FI MU

Faculty of Informatics
Masaryk University

PHC Format Description

by

Pavel Frýda
Ivan Kopeček

FI MU Report Series FIMU-RS-97-07

November 1997

PHC Format Description

Pavel Frýda and Ivan Kopeček

Faculty of Informatics, Masaryk University Brno
Botanická 68a, 602 00 Brno, Czech Republic

E-mail: `kopecek@fi.muni.cz, fryda@fi.muni.cz`
WWW: `http://www.fi.muni.cz/ kopecek/`

Abstract

The paper contains the presentation of a data format for phonetic corpora called PHC format and proposal of library of utilities designed for manipulation with PHC format.

1 Introduction

Development of speech synthesis and recognizing systems and other branches of the natural language processing area as well as creating and analysing of linguistically oriented corpora results in growing need for the standardisation of phonetic data. In what follows we propose a format designed especially for such applications. This format should be supported by a library of utilities (conversion program, editing subroutines etc.) which should make it easier and more comfortable to use the data in this format. Internally the format is used for developing of the syllable based speech synthesizer DEMOSTHENES [Kop–97].

2 Motivation

The basic ideas of the proposed format are as follows:

- The format must be flexible enough to be easily used for various applications.
- The format must enable automatic processing of the data.

Because of the heterogenity of possible applications free structure of the format header was chosen. The format is called PHC (PHonetic Corpus).

3 Format Structure

Data file in PHC format consists of the header and the data part.

3.1 Header Definition

Bytes 1-3 contain characters 'p', 'h', 'c' denoting the PHC format. Further header data may contain specifications of the type: #keyword=valueˆ and comments as free text between specifications. Keywords (specifications) can be either reserved or private. Reserved keywords have specific meaning declared in the format specification and consist of small letters. Private keywords are supposed to be explained in comments and consist of capital letters. Using the private specifications is supposed when no suitable reserved specification is available. Header is terminated by the sequence of characters '#' '#';

3.2 Data Part

The data part contains phonetic data.

3.3 Header Sections

Although the structure of the file header is free it can be thematically divided into Technical Section, Reference Section, Speaker Section and Description Section.

4 Technical Section

Technical Section contains the following technical file specifications:

KEY WORD: `channels`;
  PARAMETER: <number of channels>

KEY WORD: `frequency`;
  PARAMETER: <sampling frequency in Hz>

KEY WORD: `bits_per_sample`;
  PARAMETER: <bits per one sample in one channel>

KEY WORD: `data_type`;
  PARAMETER: <[1] | [2] | . . . >

(Type of the sample. Valid values are listed in Tab.1, see [I&M–91].)

KEY WORD: `mic_type`;
  PARAMETER: <type of microphone>

Examples

```
#channels=1ˆ
#frequency=44100ˆ
#bits_per_sample=16ˆ
#data_type=1ˆ
```

Value	Data type
1	WAVE FORMAT PCM
2	WAVE FORMAT ADPCM
5	WAVE FORMAT IBM CVSD
6	WAVE FORMAT ALAW
7	WAVE FORMAT MULAW
16	WAVE FORMAT OKI ADPCM
17	WAVE FORMAT DVI ADPCM or WAVE FORMAT IMA ADPCM
21	WAVE FORMAT DIGISTD
22	WAVE FORMAT DIGIFIX
32	WAVE FORMAT YAMAHA ADPCM
33	WAVE FORMAT SONARC
34	WAVE FORMAT DSPGROUP TRUESPEECH
35	WAVE FORMAT ECHOSC1
36	WAVE FORMAT AUDIOFILE AF18
257	IBM FORMAT MULAW
258	IBM FORMAT ALAW
259	IBM FORMAT ADPCM
512	WAVE FORMAT CREATIVE ADPCM

Table 1: Valid values of `data_type`. If some additional parameters are needed they can be specified by means of the private specifications.

5 Reference Section

Reference section contains the following references :

KEY WORD: `reference_name`;
  PARAMETER: <name of the person supervising the data>

KEY WORD: `reference_address`;
  PARAMETER: <address of the person supervising the data>

KEY WORD: `reference_email`;
  PARAMETER: <e-mail of the person that supervises the data>

KEY WORD: `institute`;
  PARAMETER: <name of the institute where the data have been sampled>

KEY WORD: `project`;
  PARAMETER: <specification of the project related to the data>

Examples

```
#reference_name=Henry Kucera^
#reference_address=Botanicka 68, 60200 Brno, Czech Republic^
#reference_email=kucera@fi.muni.cz^
#institute=Masaryk University^
```

```
#project=INCO Copernicus BEST Project^
```

6 Speaker Section

Speaker Section involves various pieces of information about the speaker.

