subtitle Workshop の標準的な作業画面例を示す。これは、プログラムを立ち上げた後、「翻訳モード」にして和文字幕と英文字幕を並べて表示した画面である。画面構成は、上段にプレビュー画面、中段に字幕リスト、下段に字幕編集エリアとなっている。

図 11　Subtitle Workshop の作業画面：翻訳モードにして和文字幕と英文字幕を並べて表示

なお、上記のような作業画面に進む前に、あらかじめメニューから「設定」→「基本設定」を選択し、必要最小限の環境設定を行っておく必要がある。本稿では環境設定の各項目についてひとつひとつ説明はしないが（詳しくは「ヘルプ」をクリックすると開くマニュアルを参照）、日本語を使う場合は必ず「文字コード」に Shiftjis を選択しておく必要がある（**図 12**）。「翻訳モード」で使用する場合は、オリジナルと翻訳の両方について文字コードを指定するが、日英の組み合わせの場合はいずれも Shiftjis でかまわない。なお、文字コード設定画面で「メイン画面に文字コードを表示する」をチェックしておくと、メイン画面（の左端＝**図 12**）に、現在選択されてる文字コードが常に表示されるようになるので、わかりやすい。このほか、「再生環境」→「字幕」のところでも、字幕の基本的な書式設定およびフォント指定を行っておくこと（ここでは、プレビュー画面に表示される字幕の設定を行う）。

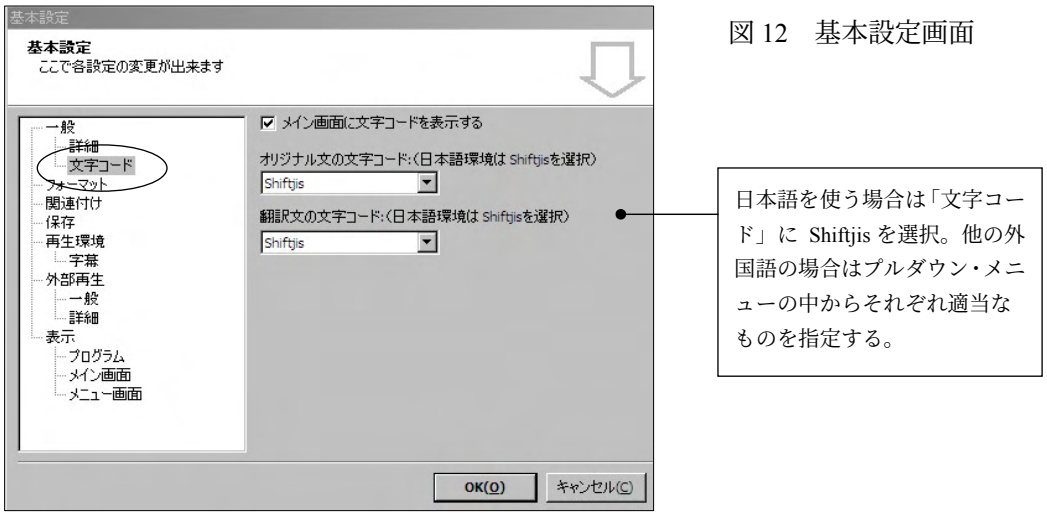


図 12　基本設定画面

Subtitle Workshop は頻繁に更新されており、そのたびに新しい機能が追加されてきている。たとえば、現在の安定バージョン（Ver. 2.51［新たに Version 4, BETA 4 が出ているが、これについては後述する］）では、他のさまざまな機能に加えて、次のような機能を利用することができる。

- 再生速度のコントロール
- テクスト検索および一括置換機能
- ファイルからのテキストデータの読み込み
- スペリングチェック
- マルチメディア拡張ファイルでの保存

このうち、動画の再生速度のコントロール機能（メニューから「映像」→「再生速度」を選択）は、10% から 100% まで 10% 刻みで再生速度を遅くすることができるというもので、タイミングの決定にはたいへん便利な機能である。これは、波形表示機能がないという Subtitle Workshop の欠点を補う機能ということができる。実際の操作では、オリジナルの 80% 程度の速度にすると、発話の開始点および終了点に合わせて映像を一時停止させる操作がかなりやりやすくなる。ただし、現バージョンでは wmv の再生速度変更はできないようである（mpg と avi では確認済み）。このあたりは次のバージョンアップに期待するということになるだろう。なお、動画の再生／一時停止の操作は画面上の操作ボタンでも行うことができるが、キーボードショートカットを使うと操作をより効率化することができる。

スペリングチェックは Microsoft Word の機能を利用しているため、パソコンに MS Word 97 またはそれ以降のバージョンがインストールされている必要がある。したがって、対応する言語は MS Word に辞書が組み込まれている言語（デフォルトでは英語）のみである。

「マルチメディア拡張ファイルでの保存」というのは従来の Subtitle Workshop にはなかった機能で、字幕編集終了後、メニューから「映像」→「マルチメディア拡張ファイルで保存」→「ASX (Microsoft Media Player)」を選択すると、ASX ファイル（本稿付録, pp. 23-24 参照）を自動生成してくれる、というものである。**図 13** の例に示すとおり、デフォルトではアクセス先が SAMI URL 構文（染谷 2008b, p.4）を使ったローカルパスで記述される。このファイルをそのままダブルクリックすれば、作成したばかりの字幕付きの動画を WMP で立ち上げることができる。

```
<ASX version ="3.0">
<Title>Sample1</Title>
<Entry>
<Ref href="C:¥Documents and Settings¥Administrator¥MyDocuments
¥MyMovie¥Sample1.wmv?sami=C:¥Documents and Settings¥Administrator
¥MyDocuments¥MyMovie¥Sample1.smi" />
</Entry>
</ASX>
```

図 13　Subtitle Workshop で自動生成される ASX ファイルの例

なお、通常の保存は、メニューから「ファイル」→「保存」→「オリジナルファイル新規保存」（または「翻訳ファイル新規保存」）を選択して保存する。ただし、既存のファイルに上書き保存をする場合は「オリジナルファイル」（または「翻訳ファイル」）を選択する。**図 13** に、新規保存時に表示される保存フォーマットの選択画面を示す（この例では SAMI 形式を選択）。

図 13　新規保存時のフォーマット選択画面※

※染谷 (2008c, 注 8 の図 39b) に同じ。

<u>Version 4, BETA 4 の新機能：字幕の自動翻訳</u>

前述のとおり、Subtitle Workshop は頻繁にバージョンの更新が行われており、この稿を書いている現時点では、Version 4 (BETA 4) が公開されている。この最新版の最大の特徴は新たに「自動翻訳」の機能が加わっていることである。この機能は、おそらくインターネット経由でグーグルの翻訳エンジンを利用しているものと思われ、したがって翻訳精度もそのレベルのものであるが、工夫次第では教育的な目的に使うことができる可能性がある。参考までに、**図 14a** に字幕自動翻訳の設定画面を、**図 14b** に翻訳結果の表示画面例をそれぞれ示す。

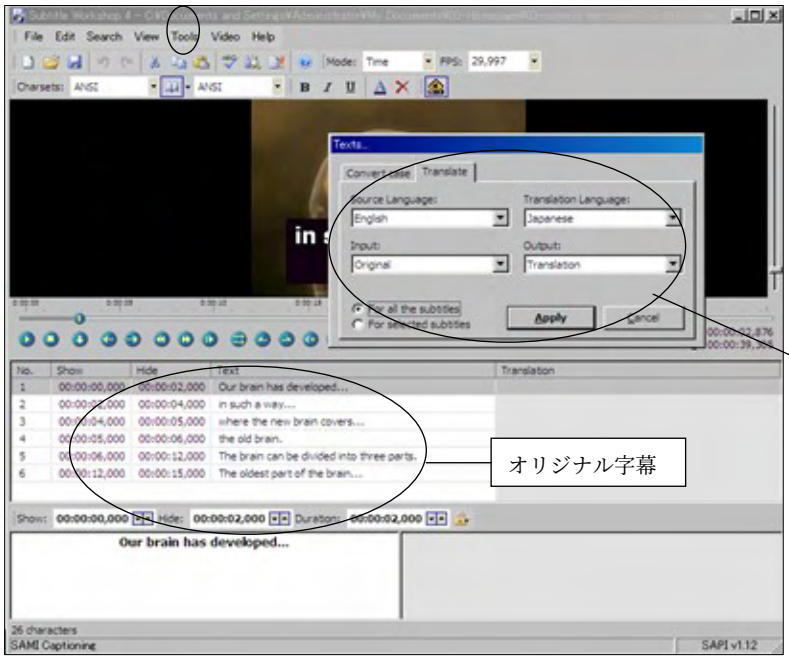


図 14a　字幕自動翻訳の設定画面 (Version 4, BETA 4)

作成済みの字幕ファイルを読み込んだあと、メニューから [Tool] → [Translator] を選択。図の画面でオリジナル字幕の言語と翻訳言語を指定して [Apply] ボタンをクリック。

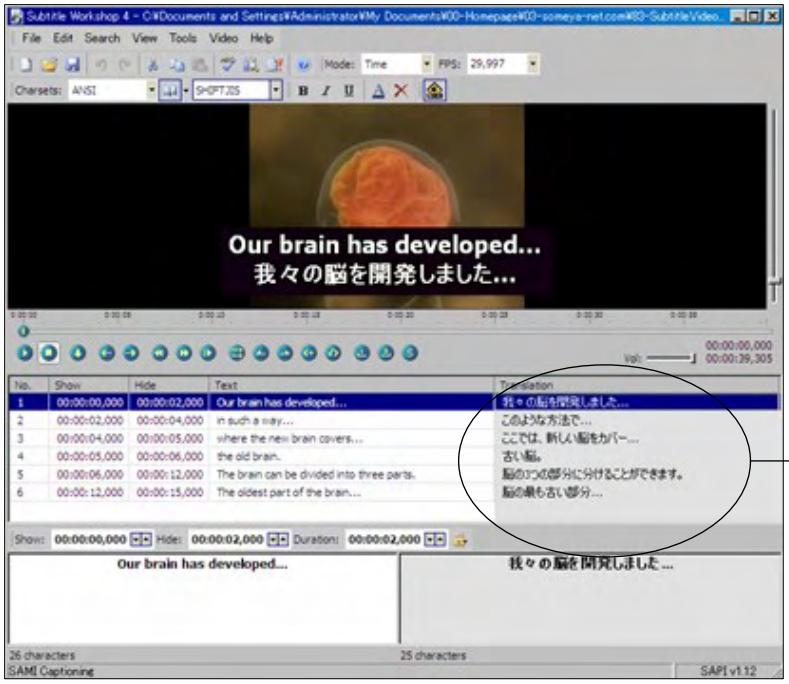


図 14b　自動翻訳機能を使った翻訳結果の表示画面例 (Version 4, BETA 4)

翻訳結果の表示。翻訳精度は満足できるものではないが、意味の明瞭な名詞句や定型的なフレーズは比較的正確に訳されていることがわかる。

表 1 は、**図 14b** に例示した自動翻訳結果をわかりやすいように表にしたものである。これを見ると、翻訳精度は決して満足できるものとは言えないが、意味の明瞭な名詞句 (the old brain; The oldest part of the brain) や定型的なフレーズ (in such a way) については、その範囲内でほぼ正確に訳されている。ただし、格助詞の使い方に典型的に表れているとおり、文要素の関係（格関係や修飾関係）を正しく判断した上で適格な文を生成するというところまではいかないようである。もっとも、これは現在の自動翻訳技術の水準からして、ごく当然のことと言うべきであろう。

	原文	翻訳文
01	Our brain has developed	我々の脳を開発しました
02	in such a way	このような方法で
03	where the new brain covers	ここでは、新しい脳をカバー
04	the old brain.	古い脳。
05	The brain can be divided into three parts.	脳の 3 つの部分に分けることができます。
06	The oldest part of the brain	脳の最も古い部分

表 1　Subtitle Workshop (Version 4, BETA 4) による字幕文自動翻訳の結果

なお、Version 4 は現在のところ開発途上のベータ版であり、導入はしばらく控えたほうがよい。今後のさらなる改善を期待しつつ、当面の間は現時点での安定版である Version 2.51 を使うことを勧めたい。

入手先：

- http://www.urusoft.net/

4. まとめ

以上、本稿では字幕ファイルの作成編集を主要なテーマに、まず第 1 章で通常の文書作成用エディタを使った字幕ファイルの作成方法について述べ、続いて第 2 章で「Windows Media ファイルエディタ」を使ったミリセコンド単位でのタイミング計測について述べた。これは、第 1 章での議論を補足するものである。第 3 章では、より正確かつ効率的な字幕作成および字幕翻訳を支援するためのツールとして、現在入手可能なフリーウェアの中から SubCreator, Subbu, ChronoSub, Media Subtitler, SubStation Alpha, Subtitle Processor, Subtitle Workshop という 7 つの字幕エディタをとりあげ、それぞれ概要を解説した。**表 2** に、本稿で紹介した字幕エディタの機能、および対応状況を比較した一覧表を示す。

	映像表示	対応動画形式	WAV 波形	ハードサブプラグイン	ﾃｷｽﾄﾃﾞｰﾀの読み込み	対応字幕ファイル形式	機能の充実度	使いやすさ
SubCreator	○	wmv, avi, mpg	×	×	○	srt , ssa	--	優
Subbu	○	avi	○	×	○	srt, ssa, ass	--	良
ChronoSub	○	avi, mp4, mkv, etc.	○	×	○	srt , ssa, ass, usf	--	良
Media Subtitler	○	wmv, avi, mpg, etc.	○	○	○	多種対応	○	良
SubStation Alpha	×	--	○	--	○	ssa	○	可
Subtitle Processor	○	wmv, avi, mpg, etc	×	×	○	多種対応	◎	やや難
Subtitle Workshop	○	wmv, avi, mpg, etc.	×	×	○	多種対応	◎	良

表 2　本稿で紹介した字幕エディタの機能および対応状況比較表

この一覧表からどれが最も優れた字幕エディタかを判断するのはなかなか難しい。誰が、何を主な目的として使うか、ということが重要な判断要素になるからである。「プロ仕様」の仕事ができるほど充実した機能を備えているものであっても、その操作を覚えるのに大きな努力と時間を要するものは、授業で使用するツールとしては不適当であると言わざるを得ない。とはいえ、使い方は簡単でも、ユーザが（あるいは、指導者がその教育目的上）必要とする機能が備わっていないのでは、そもそも使う理由がない。本稿で取り上げた字幕エディタは、それぞれ一定の基準を満たしており、本稿ではどれかひとつに絞って一律にその使用を推薦することは避けたいと考える。ただし、強いて結論めいたことを述べるとすれば、簡便性を重視する場合は SubCreator、音声波形を重視する場合は Subbu または ChronoSub、映像を見ながらのビジュアルな操作性に加えて、ハードサブの作成機能がある点をとくに評価する場合は Media Subtitler、字幕エディタとしての総合的な機能と操作性という点では Subtitle Workshop をそれぞれ採用する、ということになるだろう。

「字幕付き動画作成のための SAMI ファイルの作成・編集について」（染谷 2008d）付録

「Windows Media ファイルエディタ」を使った動画ファイルの編集について

本稿の第 2 章第 3 節 (p. 5) で、「Windows Media ファイルエディタ (Ver. 9)」（以下、WMEd）を使った動画ファイルの編集について言及した。以下、その手順を具体的に解説する。なお、本文中 (p. 3) でも述べたとおり、WMEd は「Windows Media エンコーダ 9」をインストールすることで、そのユーティリティツールのひとつとして自動的にインストールされる。ここではすでにパソコン上に WMEd がインストールされていることを前提に話を進める。

1. 「Windows Media ファイルエディタ」を使ってできること

WMEd を使うと、Windows Media 形式のファイル (wmv, wma) に対して、①メディアファイルの切り詰め編集、②再生位置マーカーの挿入、③スクリプトコマンドの挿入、④コンテンツに関する表示情報の追加、等の編集作業を行うことができる。このうち、③は指定した時間にメッセージを表示したり、他のウェブページや文書にジャンプさせたりするという機能であるが、本稿での目的に照らして有用性が少ないことから、ここでは取り上げない。また④は一目瞭然の内容であり、特に説明を要さない。したがって、ここでは①と②についてのみ解説する。

2. 動画ファイルの切り詰め編集

WMEd の「切り詰め編集」機能は、Windows Media 形式のファイルの最初と最後の部分をカットして任意の長さに編集するというもので、これを使えば、例えば 60 分の動画ファイルをいくつかの小さなファイルに分割して（個人またはグループ）プロジェクトの課題として与えたり、動画ファイルの中から必要な部分だけを取り出して授業の素材に使う、といったことが簡単にできるようになる。作業手順は以下のとおり。

1. WMEd を起動（本文の**図 2**, p. 4 参照）。
2. 起動後、「マーカー」タブをクリックした上で、メニューから「ファイル」→「開く」を選んで目的の動画ファイルを開く（以下、次ページの**図 15** 参照）。
3. 動画が表示されたら、[再生] ボタンをクリックし、新規ファイルの開始点としたい位置で映像を一時停止し、「開始マーク」ボタンをクリック。
4. 再度、再生ボタンをクリックし、新規ファイルの終了点としたい位置で映像を一時停止し、「終了マーク」ボタンをクリック。
5. 位置を変更したい場合は上記の 3, 4 を繰り返す。

図 15　切り詰め開始点（開始マーク）と終了点（終了マーク）の挿入

7. 開始点と終了点が決まったら、メニューから「ファイル」→「名前を付けて保存およびインデクス作成」を選び、新規ファイルに任意の名称を付けた上で適当なフォルダ内に保存。

以上で、動画ファイルの切り詰め編集は終了。なお、新規ファイルはカットされた部分が削除された上で保存されるので、必ず元のファイルとは異なった名称で保存すること（同じ名称にすると元のファイルに上書き保存されてしまうので、くれぐれも注意！）。それぞれのビデオクリップにタイトルや作者名、著作権情報等を加えたい場合は、各ファイルを保存する前に上記画面から「属性」タブをクリックすると、これらの情報の入力画面になるので、必要に応じて属性情報を加えることができる（保存したあとでも可。また、いつでも任意に修正・変更することができる）。

2.1　分割したファイル（ビデオクリップ）を連続して再生するための方法

いくつかに分割したビデオクリップを再生する場合（たとえば、グループプロジェクトでそれぞれ完成したものをまとめて試写する場合等）、ひとつひとつ手動でファイルを立ち上げるのは面倒であり、動画としての連続性も失われてしまう。このような場合、複数のビデオクリップを物理的にひとつのものに結合してしまう（たとえば「Windows ムービメーカ」を使って結合する）方法もあるが、もうひとつの方法として、「ASX ファイル」を作成して複数のビデオクリップを連続して配信再生することもできる。この方法は簡便な上に、個々のビデオクリップはそのまま独立した状態にあるため、いつでも任意に修正を加えることができるという利点がある。

1) 「ASX ファイル」の作成

「ASX (Advanced Stream Redirector) ファイル」とは、再生対象となるメディアファイルに関する情報を記載したもので、拡張子を `.asx` とする通常のテキストファイルとして作成する（フ

ファイルについての情報を記載したファイルなので、「メタファイル」と呼ばれる）。複数のビデオクリップを連続再生するための ASX ファイルの最もシンプルな記載例を**図 16** に示す（各行冒頭の数字は説明のもので、実際のファイルには記入しない）。

```
1  <ASX Version="3.0">  ←ヘッダ（ASX ファイルであることの宣言）
2    <entry><ref href="MyMovie/Part01.wmv" /></entry>
3    <entry><ref href="MyMovie/Part02.wmv" /></entry>
4    <entry><ref href="MyMovie/Part03.wmv" /></entry>
-    <!-- 以下、必要な数だけ <entry>〜</entry> の記述を追加 -->
5  </ASX>  ← ASX ファイル終了タグ
```

図 16　ASX ファイルの最もシンプルな記述例

このうち、Lines 2〜4 の `href=` の値には実際のファイル名（と必要に応じてパス名）を記入する。この例では ASX ファイルの置いてあるローカルフォルダの直下にある MyMovie というサブフォルダにビデオクリップが保存されているという想定である。ビデオクリップが ASX ファイルと同じフォルダにある場合は `href="Part01.wmv"` という指定になる。ただし、インターネット経由で配信する場合は `http://` から始まる URL のフルアドレスを記入しなければならない（例：`"http://www.dummy.com/MyMovie/Part01.wmv"`）。

ASX の実質的な中身を構成する `<entry>〜</entry>` の各行は、ビデオクリップの数に応じていくつでも追加することができる。また、それぞれの `<entry>` について

`<title>`ビデオクリップのタイトル情報`</title>`

`<author>`作成者情報`</author>`

`<copyright>`著作権情報`</copyright>`

などの各情報を、オプションとして、上記に例示した形式で任意に追加することができる。記入場所は `<entry>〜</entry>` の間で、通常は <ref> タグの前に行を分けて記入する。その他のオプションについては Microsoft Corp. (2000) を参照されたい。ただし、いろいろと加えるとそれだけ記載エラーが出てくる可能性が高くなるので、とくに必要でない限りできるだけシンプルな内容にしておくほうがよい。

なお、Microsoft Corp. (2000) に記載されている技術情報によれば、`wmv` 形式の動画ファイルにアクセスする場合のメタファイルの拡張子は`.wvx` となっているが、`.asx` のままでも特に問題はない（変更する場合は単に`.asx` を`.wvx` に書き換えるだけでよい）。

上記のような内容の ASX ファイルを作成したあと、これを任意のローカルフォルダ（通常はビデオクリップのあるフォルダ）に保存する。保存ファイル名は拡張子が`.asx`（または`.wvx`）であればどのような名称でもよい。ここでは仮に `SubtitleProject2008.asx` としておく。

2)　HTML ページの作成

次に、この ASX ファイルを呼び出すための html ページ（ウェブページ）を作成する。これも名称は任意のものでよいが、ここでは前記 ASX ファイルのファイル名にそろえて、仮に SubtitleProject2008.html としておく。記載例は以下のとおり（**図 17**）。

```
01  <!doctype html public "-//w3c//dtd html 4.0 transitional//en">
02  <html>
03    <head>
04    <meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
05    <title>Subtitle Project 2008</title>
06  </head>
07  <body>
08  <hr width=0 align=center>
09  <p style="text-align:center; line-height:120%; font-size:18pt; font-family:
10  Arial, Verdana, Helvetica,'ＭＳ ゴシック', sans-serif; color: blue;">
11  Welcome to the online version of<br>"Subtitle Project 2008"<br>
12  Click <a href="SubtitleProject2008.asx">here</a> to begin the video.</p>
13  <hr width=300 align=center>
14  <p style="text-align:center; line-height:120%; font-size:9pt; font-family:
15  Arial, Verdana, Helvetica,'ＭＳ ゴシック',sans-serif; color:black;">
16  Copyright 2008 by the Class of 2008, <br>English Department, ABC University, Tokyo.
17  </p></body>
18  </html>
```

図 17　ASX ファイルを呼び出すための HTML ページの記述例

この例では、単に ASX ファイルを呼び出すだけの目的であるため、修飾的な要素は排除して、ごくシンプルなものとした。Line 8 の <hr> は横線 (horizontal line) を引くためのタグであるが、ここでは width=0 として外観上の実態のない横線を引き（＝1 行分の空白行と同じ）、ページの上枠と本文の間にスペースを挿入した。Line 9 では、パラグラフ要素 <p> にスタイル属性を追加し、このあとに続く本文の書式を設定している。スタイル設定の内容についてはすでに染谷 (2000a, pp. 4-7 および**注 8**) で詳しく解説した。

本文は Lines 11-12 の 2 行のみである。Line 12 では <a> (anchor) 要素を使って目的の ASX ファイル (SubtitleProject2008.asx) にリンクを貼っている。アンカー要素は、このように href 属性と組合わせて特定の文字列や画像にリンクを貼ったり、name 属性と組み合わせてページ

内の指定箇所を参照する終点アンカーを作成するもので、HTML の最も基本的なタグのひとつである。この例では “here” という文字列にリンクが貼られており、ウェブページ上でこの文字列をクリックすると、メタファイル（SubtitleProject2008.asx）を経由して、目的のメディアファイルが呼び出され、ローカルの WMP 上でファイルが再生されることになる。もちろん、これまでの例と同じく、動画ファイルと同じフォルダ内に対応する字幕ファイル（SAMI ファイル）が置いてあれば、字幕データが自動的に読み込まれ、WMP の字幕ペインに同期表示されることになる。**図 18** に、この HTML ページをブラウザで立ち上げた時の画面、および WMP 上でのビデオクリップの画面イメージを示す。

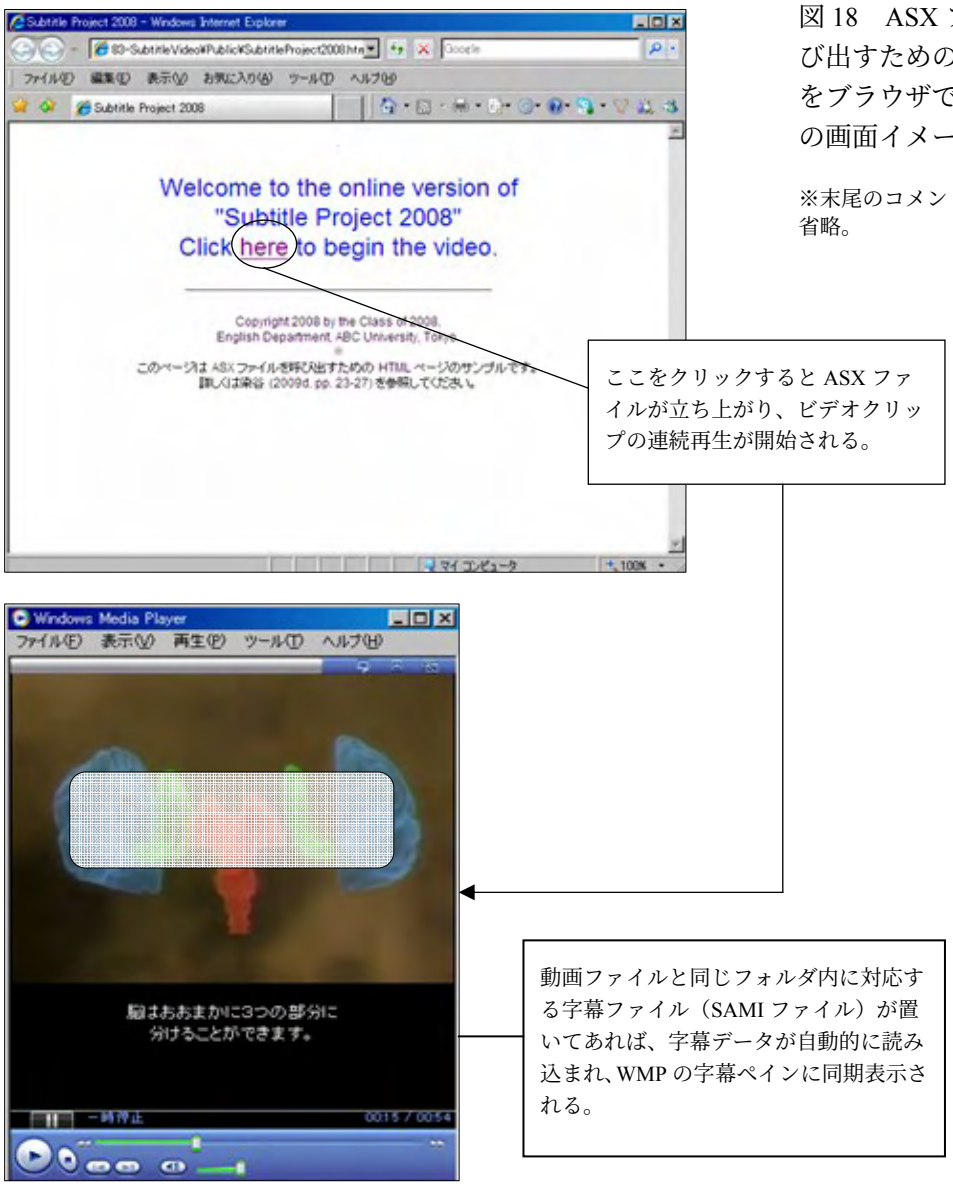


図 18　ASX ファイルを呼び出すための HTML ページをブラウザで立ち上げた時の画面イメージ

※末尾のコメント行は図 17 では省略。

以上、WMEd を使った分割したビデオクリップを、（字幕付け等の作業が終了した後）WMP 上で連続して一括再生するための方法について解説した。ここで紹介した例では、ローカルの

WMP を単独で立ち上げてビデオクリップを視聴するという方法をとったが、もちろん、すでに別稿で詳しく解説したとおり、ウェブページ上に WMP を埋め込むこともできる（染谷 2008b）。また、個々のビデオクリップをあらかじめ「ハードサブ」として作成しておけば、字幕ファイルの配信や文字化け等の心配なしに字幕付き動画を配信することも可能である（染谷 2000c）。どのような方法をとるかは、それぞれの必要に応じてということになる。

3.　「再生位置マーカー」の挿入

WMEd のもうひとつの重要な機能は「再生位置マーカー」の挿入である。これは、wmv 形式の動画ファイルをいくつかの論理的な単位に分割し、それぞれの開始点にコンピュータが認識できる形式の「標識」を埋め込むという作業である。これによって、たとえば、プレゼンテーション用の動画ファイル（例えば、染谷 2008b の付録で解説したパワーポイント動画の例参照）を作成する場合、主要なトピックごとにマーカーを追加することで、ユーザは「マーカー一覧」から任意のトピックを選択してファイルの該当部分に進むことができるようになる。また、この機能を利用して、字幕の各行をマーカーとして挿入しておけば、クリックひとつで任意の字幕位置にジャンプするといったこともできる。こうしておけば、授業で特定の箇所を学生に見せたい場合などに、場所の見当を付けて探す必要がなくなり、効率的な指導が可能になる。「マーカー」挿入の作業手順は以下のとおり。

1. WMEd を起動（本文の**図 2**, p. 4 参照）。
2. 起動後、「マーカー」タブをクリックした上で、メニューから「ファイル」→「開く」を選んで目的の動画ファイルを開く（**図 19**）。
3. マーカーを追加したい箇所でいったん再生を停止し、 [マーカー] ボタンをクリック。

図 19　WMEd の起動→動画ファイルの読み込み

4. **図 20** のような「マーカー」編集ボックスが表示されるので、画面右端にある [追加] ボタンをクリック。

図 20　「マーカー」編集ボックスの初期画面

②図 5 で「マーカー」ボタンをクリックすると左のような画面（マーカー一覧＝この時点では空白）になるので、ここで「追加」ボタンをクリック。

5. 続いて **図 21** のような「マーカーのプロパティ」入力ボックスが表示されるので、[名前] のところにマーカーとして入力したい情報を入力。この例では、字幕の各行にジャンプさせるために、字幕番号（連番）と当該字幕行の冒頭部分を入力している。

図 21　「マーカーのプロパティ」入力ボックス

③図 6 で「追加」ボタンをクリックすると左のような画面になるので、ここでマーカーの名称（または字幕の内容）を記入して OK ボタンをクリック。

なお、マーカーとして入力したテクストはパソコンの画面横幅いっぱいまで表示されるが、あまり長いと一覧性が低くなるので、**図 21** の入力ボックスの範囲内に収めておくのがよい。マーカーの [時間] は**図 19** で停止した箇所の時間が 100 msec（0.1 秒）単位で自動的に表示される。なお、SAMI ファイルを使って字幕を同期表示する場合、基本的には各マーカーのタイミングは SAMI ファイルでの字幕各行の同期時間に合わせることになるが、実際には各マーカーのタイミングを、SAMI ファイルでの字幕各行の同期時間より 100 msec 早く設定しておくと比較的スムースに同期表示が行われるようである。

6. [マーカー] の入力を終え、[OK] ボタンをクリックすると画面は**図 22** のようになり、挿入したマーカーが「マーカーバー」に沿って赤い印で表示されているのが確認できる。

図 22　マーカーインジケータの表示

④図 7 で OK ボタンをクリックすると左図のような画面に戻る。先ほど挿入したマーカーが赤いインジケータで示されている。

7.　以上の作業を必要なだけ繰り返し、最後までマーカーを挿入。**図 23** にすべてのマーカーの挿入が終了した状態のイメージを示す。

図 23　すべてのマーカーの挿入が終了した状態

8.（「マーカー一覧」の表示とマーカーの編集）**図 23** の状態から [マーカー] ボタンをクリックすると、**図 24** のような「マーカー一覧」が表示される。マーカーを編集する場合は、目的のマーカーをマウスで選択し（選択したマーカーが強調表示される） [編集] ボタンをクリ

ック。**図 21** の画面になるので、当該マーカーのテクストまたは時間の値を適宜変更する。

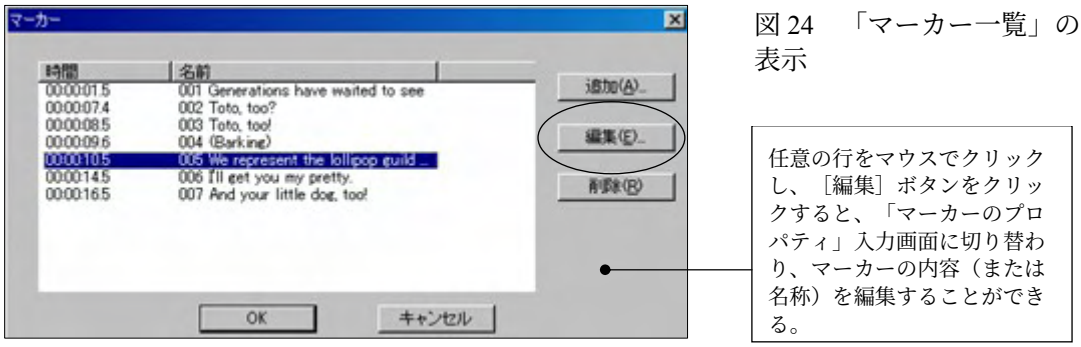


図 24　「マーカー一覧」の表示

9.（マーカーデータの保存）すべてのマーカーの入力（と編集）が終了したら、最後にメニュー

ーの「ファイル」→「保存およびインデクス作成」を選択（**図 25**）。これでマーカーデータが動画ファイル中に保存される。

※新規ファイルとして保存したい場合は上記で「名前を付けて保存およびインデクス作成」を選択。ただし、マーカーデータはいつでも編集・削除できるので、オリジナルファイルに上書き保存してもまったく問題はない。

10. 保存が終了したら WMEd を終了させて作業終了。

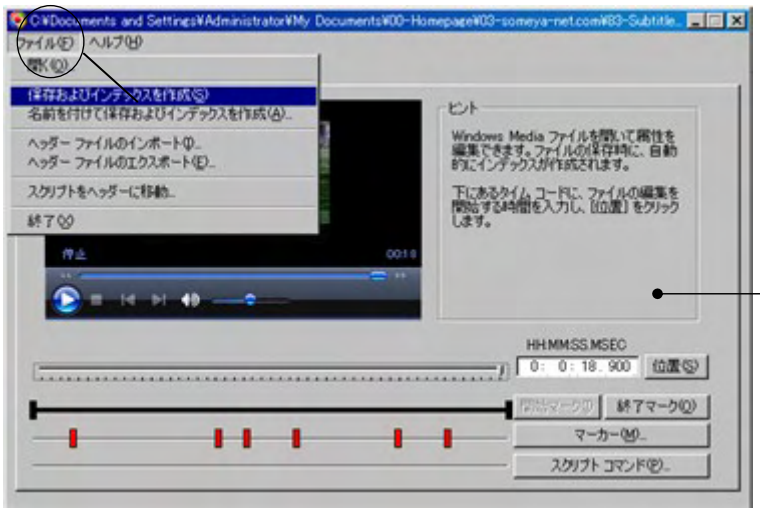


図 25 作成したマーカーデータの保存

メニューから「ファイル」→「保存およびインデクスを作成」を選択。→マーカーデータが自動的に保存されるので、保存終了後、WMEd を閉じる。

4. WMP で「再生位置マーカー」付きの動画ファイルを再生する

動画ファイルへの「再生位置マーカー」の挿入が終了したら、次に WMP でこの動画ファイルを立ち上げ、マーカーが適切に挿入されているかどうかを確認する。WMP 上でマーカーを表示させるには、**図 26** に示すとおり、当該の動画ファイルを読み込んだあと、再生をいったん停止（または一時停止）し、WMP のメニューから「表示」→「ファイルマーカー」を選択する。これで、「ファイルマーカー一覧」が自動的にポップアップ表示される。

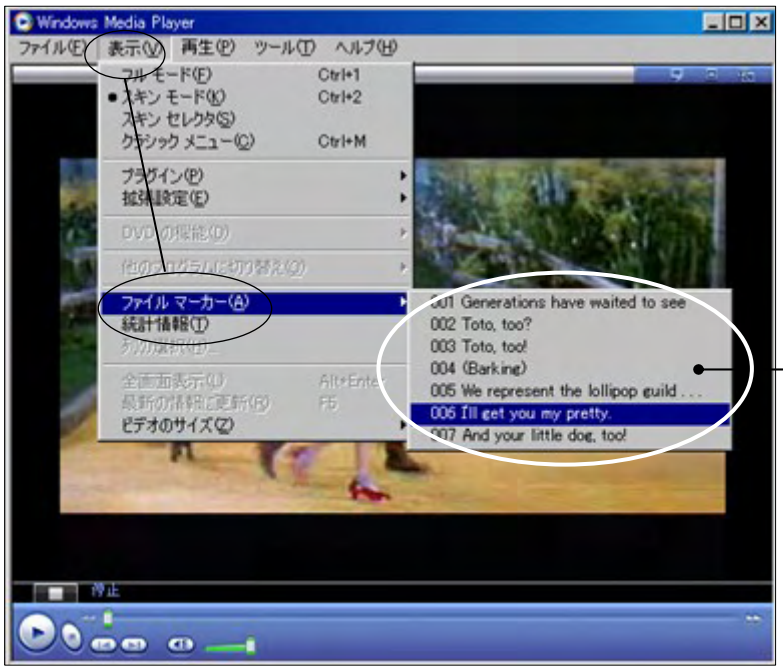


図 26 「再生位置マーカー」付きの動画ファイルを WMP で立ち上げる

メニューから「表示」→「ファイルマーカー」を選択すると、「ファイルマーカー」一覧がポップアップ表示される。

この「ファイルマーカー一覧」の中から任意のマーカーを選択してクリックすると、再生イ

ンジケータが該当箇所に自動的にジャンプする。続いて「再生」ボタンをクリックするとその位置から動画の再生が開始される。なお、当該の動画ファイルに対応する字幕ファイル（SAMI ファイル）が別途用意されている場合は、WMP の字幕表示機能をオンにしておくと（メニューから「再生」→「歌詞、キャプションおよび字幕」を選択）、**図 27** に示すように、映像と同時に字幕を表示させることができる。

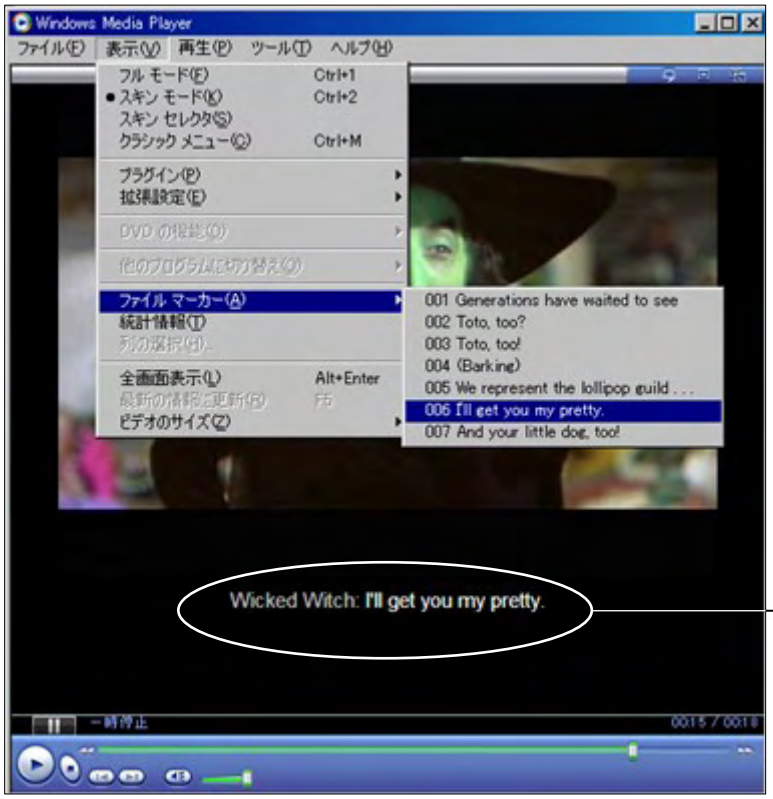


図 27 「再生位置マーカー」付きの動画ファイルを WMP で立ち上げ、同時に字幕表示をさせた状態

別途用意した字幕ファイル（SAMI ファイル）のデータを表示。

なお、ファイルマーカー（＝再生位置マーカー）のデータと字幕ファイル（SAMI ファイル）とは別ものであり、ファイルマーカーに字幕ファイルと同じデータが保存されていても、それだけでは字幕表示はできないので注意されたい。また、字幕表示をさせた状態で再生位置をジャンプさせた場合、字幕の表示が間に合わないことがある。この場合は、再度、同じ操作を繰り返すことで、ほぼ確実に該当箇所の字幕が適切に表示されるようになる。

5. ウェブページ上に埋め込まれた WMP での表示と操作

前項では WMP を単独で立ち上げた場合の例を示したが、ウェブページ上に埋め込まれたメディアプレーヤ上でも同じ操作が可能である。この場合の操作手順は以下のとおり。なお、ウェブページへの動画の埋め込みおよび字幕ファイルとの同期については染谷 (2008b) ですでに述べたとおりである。

1. 当該の動画が埋め込まれたウェブページを開く（**図 28**）。
2. 動画が表示されたらいったん再生を停止し、画面上にマウスを置いて右クリック。

3. **図 28** の下図に示すような「ファイルマーカー一覧」ボックスがポップアップ表示されるで、プルダウンメニューから任意の箇所を選択して [OK] をクリック。

図 28　ウェブページ上に埋め込まれた WMP での表示と操作

※ [URL]　http://www.someya-net.com/83-SubtitleVideo/Public/Wizard_of_OZ(marker).html

4. 次に、プレーヤの再生ボタンをクリック。これで、選択した位置から動画の再生が開始される。

なお、字幕表示はウェブページ上で制御しているので、ローカルの WMP の字幕機能はオフでもかまわない。また、スタンドアロンの WMP の場合と同じく、字幕表示させた状態で再生位置をジャンプさせた場合、字幕の表示が間に合わないことがある。この場合、通常は、もういちど同じ操作を繰り返すことで、ほぼ確実に該当箇所の字幕が適切に表示されるようになる。

ただし、字幕表示を伴わない場合、または「ハードサブ」動画の場合はこの問題は起こらない（HTML_Template7.html（「ハードサブ」動画のサンプル）参照）。

【注】

1) 「Windows Media エンコーダ 9」だけでなく、そのほかのコンテンツ作成および編集ツールを含む「Windows Media ツール 4.1」という総合パッケージをまるごとインストールしてもよい。こちらのダウンロード先は以下のとおり

- http://www.microsoft.com/japan/windows/windowsmedia/download/tool41.aspx

なお、「Windows Media エンコーダ 9」のヘルプファイルには「（Windows Media ファイルエディタの）正式名称は Windows Media ASF Indexer です」とある。ただし、「Windows Media ツール 4.1」に含まれる「Windows Media ASF Indexer」は「Windows Media ファイルエディタ」とは異なったソフトウェアになっている。このあたりの事情は不明だが（おそらくは、他の例にも見られるバージョンアップや統合化に伴う記述上の矛盾ということだと思われるが）、マイクロソフトの言う「正式名称」にかかわらず、本稿で使用するのは「Windows Media ファイルエディタ (ver. 9)」である。

補足：ちなみに、ASF Indexer にもミリセコンド単位の時間表示機能があり、WMEd と同様の作業をすることができる。ただし、ASF Indexer では拡張子 asf 以外のメディアファイルは読み込めない仕様になっているため、wmv 形式の動画ファイルの編集をするためには、いったん拡張子を asf に書き換える必要がある（作業終了後、再度、wmv に書き戻す）。

2) 「Windows ムービーメーカ」については染谷（2008b）の巻末付録として添付されている「パワーポイントスライドの動画化について」と題する小論の中で、その具体的な使用例について言及している。

3) **図 2** では「Windows Media ツール 4.1」を同時にインストールしてあるため、「Windows Media エンコーダ 9」を単独でインストールした場合と多少異なった構成になっている。

4) 表示形式は HH:MM:SS:MSEC（HH=時、MM=分、SS=秒、MSEC=ミリ秒）で、MSEC は最大を 999 とする 3 桁の数字で表示される。

5) なお、このことから、一般に字幕のスタートは厳密に音声の開始時間に合わせるのではなく、音声より少し早めに設定する（実際の音声より 100～200ms ほど前に字幕を出す）のがよいと考えられる。こうすることで、視聴者はあたかも字幕と音声がぴったりと同期しているかのように感じるからである（染谷 2008b, **注 21** 参照）。字幕表示タイミングの設定に当たってはこのあたりのことも考慮に入れておくとよい。

6) もっとも、教員が実験・研究用に入手する場合は、有償・無償を問わず役に立ちそうなものは入手しておいたほうがよい。有料のソフトウェアとしては米国の HiSoftware 社が開発・販売している **Hi-Caption Studio** (Ver. 9.0.0.1) がある。これは SAMI ファイルの作成・編集に特化した字幕エディタで、マイクロソフト社の技術文書中でも紹介されているものである。ユーザインターフェイスもシンプルで、たいへん使いやすいソフトウェアである。最新版では、作成した SAMI ファイルを SMIL 形式にインポートする機能が加わり、RealPlayer/RealOne および QuickTime での使用にも対応するようになっている。詳しくは同社の HP (http://www.hisoftware.com/hmcc/) を参照。また、同じく米国の TechSmith 社が開発・販売している **Camtasia Studio**（最新版は Version 6）というソフトウェアもぜひ紹介しておきたい。これは、screen recording software ということになっており、基本的には画面上の

操作をそのまま動画として記録し、必要に応じて音声やコメント表示を加えたりするためのソフトウェアである。この機能を利用してパワーポイントを動画化することもでき、さらに動画ファイルを読み込んで字幕を加える機能もある。字幕機能はとくに秀逸で、あらかじめテキストファイルとして用意した字幕データを、マウスによるごく簡単な操作のみで動画と同期させることができるようになっている。また、作成した字幕付き動画は swf/flv, wmv, avi, mov, m4v, rm など任意のフォーマットで保存することが可能である。詳しくは同社の HP（http://www.techsmith.com/）を参照されたい。

7) Subtitle Workshop を使ったファイル形式の変換については染谷 (2008c) の**注 8** 参照。なお、注 8 では smi 形式を srt 形式に変換する例を解説しているが、その逆の場合も基本的な手順は同じである。

8) 動画ファイルからの音声の切り出しはさまざまな方法で可能だが、もっとも簡単な方法として WMP に Chronotron WAV Output Plug-in (http://www.wmplugins.com/ItemDetail.aspx?ItemID=820) という WAV 音声抽出専用のプラグインを入れておく方法がある。このプラグインを入れておくと、WMP で再生した音声（映像の場合はその音声）をバックグラウンドで自動的に別ファイルに排出してくれる（デフォルトの排出先は [マイドキュメント] の [WAV Output] フォルダ）。詳しくは Chronotron WAV Output Plug-in の解説ページ (http://wweden.s18.xrea.com/ability/l_s_wmp.html) を参照されたい。

このほか、染谷 (2008c, pp. 18-19) で解説した Media Subtitler の音声抽出機能を使う方法もある。

9) mkv は通称 Matroska と呼ばれるオープンソースの動画コンテナ＝動画フォーマット (Matroska Video) の拡張子。ロシアの入れ子人形マトリョーシカにちなんで名付けられたもの。音声コンテナは mka ファイル (Matroska Audio) として知られている。

10) ogm は Ogg Media または Ogg Movie の略で、Ogg（オッグ）と呼ばれるコンテナ形式の仕様を元に関係者が個人で開発した動画形式。一般の Windows 用メディアプレーヤで ogm 形式の動画を再生するには対応するコーデックが必要。前述の mkv とともに、コンテナフォーマットとして avi より進化した形式とされている。

11) 既存の wav ファイルを SubStation Alpha に対応する形式に書き換えたり、映像ファイルから音声だけを wav ファイルとして取り出すには、染谷 (2008c, p. 22) で紹介した TMPG Encoder を使うのがもっとも手っ取り早く、かつ確実な作業ができる。ごく簡単に手順を紹介しておく。①まず、対象となる音声ファイルまたは映像ファイルを TMPG Encoder に読み込む。②メニューから [ファイル] → [ファイルに出力] → [WAVE ファイル] を選択。③保存画面が出てくるので、ここで「音声を出力」にチェックを入れ、右側の [設定] ボタンをクリック。④「サウンドの選択」画面になるので、ここで「形式」に「PCM」を、「属性」に「8000Hz, 8 ビット，モノラル」をそれぞれ選択して OK をクリッ

ク。⑤あとは、このファイルに適当なファイル名を付けた上で保存。なお、wav 音声の編集には SoundEngine Free (http://www.forest.impress.co.jp/lib/pic/music/soundedit/soundengine.html) がおすすめ。

12) 字幕ファイル形式にはさまざまなものあり、srt, ssa, ass などの標準的なもののほかに、各社がそれぞれ独自の形式を開発採用している。参考までに、ここに例示されているものについて拡張子とその対応形式・搭載製品名の一覧を挙げておく（URUWorks. (2001-8) より引用）。

拡張子	対応形式・搭載製品名
srt	SubRip
ssa	SubStation Alpha
ass	Advanced SubStation Alpha
sub	DVDSubtitle, MicroDVD, SonicDVD Creator, SubViewer, SubSonic, etc.
txt	Adobe Encore DVD, DVD Subtitle System, MAC DVD Studio, Pinnacle, QuickTime Text, SubCreator, SubCreator, Ulead DVD Workshop, etc.
dks	DKS Subtitle Format
scr	MacSUB
mpl	MPlayer
pjs	Phoenix Japanimation Society
psb	PowerDivX

【参考文献･資料※】

[01] Microsoft Corp. (2000) 「Windows Media メタファイルの活用」（日本語版最終更新日 2001 年 12 月 3 日） [Online] http://www.microsoft.com/japan/msdn/windowsmedia/production/asx.aspx

[02] URUWorks. (2001-8) “Subtitle Workshop -- Current list of supported formats.” [Online] http://www.urusoft.net/products.php?cat=sw&lang=1

[03] 佐藤方彦（監修）(1992) 「人間工学基準数値数式便覧」技報堂出版

[04] 染谷泰正 (2008a) 「動画への字幕の挿入について： 英語教員のための Do-It-Yourself マニュアル (1)」

[05] ---------- (2008b)「字幕付き動画のオンライン配信について:英語教員のための Do-It-Yourself マニュアル (2)」

[06] ---------- (2008c) 「『ハードサブ』動画の作成について：英語教員のための Do-It-Yourself マニュアル (3)」

※ URL はいずれも 2008 年 6 月 30 日現在のもの。

青山学院大学
Aoyama Gakuin University