KEY WORD: `speakers_number`;
  PARAMETER: <[1] | [2] | ...>

KEY WORD: `speaker1_name`;
  PARAMETER: <name of the speaker>

KEY WORD: `speaker1_sex`;
  PARAMETER: [m] | [f] (m for male, f for female);

KEY WORD: `speaker1_age`;
  PARAMETER: <age of the speaker>

KEY WORD: `speaker2_name`;
  PARAMETER: <name of the speaker>

KEY WORD: `speaker2_sex`;
  PARAMETER: [m] | [f] (m for male, f for female);

KEY WORD: `speaker2_age`;
  PARAMETER: <age of the speaker>

⋮

KEY WORD: `speaker'n'_name`;
  PARAMETER: <name of the speaker>

KEY WORD: `speaker'n'_sex`;
  PARAMETER: [m] | [f] (m for male, f for female);

KEY WORD: `speaker'n'_age`;
  PARAMETER: <age of the speaker>

Examples

```
#speakers_number=1^
#speaker_name=Jan Noha^
#speaker_sex=m^
#speaker_age=25^
```

7 Description Section

Description Section contains the following specifications:

KEY WORD: `method_of_sampling`;
  PARAMETER: <the used method of sampling>

KEY WORD: `text`;

PARAMETER: <text relevant to the sample>

KEY WORD: `text_code`;
  PARAMETER: <text code specification>

Examples

```
#method_of_sampling=text reading^
#text=first sentence^
#text_code=ascii^
```

8 Library of Utilities for PHC Format Support

PHC format will be supported by the tools and utilities freely available to PHC-users via WWW. The utilities will be developed for MS-DOS, WINDOWS and UNIX operating systems.

8.1 Conversion tools

convert other sound formats (like WAV, VOC etc.) into PHC format. Reverse conversions will also be available.

```
wav2phc.exe in.wav out.phc [in2.hdr]
```

`wav2phc` converts `in.wav` file in WAV format into `out.phc` file in PHC format. Since WAV file doesn't need contain all PHC information this additional information can be specified in `in2.hdr`.

```
phc2wav.exe in.phc out.wav [out2.hdr]
```

`phc2wav` converts PHC format into WAV (analogously to the previous description).

```
voc2phc.exe in.voc out.phc [in2.hdr]
```

```
phc2voc.exe in.phc out.voc [out2.hdr]
```

The utilities work in the same way but instead of WAV format they convert VOC format.

8.2 Tools for play-back and recording

Here we present the tools that allow sound recording and storing in PHC format or play back the sound from PHC files.

```
phc_play.exe in.phc
```

`phc_play` plays back the sound stored in PHC file `in.phc` and displays the information about the sample.

```
phc_rec.exe out.phc [in.hdr]
```

`phc_rec` records sound and then stores it in the PHC file `out.phc`. The PHC file header can be specified in `in.phc`.

8.3 Editing tools

Editing tools can be divided into two groups. The first one includes tools that can edit or modify header of PHC file. The second one includes tools that can modify samples.

```
headedit.exe file.phc
```

`headedit` allows to edit the header of PHC file `file.phc`. Items can be added, removed, modified or updated.

```
phc_add.exe [file.phc | ALL] phc_add.txt
```

`phc_add` adds text from the file `phc_add.txt` into the header of PHC file `file.phc` (or to all PHC files in the current directory if ALL is specified.)

```
phc_smp.exe
```

`phc_smp` allows to work directly with the sample contained in PHC file. The sample can be split into two PHC files, parts of the sample can be deleted etc.

```
phc_chck.exe in.phc
```

Checks syntax of the header of PHC file `in.phc`.

8.4 Utilities for program developers

The described utilities support programming of the systems that use PHC format.

```
phc_head.h, phc_head.cpp
```

The include file for basic operations with PHC format.

```
phc_play.cpp, phc_rec.cpp
```

The sources of utilities `phc_play` and `phc_rec`.

9 Conclusions, Future Work

We suppose that the preliminary version of PHC format description will be updated in near future. The library of procedures for PHC format support will be enlarged and modified according to the users needs as well.

References

[Kop–97] I. Kopeček: Syllable Based Speech Synthesis; Proceedings of the 2nd International Workshop Speech and Computer - SPECOM'97, Cluj-Napoca, 1997, pp. 161-165.

[I&M–91] IBM, Microsoft: Multimedia Programming Interface and Data Specifications 1.0, August 1991

Publications in the FI MU Report Series are in general accessible
via WWW and anonymous FTP:

`http://www.fi.muni.cz/informatics/reports/`
`ftp  ftp.fi.muni.cz (cd pub/reports)`

Copies may be also obtained by contacting:

Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic