



# Web Delivering of Music Scores

## DE4.5.1 Interfaces for Blind People

**Version: v1.4**

**Date:** 15/01/02

**Responsible:** ILSP

Project Number: IST-1999-10165  
Project Title: Web Delivering of Music Score  
Deliverable Type: restricted

Deliverable Number: DE 4.5.1  
Contractual Date of Delivery: 31 December 2001  
Actual Date of Delivery: 15 January 2002  
Title of Deliverable: Interfaces for Blind People  
Work-Package contributing to the Deliverable: WP3  
Nature of the Deliverable: RE  
Author(s): ILSP, SVB, ARTEC, IRCAM, DSI

**Abstract:**

This document describes the VIP Editor that has been implemented according to the specifications defined in WP 3. The VIP Editor is a software module that permits blind musicians to edit/access the musical content of wedel music objects. The blind user interacts with this module using the keyboard and receives feedback with synthetic speech (using text to speech synthesis technology) and Braille printer. It should be noted that this module has been built using functions of the WEDEL Tool Kit provided by DSI.

**Keyword List:**

music, internet delivering, audio format, image format, symbolic format, watermarking, protection, encryption, music distribution, copyrights, mp3, visually impaired people, speech music, Braille music, music analysis, music printing, music format.

# Table of Content

<b>9</b>	<b>WP4.5 -- Interface for Blind People.....</b>	<b>3</b>
9.1	Introduction .....	3
9.2	General Issues of the WEDELMUSIC Interface for Blind People .....	3
9.3	Architecture of the Implemented Module.....	3
9.3.1	General Module Design Considerations.....	3
9.3.2	General Architecture of the VIP Module.....	6
9.3.3	The WEDELMUSIC VIP EDITOR Module .....	7
9.3.4	The Braille Printing Module.....	23
9.3.5	The Spoken Music Module.....	26
9.4	Implementation Strategy.....	33
9.4.1	Conditions Needed for the Development .....	34
9.4.2	Interrelations With Other Modules/Components.....	34
9.4.3	Technology Used.....	35
9.4.4	Implementation Platform and Development Tools.....	35
9.5	Appendix: Index of Signs in Standard Braille Order.....	36
9.6	Appendix: Index of Signs in Standard Braille Order.....	47
9.6.1	TABLE 1 - Basic Signs .....	47
9.6.2	Table 2 - Clefs .....	47
9.6.3	Table 3 - Accidentals, Key & Time Signatures .....	48
9.6.4	Table 4 - Rhythmic Groups .....	48
9.6.5	Table 5 - Chords .....	49
9.6.6	Table 6 - Slurs & Ties.....	49
9.6.7	Table 7 - Tremolos .....	50
9.6.8	Table 8 - Fingering .....	50
9.6.9	Table 9 - Bar Lines & Repeats .....	51
9.6.10	Table 10 - Nuances .....	52
9.6.11	Table 11 - Ornaments .....	52
9.6.12	Table 12 - Theory .....	53
9.6.13	Table 13 - Modern Notation .....	54
9.6.14	Table 14 - General Organization .....	54
9.6.15	Table 15 - Keyboard Music .....	54
9.6.16	Table 16 - Vocal Music .....	55
9.6.17	Table 17 - String Instruments .....	56
9.6.18	Table 18 - Winds & Percussion.....	56
9.6.19	Table 19 - Accordion.....	56
9.7	Abbreviations .....	58
9.8	References .....	58

## 9 WP4.5 -- Interface for Blind People

### 9.1 Introduction

This document describes the Interface for Blind People that has been implemented according to the specifications set in WP3. This interface enables the blind musicians to have access and share the facilities of the WEDELMUSIC system like the sighted people. This implementation used many state of the art as well as conventional technologies like speech synthesis from text, Braille printing and technologies developed by the WEDELMUSIC project.

### 9.2 General Issues of the WEDELMUSIC Interface for Blind People

The key element of this interface for blind people is the **speech engine**. Of course, the speech engine by itself does not provide all the required functionality to render the WEDELMUSIC system truly accessible. A screen-reader program is used instead to surrogate the lack of visual feedback by conveying audible feedback to the user.

For visual feedback to be fully and efficiently replaced, any activity that results in change of focus, navigation through the current window or through the list of currently open windows, interfacing elements especially menu items or controls that represent command from the user to the system (e.g. buttons) or options made by the user to prescribe the behaviour of the system (e.g. dialog boxes, containing check boxes, radio buttons, list views, etc.), is reported to the user. Moreover, the users gain a precise understanding of the current system or application state and of the effects their commands had on it.

The editor supports **multi-modality** so as to increase the efficiency of the conveyed information. In fact the editor integrates both synthetic speech and Braille bar output to provide more complete and often redundant information to the users in a form of their preference.

Practice has shown that computer users (whether they are impaired or not) present a “loyalty” to software products, this meaning that they are not too willing to change them for new ones even when the replacement involves a more “advanced” system. This is especially true for VIPs whose acquisition time and effort tend to be higher.

There are a few widely spread and largely acceptable screen-reader systems available in the market. Indicative solutions include the HAL system, the JAWS system, etc. They represent the result of many years of effort and continuous feedback from the visually impaired persons community, managing to reach a high level of acceptance and a wide audience.

The WEDELMUSIC SYSTEM’s support for visually impaired persons relies on the presence of an adequate screen-reader application that helps in the following tasks:

- Browsing in the local computer’s software and file contents to locate the installed Internet browser.
- Text to Speech conversion of the distributor’s web page contents to logon, to set search criteria, to locate and select the required WEDELMUSIC file, and finally proceed with downloading it.
- Browsing in the local computer’s software and file contents to locate and execute the WEDEL VIP EDITOR application.
- Navigation in the WEDELMUSIC VIP EDITOR application’s menus and commands to perform desired actions.

Such an off-the-shelf solution ensures a larger acceptance by the community of blind people.

### 9.3 Architecture of the Implemented Module

#### 9.3.1 General Module Design Considerations

##### 9.3.1.1 Component and Subsystem Modularity Issues

The WEDELMUSIC VIP MODULE is part of the overall WEDELMUSIC SYSTEM, and divided into smaller modules and components.

##### 9.3.1.2 Current Location Information

One of the most important thing that VIP users need to know is the information concerning the **current location** in the editing area that is in the score. Although for sighted users this information can be obtained by simply looking at the graphical representation of the score, VIP users need to memorise their position and keep track to its changes as they proceed with editing and issuing commands. The current location is of

additional importance because it signals the **context** of numerous context-sensitive operations involved throughout editing a score.

In the implementation of the current location information the (*current measure, offset within the measure*) pair is reported to the user along with additional contextual information that includes the current part or layer being edited in a multi-part score. This is done through an appropriate command placed at the **Edi t** menu described later. It is the **Edi t/Where Am I?** command.

A caret-like mechanism has been implemented to keep track of the current location. This caret will explicitly define the **active position** within a score, i.e. the place where all editing will be targeted. Consequently, we define the **active symbol** to be the symbol immediately following the caret, the **active measure** to be the measure that contains the caret, the **active part** to be the part currently being edited, and so on.

The WEDELFORMAT supports multi-part and multi-layer scores. Although editing a multi-part score is quite straightforward for sighted users, it could be the source of confusion for VIP persons. Thus, editing takes place in a part-by-part basis to simplify the overall process. Users are provided the means to switch among different parts in a simple way (the Page-Up key goes the user to the previous Part while the Page-Down key to the next Part).

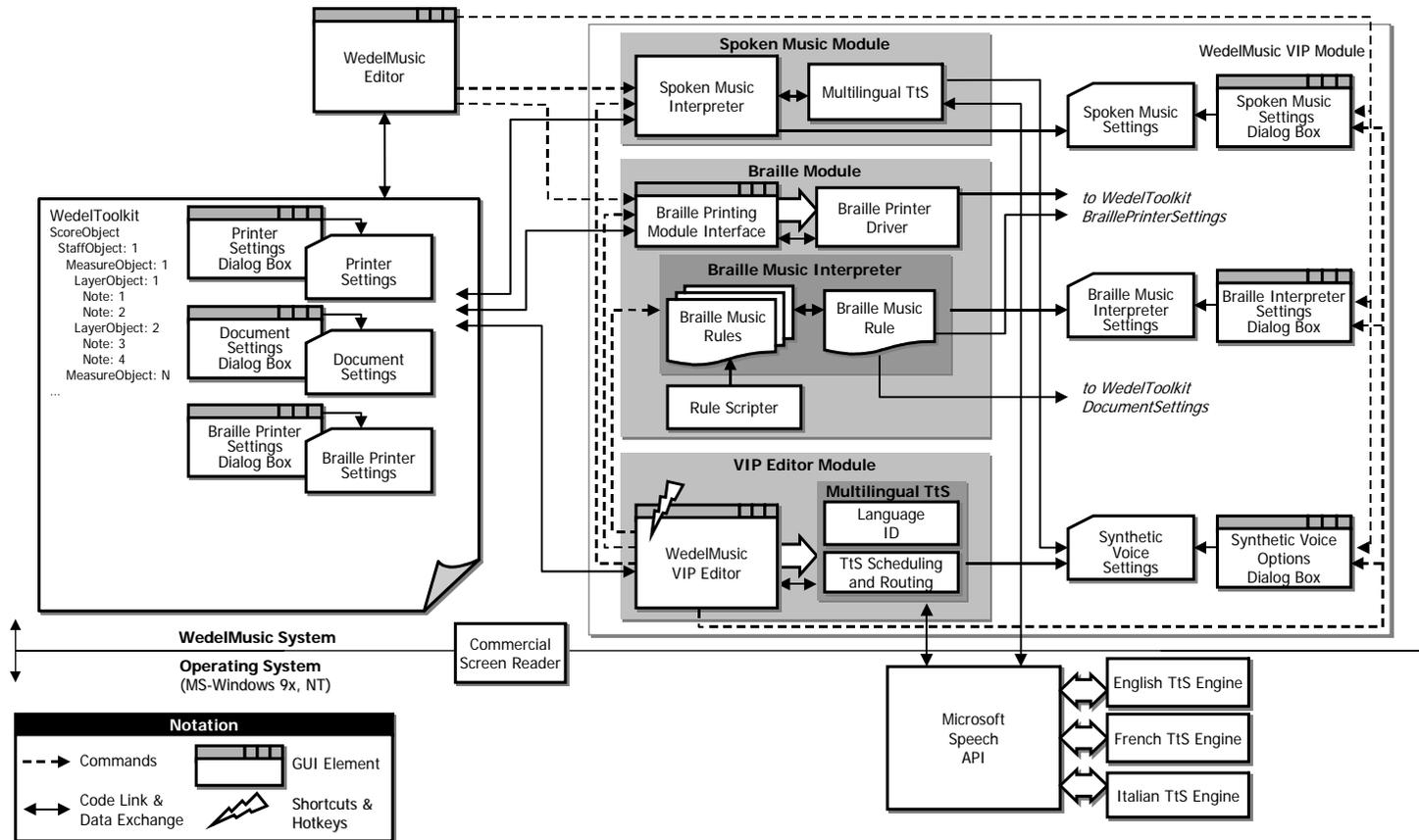
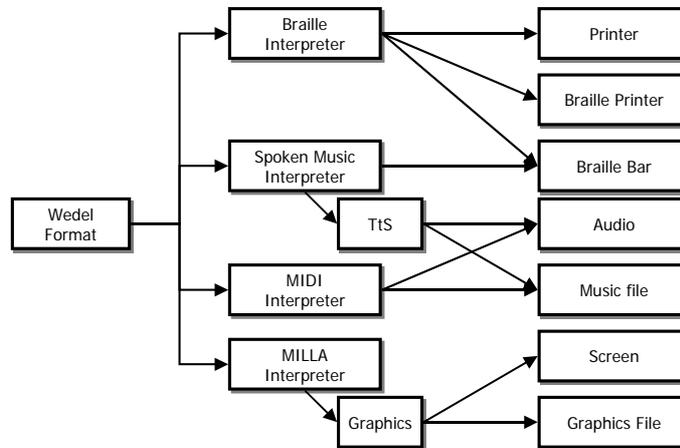


Figure 1. The layout of the WEDELMUSIC VIP MODULE.

### 9.3.1.3 Interpretations

There is a number of data interpretations that need to take place during the operation of the WEDELMUSIC system. Information from the WEDEL object needs to be transformed by appropriate interpreters to different formats to be directed to different output devices. An indicative schematic representation of such involved transformations is given in Figure 2.



**Figure 2.** Data interpretations and target output devices.

The VIP module contains two interpreters the following:

- **The Braille Interpreter.** Based on the information from the WEDEL object, it is targeted to both a (normal or Braille) printer using Braille Music format and to the Braille bar. Although these two interpretations present some differences, their backbone and most of their details are common.
- **The Spoken Music Interpreter.** The main purpose of this module is to interpret information from the WEDEL object to a descriptive textual form. This text is then converted to speech via a speech synthesiser.

### 9.3.1.4 Component Interrelations

The BRAILLE MODULE and the SPOKEN MUSIC MODULE are accessible from the main WEDELMUSIC Editor since they implement functionalities that may be relevant to sighted persons as well.

## 9.3.2 General Architecture of the VIP Module

The VIP MODULE represents the WEDELMUSIC SYSTEM's extensions for visually impaired persons. However, the module contains tools that need to be accessible from sighted persons as well, such as the SPOKEN MUSIC MODULE and the BRAILLE PRINTING MODULE. Apart from these tools, the main role of the VIP MODULE is to render the WEDELMUSIC EDITOR's functionality accessible to VIPs.

To this end, it makes implicit or explicit use of the following services:

- **Screen-Reader:** The VIP MODULE does not actually communicate directly with this component, but it certainly relies on its presence to handle focus changes, menu navigation, etc. providing appropriate audible feedback. After all, the WEDELMUSIC VIP EDITOR is a typical Windows application itself and can be treated by the screen-reader the same standard way. We used the Henter-Joyce JAWS for Windows 3.7 screen reading software.
- **Speech Synthesiser:** In order to provide additional information to the user, when appropriate, a TtS converter is required that supports the three project languages. This

TtS converter is explicitly controlled by the system. We used Microsoft Speech API (SAPI) and the speech components for English, French and Italian languages (from www.digalo.com : **Elan informatique** 4 rue Jean Rodier, 31400 Toulouse, France).

These are basically the modalities external to the overall system that the WEDELMUSIC VIP MODULE interacts (implicitly or explicitly) with.

The WEDELMUSIC VIP MODULE is divided into the following independent modules and components:

- The **WEDELMUSIC VIP EDITOR**. It is a simplified version of the WEDELMUSIC EDITOR targeted to the visually impaired persons. It contains a set of commands for:
  - **Retrieving** information related to the WEDELFORMAT header of the active file, as well as classification, identification, structural, and other information.
  - **Browsing** through a score using a caret-like mechanism. Information is automatically provided relatively to both the current caret location and the symbols in the vicinity of it.
  - **Editing** by means of inserting, deleting, modifying, marking, copying and pasting isolated symbols or entire measures.
  - **Reproduction** of any portion of the score. This involves diverse actions such as playing back, producing a hardcopy using a Braille printer, or transforming the part of the score to spoken music.
- **Multilingual TtS**. Spoken messages are involved in all the previously described actions, both to assist the VIP users to select the respective commands and to provide feedback on the results of their actions on the currently active score. It is more than usual that mixed-language messages may need to be conveyed to the user by the WEDELMUSIC VIP EDITOR, e.g. when combining information in the user's language with information in the system's language or in the default language used by a third party application. Those cases are handled by the next two components, namely the LANGUAGE IDENTIFICATION COMPONENT and the TtS SCHEDULING AND ROUTING COMPONENT described in the following.
- The **LANGUAGE IDENTIFICATION COMPONENT**. This component breaks the target text or sentence down to monolingual parts. Each part is sent to the corresponding TtS engine.
- The **TtS SCHEDULING AND ROUTING COMPONENT**. This component serves requests for TtS conversion for a set of monolingual texts. Each text is routed to the appropriate TtS engine to convert it to speech, while the next text is waiting in an internal queue the TtS engine to finish the current conversion and be ready for the next one. Also this component schedules the overall process and streams the audio samples to the audio hardware.
- The **BRILLE PRINTING MODULE**. This module handles all the details for printing a portion of a score to a Braille printer, e.g. selecting the target printer, setting the printing options, and performing the actual data transmission.
- The **SPOKEN MUSIC MODULE**. This module handles all the details for music-to-speech conversion in the user selected language.

Each of these modules and components is described in more detail in the following sections.

### 9.3.3 The WEDELMUSIC VIP EDITOR Module

This module is a simplified version of the WEDELMUSIC EDITOR targeted to the visually impaired persons. However, it contains some additional functions that are located in the WDF EDITOR AND INTEGRATOR module, so that, it can also be accessed by the blind user.

This component is the basic front-end of the WEDELMUSIC VIP MODULE. Although the VIP MODULE is addressed to visually impaired persons, a graphical user interface (GUI) has been implemented because as it has been said above an off-the-shelf commercial screen reader is used. This gets information from the standard GUI application window and provides the VIP a set of options and commands. The menus, dialog boxes, checkboxes and all other controls of this window are readily supported by the commercial screen-reader, providing the VIP with the ability to immediately start using them without the need of a training phase in case he is already accustomed to them.

After all, the VIP interface is an application and the visual impaired persons interact with it just the way they are used to interacting with other applications.

The existence of GUI enables the VIPs to *work co-operatively* with (and possibly be assisted by) non-VIP persons, at least during the first stages of their involvement with the system.

Finally the GUI in the VIP module provides a means for communication of the program to the Braille bar through the screen-reader, but more detail about it is given below.

### 9.3.3.1 The WEDELMUSIC VIP EDITOR User Interface Component

Figure 3 provides an indicative visualisation of the WEDELMUSIC VIP EDITOR User Interface.

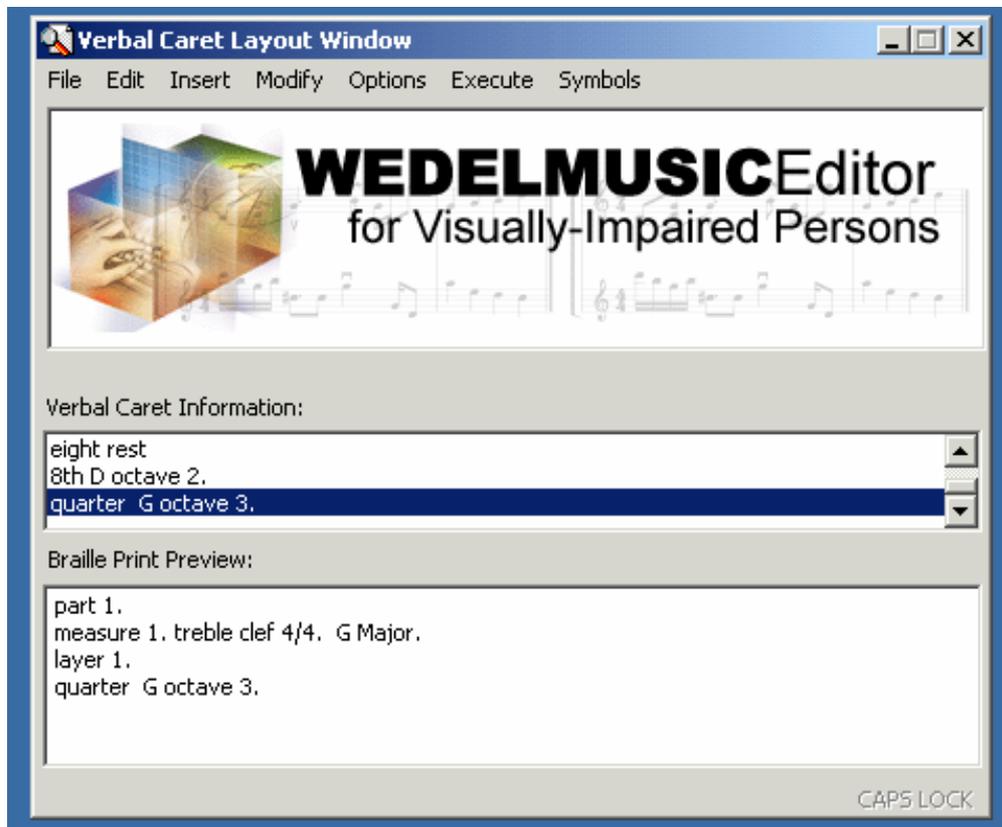


Figure 3. An indicative version of the WEDELMUSIC VIP MODULE interface.

It consists of:

- A **title bar** describing the purpose of the overall window.  
The screen-reader uses the text in the title bar to announce events on the window (e.g. gaining or loosing focus, creation, destruction, etc.).
- A **menu bar** containing all the commands available to the user.  
The various commands are structured into nested submenus and organised by their functionality. The complete list of the submenus and the menu items is provided in the following paragraphs.
- A **client area** that contains the logo of the VIP Editor from WEDELMUSIC project.  
It is questionable whether an application addressed to VIPs needs to contain any graphics. However, we decided to put it because such a visually attractive interface element does contribute to the overall impression taking into account that a non-VIP user could probably co-operate with a VIP person, at least during the very first steps of getting familiar with the WEDELMUSIC system.
- The **Verbal Caret Information**.

This List Box Control contains the history of the actions of the VIP user onto the score. The navigation through the score and the corresponding text that has been sent to the speech synthesizer is included into the Verbal Caret Information area.

- The **Braille Print Preview**

This text container field shows the output of the BRAILLE INTERPRETER. Whenever the VIP MODULE needs to explicitly port information to the Braille bar, the respective characters are written to this text container. The installed screen reader detects the change of text in a visual element and reports it to the user using all the available output devices. Thus, by inhibiting the speech output (reserving temporarily the sound card) the target text is sent to the Braille bar and consequently to the user. This technique presents a quite simple and straightforward alternative for accessing the Braille bar transparently.

Below follows a detailed presentation of all the submenus and menu items included in the WEDELMUSIC VIP INTERFACE.

### 9.3.3.2 Menu Bar, Submenus, and Menu Items

The menu bar of the system consists of nine items, each one leading to a respective submenu. These are:

-  The **File** menu as depicted in Figure 4, includes commands for handling importing, exporting, closing, etc. of files, commands for handling lists of changes, commands for setting printing parameters and performing the actual printing, and finally an option to leave the program.

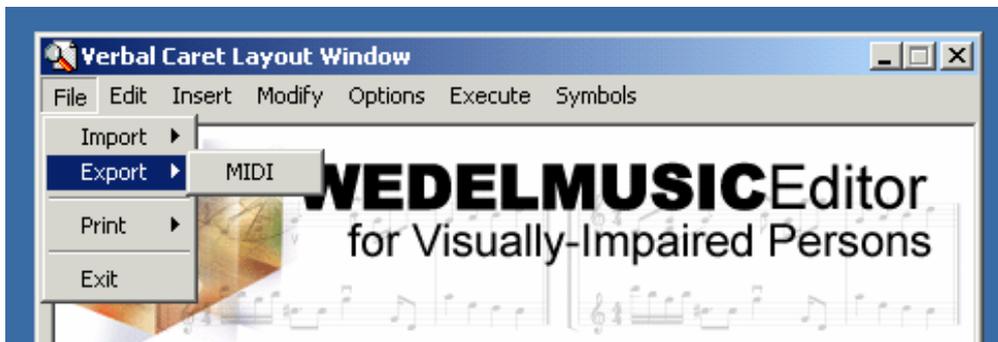


Figure 4: File menu: for importing, exporting, closing, printing commands.

-  The **Edit** menu as shown in Figure 5, includes commands for navigation (i.e. moving the caret position forward or backward), and control of selected regions (i.e. selecting a measure, cutting, copying, pasting a selection, etc.). A special menu item (Where Am I?) provides the users with the very important information about their current location.

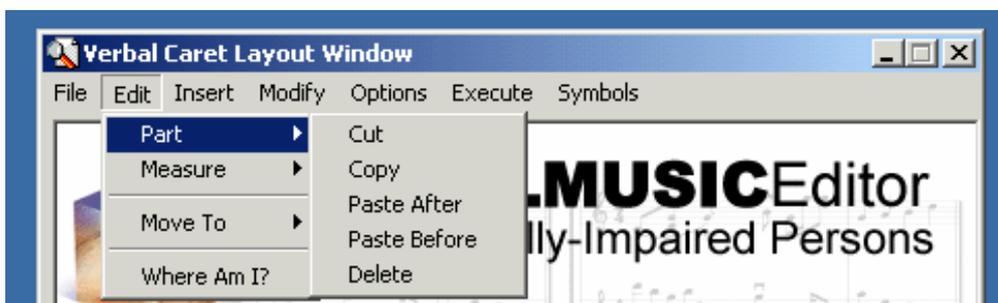
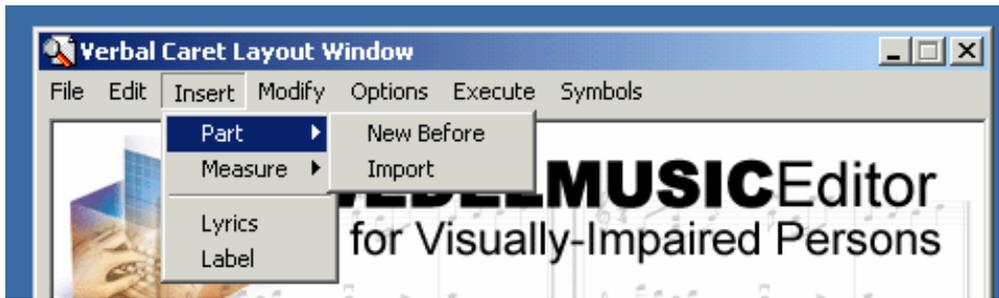


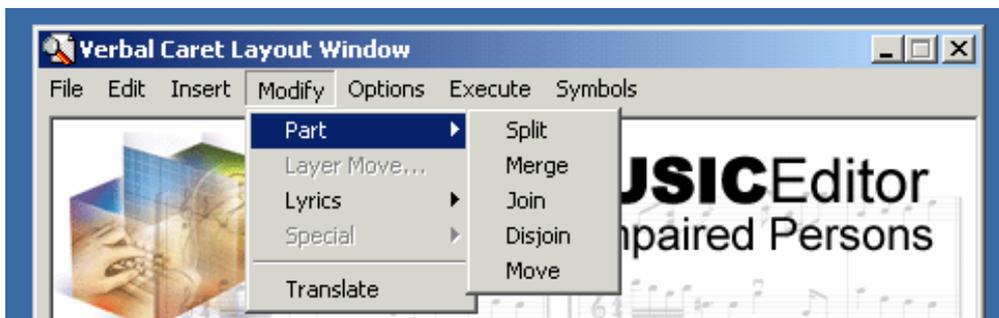
Figure 5: Edit menu: for navigation into the WEDELMUSIC score.

- The **I**nser**t** menu illustrated in Figure 6, contains commands for inserting new elements into the current score. Such elements may be of different kinds. Thus, commands are included for inserting parts, measures, lyrics and label.



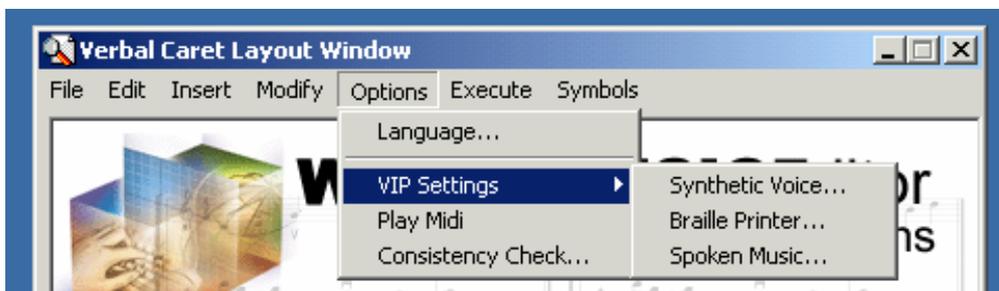
**Figure 6:** Insert menu: for inserting parts, measures into the WEDELMUSIC score.

- The **M**odi**f**y menu (Figure 7), contains commands for altering and/or setting the properties of symbols previously inserted through the **I**nser**t** menu.



**Figure 7:** Modify menu: for altering/setting properties of symbols inserted by Insert menu.

- The **O**pti**o**ns menu (Figure 9,10), leading to appropriate dialog boxes containing settings for the synthetic voice, the Braille interpreter, and the Spoken Music interpreter.



**Figure 9:** Options menu: configuration of Text 2 Speech synthesizer, Spoken Music etc.

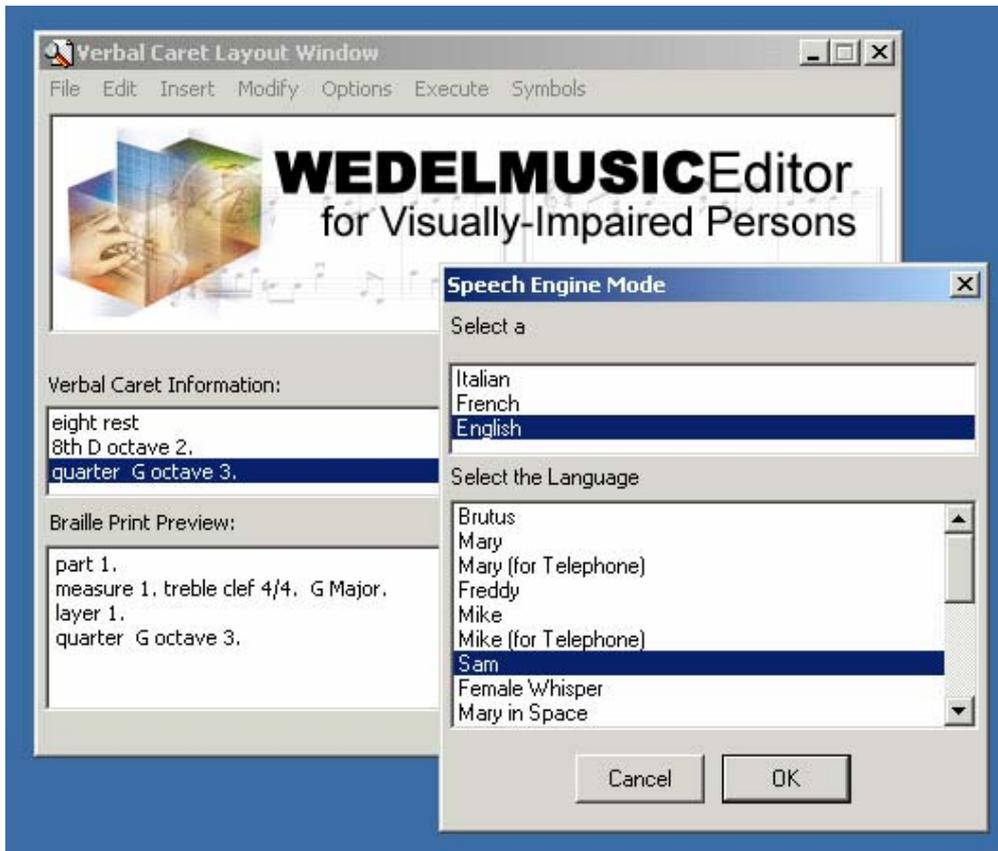


Figure 10: Options->Language.. Selection for text to speech synthesizer

- The **Execution** menu (Figure 12), containing commands for controlling the reproduction of the score either using audio or spoken music.

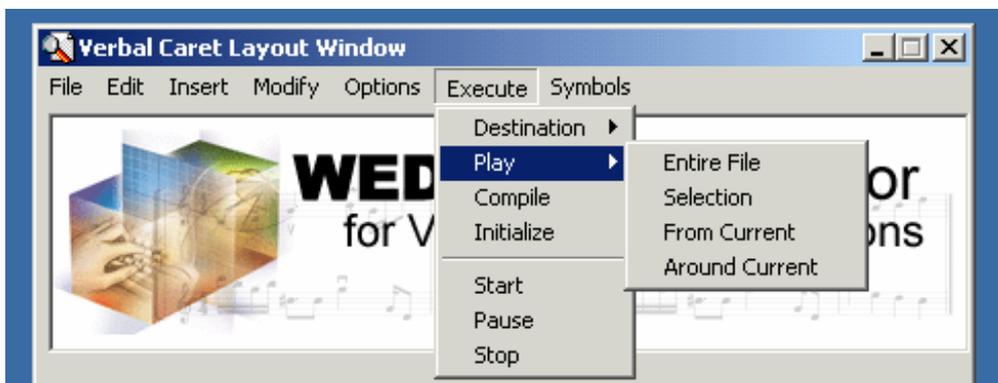
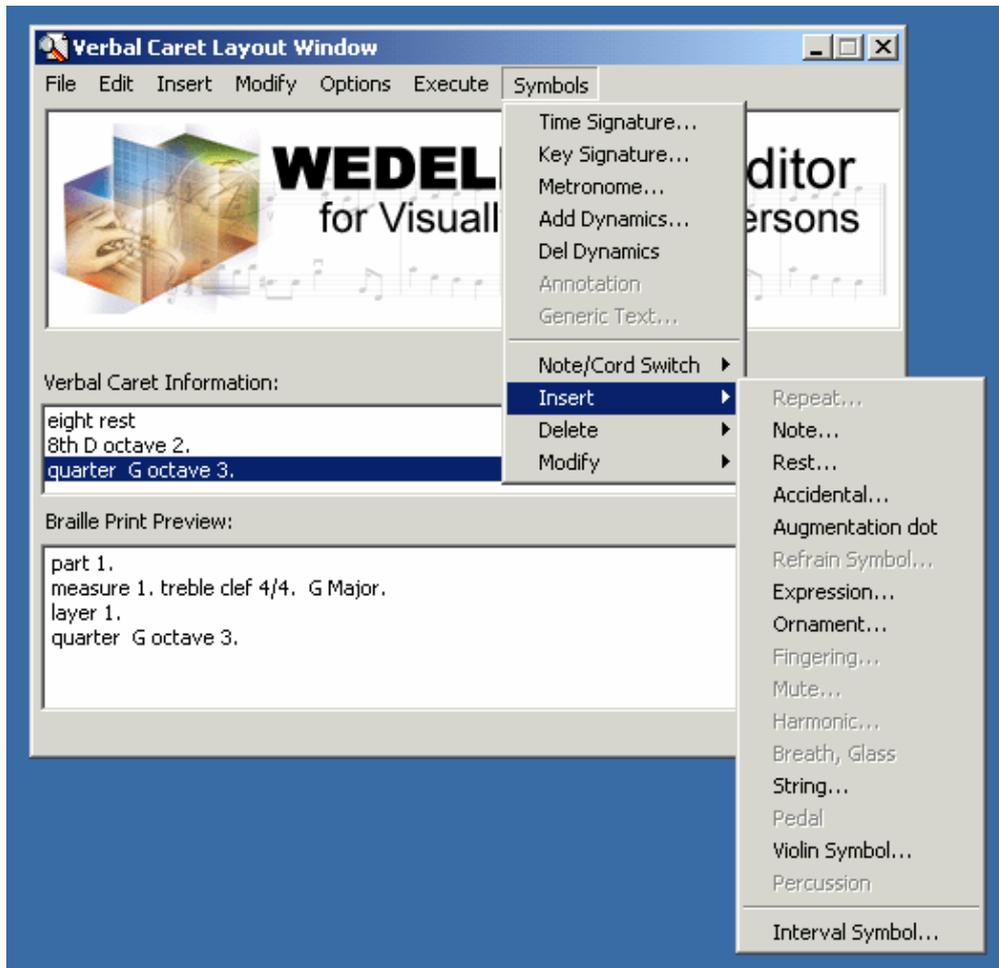


Figure 12: Execute menu: reproduction of the score either using audio or spoken music.

- The **Symbol s** menu contains (Figure 13) all the required commands for inserting notes, half notes, rests, and so forth.



**Figure 13:** Symbols menu: manipulation of music objects of the WEDELMUSIC score.

These menus are described in more detail in the following paragraphs.

File	Edit	Insert	Modify	Options
Import ▶ MIDI	Part ▶ Cut Copy Paste After Paste Before Delete	Part ▶ New Before Import	Part ▶ Split Merge	Language
Export ▶ MIDI	Measure ▶ Cut Copy Paste Delete	Measure ▶ New Before New After	Join Disjoin	VIP Settings ▶ Synthetic Voice Braille Printer Spoken Music
Print ▶ Braille Music	Move To ▶ Previous Part Next Part Previous Measure Next Measure Object (Part, Measure, S-Measure)	Lyrics Label	Move Layer Move. .	Play MIDI Consistency Check
Exit	Where Am I?		Lyrics ▶ Edit Remove	
			Special ▶ Transpose Clef Transpose Key	
			Translate	

Executi on	Symbol s
Destinati on ▶ <input checked="" type="checkbox"/> Audio <input checked="" type="checkbox"/> Spoken Music Play ▶ Entire File Selection From Current To Current Around Current Compile Initiali se ▶ Start    Pause ■ Stop	Time Signature Key Signature Metronome Add Dynamics Del Dynamics Annotation Generic Text Note/Cord Switch ▶ Shallow Note Editing Deep Note Editing Insert ▶ Repeat Note Rest Accidental Augmentati on dot Refrain Symbol Expressi on Ornaments Fingeri ng Mute Harmoni c Breath, Glass Stri ngs Pedal Viol in Symbol Percussi on Interval Symbol Delete ▶ Repeat Note Rest Accidental Augmentati on dot Refrain Symbol Expressi on Ornament Fingeri ng Mute Harmoni c Breath, Glass Stri ng Pedal Viol in Symbol Percussi on Interval Symbol Modi fy ▶ Clef Bar Line Repeat Note Rest Accidental Refrain Symbol Expressi on Ornament Fingeri ng Mute Harmoni c Breath, Glass Stri ng Pedal Viol in Symbol Percussi on Horizontal Symbol s

### 9.3.3.2.1 The File Menu

The **File** menu contains commands for manipulating files stored in local disk space. These are described in more detail in the following:

- **File/Import**  
Imports a file from a format other than the WEDELFORMAT. A list of supported formats follows:
  - MIDI  
 Same as the respective **File/Import MIDI** menu item of the main WEDELMUSIC EDITOR MODULE.
- **File/Export**  
Export data to a file in a format other than the WEDELFORMAT. An list of the supported formats follows:

- MIDI
  - ☞ Same as the respective **File/Export MIDI** menu item of the main **WEDELMUSIC EDITOR MODULE**.
- **File/Print**

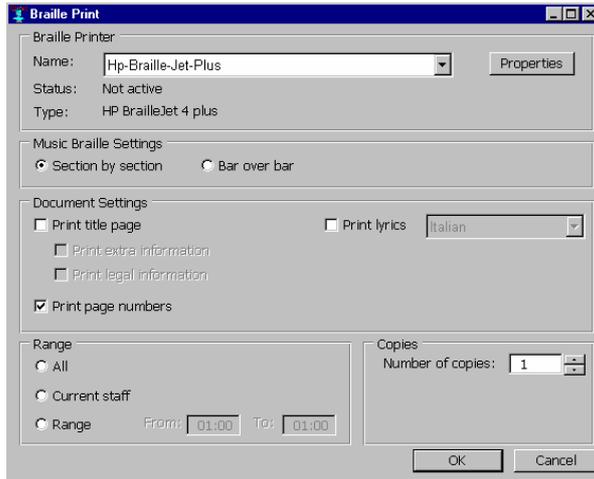
If data are present, i.e. a file has been opened or imported and converted to the WEDELFORMAT, this command routes (a selected part of) the score to the appropriate printer output device in a unified way, whether it is:

  - Braille music (using a proper font)

It is important to note that data in any of these forms can be sent to either a normal paper printer or a Braille printer.

The **File/Print/Braille** dialog

When the user chooses the **File/Print/Braille** the Braille Printer dialog pops up, which allows the modification of the contents of the related Settings objects.



**Figure 14** An indicative version of the Braille printer dialog box.

The Braille printer dialog contains 4 different setting categories, which we will describe:

1. The first section determines the currently selected printer. If more than one printer is available the user can select the required one. There can be two possible device types in the printer selection control, namely: a Braille printer and a conventional printer. The actual type of printer can be read in the Type field. If the type of the current printer is a Braille printer, the interpreted Braille Music is sent to the this printer. If the type of the current printer is a conventional printer, the interpreted Braille Music is printed using a Braille font. By selecting the Properties button next to the active printer selection control, the user can edit the settings of the selected Braille printer. For an indicative example of this Printer settings dialog see the next section *The Braille Printer Driver*.
2. The Braille Music Settings region allows modification of Braille Music Interpretation properties. At this moment only two possibilities exist: Interpretation on basis of 'section by section' format and interpretation on basis of 'bar over bar' format. In the future more properties could be added here. Selection and/or modifications in this area can influence the actual output of the interpretation module.
3. The document Settings region allows selection of additional information to be printed in the Braille document. At this moment the following information can be added to the printed document:
  - a. Print Title page : A title page is inserted at the beginning of the score document on which the name, author and publish date can be displayed. When selecting this option more specific information concerning the title page content become available.
  - b. Print page numbers : On every page a page number is inserted.

- c. **Print lyrics** : When this option is selected the lyrics will be printed (if available) in the selected language. More than one languages can be available in the WEDELOBJECT. The lyric language selection control will be enabled when the Print lyrics option is selected and reflect the available lyric languages in the WEDELOBJECT.
  4. The **Range** area allows selection of the actual object being printed, choices being:
    - a. **All** : The entire score will be printed, including additional information on the title page, if selected.
    - b. **Current Staff** : Only the current staff is printed. In case of a multi-staff score, the user can edit only on staff at the time. The user is able to quickly navigate from one staff to the other. The staff the user is working on in the VIP editor is defined as the current staff.
    - c. **Range** : Allows the user to enter a range of the score in measures. The selected range of the complete score will be printed. [Note: Allow selection of: Current Staff, Current Staff range X Y, All, Range X Y and provide controls to modify ranges]
    - d. The **Copies** region allows selection of the number of copies printed of the score or selection.
- **File/Exit**  
Closes any open or imported files and exits the program. If any changes are made, the user is prompted to save or ignore them.  
 Same as the respective **File/Close** menu item of the main WEDELMUSIC EDITOR MODULE.

#### 9.3.3.2.2 The Edit Menu

The **Edit** menu contains commands for manipulating the caret, defining/manipulating selections, cutting/copying/pasting/deleting objects or groups etc. These are described in more detail in the following:

- **Edit/Part**  
This item leads to a submenu containing commands for importing a part as well as commands for cutting and pasting for the current part:
  - **Edit/Part/Cut**  
 Same as the respective **Edit/Cut a Part** menu item of the main WEDELMUSIC EDITOR MODULE.
  - **Edit/Part/Copy**  
 Same as the respective **Edit/Copy a Part** menu item of the main WEDELMUSIC EDITOR MODULE.
  - **Edit/Part/Paste->After/Before**  
A previously cut part is fetched and placed to the end of the current score. It can then be moved to any desired position using the **Modify/Part/Move** command.  
 Same as the respective **Edit/Paste a Part** menu item of the main WEDELMUSIC EDITOR MODULE.
  - **Edit/Part/Delete**  
 Same as the respective **Edit/Copy a Part** menu item of the main WEDELMUSIC EDITOR MODULE.
- **Edit/Measure**  
This item leads to a submenu containing cutting, copying, pasting and deleting options for the current measure:
  - **Edit/Measure/Cut**  
The active measure or column of measures is cut, removed from the score and stored to a local temporary file or internal memory. This measure will be available for pasting operations through the **Edit/Measure/Paste** command.  
 Same as the respective **Edit/Cut a Measure** menu item of the main WEDELMUSIC EDITOR MODULE.
  - **Edit/Measure/Copy**  
The active measure or column of measures is stored to a local temporary file or

internal memory. This measure will be available for pasting operations through the **Edi t/Measure/Paste** command.

 Same as the respective **Edi t/Copy a Measure** menu item of the main WEDELMUSIC EDITOR MODULE.

- **Edi t/Measure/Paste**  
A previously cut measure or column of measures is inserted either from the respective local temporary file or from the internal memory into the current score.  
 Same as the respective **Edi t/Paste a Measure** menu item of the main WEDELMUSIC EDITOR MODULE.
- **Edi t/Measure/Del ete**  
A previously cut measure or column of measures is inserted either from the respective local temporary file or from the internal memory into the current score.
- 
- **Edi t/Move To**  
This item leads to a submenu containing commands for moving the caret. If there is an active selection and the **Shi ft** key (left or right) is down then the selection is altered accordingly (shrunk or expanded). If there is an active selection and the **Shi ft** key is not down, the symbols are deselected.
  - **Edi t/Move To/Previ ous Part**  
The previous part (if any) is made active. Any further editing operations are directed to that part.
  - **Edi t/Move To/Next Part**  
The next part (if any) is made active. Any further editing operations are directed to that part.
  - **Edi t/Move To/Previ ous Measure**  
Caret is moved to the previous measure (if any).
  - **Edi t/Move To/Next Measure**  
Caret is moved to the next measure (if any).
  - **Edi t/Move To/Obj ect (Part/Measure/S-Measure) . .**  
Move to the selected part, measure of the music score.
- **Edi t/Where Am I ?**  
This item results in the VIP MODULE providing audible feedback concerning the current active location. As discussed earlier, this can be identified by a (measure, offset) pair.

### 9.3.3.2.3 The Insert Menu

The **Insert** menu contains all the commands for inserting any kind of new item in the score. New items may be any of the following:

- parts,
- lyrics,
- measures, and
- label

These are described in more detail in the following.

 Most of these menu items correspond to the items found in the **Symbol s** menu of the main WEDELMUSIC EDITOR MODULE.

- **Insert/Part**  
This item leads to a submenu containing options for new parts to be introduced to the score. The options include:
  - **Insert/Part/New Before**  
A new part is introduced to the score. A dialog box requesting the number of staff lines in the new part is invoked.  
 Similar to the **Edi t/New Part** menu item of the main WEDELMUSIC Editor Module.
  - **Insert/Part/Import**  
A part is imported from a single external SWF file. The source file is selected from the local storage using a typical file-open dialog box.

 Similar to the **Edi t/Import a Part** menu item of the main WEDELMUSIC Editor Module.

- **Insert/Measure**

This command is used to insert a new measure of column of measures just before the active caret measure.

- **Insert/Measure/New Before**

 Same as the **Edi t/New Measure Before** menu item of the main WEDELMUSIC Editor Module.

- **Insert/Measure/New After**

 Same as the **Edi t/New Measure After** menu item of the main WEDELMUSIC Editor Module.

- **Insert/Lyrics**

This command associates a lyrics file to the current part. The LYRICS EDITOR MODULE is invoked to either import an existing file in text format or create it from scratch.

 Same as the respective **Misc/All Lyrics to a Part** menu item of the main WEDELMUSIC EDITOR MODULE.

- **Insert/Label**

This item leads to a submenu containing options for inserting “special” symbols. The submenu may include the following options:

 Similar to the respective **Execution/Labels** menu item of the main WEDELMUSIC EDITOR MODULE.

#### 9.3.3.2.4 The Modify Menu

The **Modify** menu contains all the commands for altering/modifying any kind of item previously inserted in the score. Items may be any of the following:

- parts,
- lyrics,
- layers, etc.

These are described in more detail in the following:

- **Modify/Part**

This item leads to a submenu containing options for modifying parts. The options include:

- **Modify/Part/Split**

 Same as the respective **Edi t/Splitting Part** menu item of the main WEDELMUSIC EDITOR MODULE.

**VIP Specific Notes:** To determine which voice/layer is the current, we can either assume that editing takes place on a single-voice basis or that the active note is used to identify the current voice/layer. The active note is the one immediately following the active caret position.

- **Modify/Part/Merge**

 Same as the respective **Edi t/Merging Parts** menu item of the main WEDELMUSIC EDITOR MODULE.

**VIP Specific Notes:** Selection of the two parts to be fused is probably made through a respective dialog box.

- **Modify/Part/Join**

 Same as the respective **Edi t/Joining Parts** menu item of the main WEDELMUSIC EDITOR MODULE.

**VIP Specific Notes:** Selection of the parts to be joined is probably made through a respective dialog box.

- **Modify/Part/Dissjoin**

 Same as the respective **Edi t/Dissjoin Parts** menu item of the main WEDELMUSIC EDITOR MODULE.

- **Modify/Part/Move**

 Same as the respective **Edi t/Move Parts** menu item of the main WEDELMUSIC EDITOR MODULE.

**VIP Specific Notes:** The part to be moved is the currently active part, i.e. the part being currently edited by the user.

- **Modi fy/Layer**  
This item leads to a submenu containing commands to manipulate layers. It contains the following options:
  - **Modi fy/Layer/Move**  
☞ Same as the respective **Edi t/Move a Layer 1, 2, 3, 4** <-> menu item of the main **WEDELMUSIC EDITOR MODULE**.
- **Modi fy/Lyri cs**  
This item leads to a submenu containing options for modifying lyrics. The options include:
  - **Modi fy/Lyri cs/Edi t**  
☞ Same as the respective **Mi sc/Edi t Lyri c** menu item of the main **WEDELMUSIC EDITOR MODULE**.
  - **Modi fy/Lyri cs/Remove**  
☞ Same as the respective **Mi sc/Remove Lyri c** menu item of the main **WEDELMUSIC EDITOR MODULE**.
- **Modi fy/Speci al**  
This item leads to a submenu containing options for modifying “special” symbols such as clefs and keys or for performing transformations. The submenu may include the following options:
  - **Modi fy/Speci al /Transpose Cl ef**  
☞ Same as the respective **Mi sc/Transpose Cl ef** menu item of the main **WEDELMUSIC EDITOR MODULE**.
  - **Modi fy/Speci al /Transpose Key**  
☞ Same as the respective **Mi sc/Transpose Key** menu item of the main **WEDELMUSIC EDITOR MODULE**.
  - **Modi fy/Speci al /Transl ate**  
☞ Same as the respective **Mi sc/Transl ate** menu item of the main **WEDELMUSIC EDITOR MODULE**.

#### 9.3.3.2.5 The Opti ons Menu

The **Opti ons** menu contains items for accessing settings dialog boxes for each of the components of the VIP MODULE and of components that are relative to it as, for example, execution settings.

The different menu items are described in more detail in the following:

- **Opti ons/Language. . .**  
This is the place for the users to select the desired language for their interaction with the VIP MODULE. There will be three options, namely English, French, and Italian. Since information concerning the language is requested by various modules and components, it is suggested that this information is stored in the system registry. (See §9.4. Additional Thoughts and Comments).
- **Opti ons/Setti ngs**  
Leads to a submenu containing settings commands.
- **Options/Synthetic Voice Options**  
Leads to a dialog box containing settings for the synthetic voice to be used both for providing audible information to the user by the VIP EDITOR MODULE and by the SPOKEN MUSIC MODULE.
- **Opti ons/Brai l l e Pri nter. Setti ngs**  
Leads to a dialog box containing settings specific to the Braille printer. The actual Braille Printer is a standard LPT: device and can be treated as a normal printer. The most important work is the conversion of text and music to the appropriate ASCII codes. This translation task is done by the Braille Interpreter module, which will be described in the next section. The Braille printer allows some basic settings, which can be manipulated using the following dialog. The results are stored in the BRAILLEPRINTERSETTINGS object. The Braille Printer Settings dialog contains the following regions:
  - Margin settings, which allows the modification of the margin definition of the used Braille paper.

← --- Formattati: Elenchi puntati e numerati

- Paper size settings, which allow the modification of the paper size of the used Braille paper. Note that the number of characters that can be printed on a line, can influence the Braille Music Interpretation results.  
The Braille Printer Settings dialog:

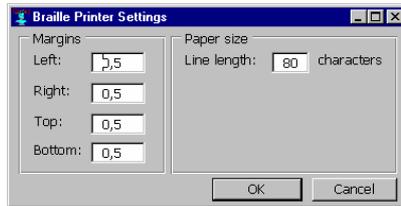


Figure 18. The Braille Printer Settings dialog box.

- **Options/Spoken Music Settings**

Leads to a dialog box containing settings specific to the SPOKEN MUSIC INTERPRETER. Voice settings are accessed through the **Options/Synthetic Voice Options** dialog box and are not duplicated here.

- **Options/Play MIDI**

 Same as the respective **Execution/Play MIDI** menu item of the main WEDELMUSIC EDITOR MODULE.

- **Options/Consistency Check**

 Same as the respective **Consistency Check** menu item of the main WEDELMUSIC EDITOR MODULE.

← --- **Formattati:** Elenchi puntati e numerati

### 9.3.3.2.6 The Execution Menu

The **Execution** menu contains items for controlling reproduction of music pieces to appropriate output devices, the available options being audio and spoken music. The options are described in more detail in the following:

- **Execution/Destination**

Contains mutually exclusive sub-items to select the destination. The available options are:

- **Execution/Destination/Audio**  
Interpreting the symbolic form to music to perform any playback.
- **Execution/Destination/Spoken Music**  
Interpreting the symbolic form to spoken music to perform any playback.

- **Execution/Play**

Leads to a sun index of help terms for the user to select from.

- **Execution/Play/Entire File**  
The entire score is played back.
- **Execution/Play/Selection**  
If there is an active selection, the selected portion of the score is played back.
- **Execution/Play/From Current**  
Playback starts from the caret position to the end of the file.
- **Execution/Play/To Current**  
Playback starts from the beginning of the file to the caret position.
- **Execution/Play/Around Current**  
A small piece of the score around the current caret position is played back.

- **Execution/Compile**

 Same as the respective **Compile** button of the **Execution Tool bar** of main WEDELMUSIC EDITOR MODULE.

- **Execution/Initialise**

 Same as the respective **Initialise** button of the **Execution Tool bar** of main WEDELMUSIC EDITOR MODULE.

- **Execution/Start**

 Same as the respective **Start** button of the **Execution Tool bar** of main WEDELMUSIC EDITOR MODULE.

- **Execution/Pause**  
 Same as the respective **Fermata** button of the **Execution Tool bar** of main **WEDELMUSIC EDITOR MODULE**.
- **Execution/Stop**  
 Same as the respective **Stop** button of the **Execution Tool bar** of main **WEDELMUSIC EDITOR MODULE**.

#### 9.3.3.2.7 The Symbol s Menu

The **Symbol s** menu contains items for performing all editing operations. Through the items of this menu, the user can insert notes, small notes, rests, etc., alter their attributes, delete them etc. Items are also provided for switching between the various layers. The items are described in more detail in the following:

- **Symbol s/Time Signature**
- **Symbol s/Key Signature**
- **Symbol s/Metronome**
- **Symbol s/Add Dynamics**
- **Symbol s/Delete Dynamics**
- **Symbol s/Annotation**
- **Symbol s/General Text**
- **Symbol s/Insert**  
Toggles between the various available noteheads.  
 Includes items that are present in the various toolbars of the main **WEDELMUSIC EDITOR MODULE**.  
It includes the following items for inserting music notation objects:
  - **Repeat**
  - **Note**
  - **Rest**
  - **Accidental**
  - **Augmentation dot**
  - **Refrain Symbols**
  - **Expression**
  - **Ornament**
  - **Fingering**
  - **Mute**
  - **Harmonic**
  - **Breath, Glass**
  - **Strings**
  - **Pedal**
  - **Violin Symbol**
  - **Percussion**
  - **Interval Symbol**
- **Symbol s/Modify...**  
Allows the user to modify the active object, i.e. the object immediately following the caret. The type of the object is automatically determined and an appropriate dialog box containing the active object's attributes is displayed.  
 Same as the respective **Edit** button of the **Editing Tool bar** of main **WEDELMUSIC EDITOR MODULE**.
- **Symbol s/Delete...**  
Allows the user to delete the active object, i.e. the object immediately following the caret. The user is warned and prompted for confirming this action.  
 Same as the respective **Delete** button of the **Editing Tool bar** of main **WEDELMUSIC EDITOR MODULE**.

#### 9.3.3.3 Shortcuts

A shortcut key provides the user of a program the ability to avoid opening menus, entering submenus etc to locate and issue a command. While this feature is considered optional and is

rarely used by novice users, it is **very important** for advanced users and users that have a certain experience on the specific program since it provides the easiest and fastest way to perform operations.

There are (at least) three main types of shortcuts:

- **Key Sequences:** They are usually declared as underline under one letter of a menu item. The user can invoke the specific command by implicitly navigating through the menu lists by pressing a sequence of letters. For example, to save the current file in MS-Word, the user can press the sequence: **Alt+F, Alt+S**. The first key opens the **File** menu from the menu bar, while the second selects the **Save** menu item from the **File** menu list that popped up. Usually, key sequences consist of a list of **Alt+key**.
- **Keyboard Accelerators:** They are explicitly declared in a program when creating the menu. They allow the user to issue command through a single key. For example, to save the current file in MS-Word, the user can press **Ctrl+S**. Usually, keyboard accelerators consist of a **Ctrl+key** or a function key.
- **Hotkeys:** They must be explicitly linked to program commands by providing code to handle them since they reduce to messages sent by the operating system to the application.

It is more than obvious that the first two kinds of shortcuts are trivial to support.

The module does **not** depend on the definition of any kind of shortcuts. The VIP user is constantly assisted in issuing a command by providing audible feedback on every menu and submenu that pops up, every menu item that is highlighted etc.

Thus, the primary purpose of providing a shortcut mechanism is to render the overall system more usable even for non-VIP persons. Of course, VIP persons will also benefit from it.

However, the only way to enable the user to issue a command of a program when this program is not active is through the last kind of shortcuts, i.e. hotkeys. Hotkeys are registered by an application and instruct the operating system to send a message to the registered application even when it does not have the focus.

Keys navigating the score:

- **Left/Right Arrow:** Navigate to the previous/next figure of the music score.
- **Up/Down Arrow:** Navigate the the previous/next layer of the music score.
- **PageUp/PageDown Key:** Go to the previous/next part of the music score.
- **Ctrl+Shift+Tab/Ctrl+Tab:** Go the previous/next measure of the music score.

#### 9.3.3.4 The Multilingual TtS Component

Spoken messages are involved in all the previously described actions, both to assist the VIP users to select the respective commands and to provide feedback on the results of their actions on the currently active score.

It is more than usual that mixed-language messages may need to be conveyed to the user by the WEDELMUSIC VIP EDITOR, e.g. when combining information in the user's language with information in the system's language or in the default language used by a third party application. Those cases are handled by the MULTILINGUAL TtS COMPONENT assisted by the LANGUAGE IDENTIFICATION COMPONENT and the TtS SCHEDULING AND ROUTING COMPONENT described in the next paragraphs.

#### 9.3.3.5 The Language Identification Component

This component is responsible for separating a multilingual text into a series of monolingual pieces. This is achieved by identifying the language, often on a word-by-word basis.

#### 9.3.3.6 The TtS Scheduling and Routing Component

The conversion of multilingual sentences to speech, involves (at least) the following system resources (apart from CPU-time and memory which are automatically shared among different processes and threads by the operating system):

- one or more TtS engines

- a sound card hardware for output (playback) of the synthetic speech

Obviously, controlling the access of each TtS engine to the sound card and initiating/terminating a “job” is a task that needs to be explicitly handled for synthetic speech to be uttered seamlessly. This is the very task of the TTS SCHEDULING AND ROUTING COMPONENT.

Thus, an efficient internal notification scheme is set up to marshal the details of synchronising monolingual TtS engines to transparently integrate them into a single multilingual TtS engine.

### 9.3.3.7 The Screen-Reader Application

Based on the discussion that preceded in the introduction of paragraph §9.1. “*General Issues of the WEDELMUSIC Interface for Blind People*”, it is clear that the presence of a commercial off-the-shelf screen-reader application is a prerequisite for the operation of the WEDELMUSIC VIP EDITOR.

However, no interaction between the module and the screen-reader application takes place (either implicitly or explicitly) and thus no assumption is made concerning the specifications of this screen-reader application. This alleviates the VIP users from any constraints and enables them to work with the screen-reader of their preference.

## 9.3.4 The Braille Printing Module

### 9.3.4.1 Braille Printing Description and Scope

The Braille printing module provides all resources needed for the following activities:

- Setting Braille printing related settings concerning the following components:
  - Braille Printer
  - Braille Music Interpretation
  - Braille Document
- Interpretation of the WEDELTOOLKIT music model to Braille Music.
- Transmitting interpreted musical information to a Braille printer.

### 9.3.4.2 Braille Printing and WEDELMUSIC System

According to the system general architecture described in the TA, the BRAILLE PRINTING MODULE interacts basically with two other components of the WEDELMUSIC System:

- the WEDEL toolkit libraries (or WEDEL navigator);
- the Interface for Blind people (the WEDELMUSIC VIP EDITOR, in particular).

The Braille Printing module uses functions provided by the WEDEL toolkit in order to navigate through the WEDELMUSIC object and in particular to manage components and entities of the music score.

### 9.3.4.3 BRAILLE PRINTING MODULE Architecture

The BRAILLE PRINTING MODULE is basically composed of two logical sub-modules:

- the Braille Printing Interpreter,
- the Braille Printer.

The **BRAILLE PRINTING INTERPRETER** “translates” the symbolic music notation contained in the symbolic music file into a “braille” text in concordance with the *New international Manual of Braille Music Notation*. This text will be sent to the Braille printer.

The **BRAILLE PRINTER** contains all objects that the Braille Printing Module needs to communicate with local and network (Braille) printers. An important detail is the fact, that the Braille Printer module uses the standard Windows Spool to provide the printers with data. This data will be interpreted Braille Printing Module expressed in ASCII strings. The BraillePrinterSettings class contains hardware specific details of the BraillePrinter.

### 9.3.4.4 Braille Printing Module Data Structure

The interpretation component of the Braille printer module is a resident in memory.

In the VIP MODULE layout all modification actions in specific dialogs actually modify the appropriate properties in their corresponding Settings objects. Thus an explicit separation is made between the WEDELTOOLKIT (containing the original settings), The visual dialogs and the Setting objects, which have a role as settings buffer and are part of the application logic. The BRAILLEMUSICINTERPRETER is able to access all components containing properties, that can influence the Braille Music interpretation.

Main goal is to achieve maximum decoupling of the different modules in the VIP part of the project. To achieve this object representations are defined for the different application layers. Below the different object layers are defined:

- 1) Devices: hardware represented in the form of an object. Each object manages it's own persistence (reading and writing of settings etc.) and provides methods to obtain specific device settings. Also the actual transmission of data is handled by these objects. Devices such as audio and printers, typically are predefined OS objects.
- 2) Device settings: settings used by a device object. The values for these settings are provided by the WEDELTOOLKIT. We have to decide to directly obtain the settings from the WEDELTOOLKIT or access the settings via the proposed Settings object layer. The values stored in the device setting objects are retrieved from the appropriate parameter blocks available in the WEDELOBJECT
- 3) Non-visual components: BRAILLEMUSICINTERPRETER and related objects are example of non-visual components. The only way the user interacts with these components is done using the appropriate setting dialogs (if available). All other interactions with these objects is done transparently for the user.
- 4) Visual components: Some of the non-visual components have properties, that can be manipulated by the user. These properties are accessed through an 'accessible' interface as defined in the *Accessibility Rules for VIP User Interface*. This means that these modifiable components have visual counterparts. These visual counterparts are used to manipulate the related component's settings. The names of the visual components are directly related to the names of their non-visual counterparts. For example: BRAILLEPRINTERSETTINGS (a non-visual object) and BRAILLEPRINTERSETTINGSDIALOG (the visual dialog).
- 5) Rules: The BRAILLEMUSIC interpretation mechanism is represented using a meta-system. Interpretation knowledge is hard coded into objects.

#### 9.3.4.5 The Braille Interpreter

In BRAILLEMUSIC everything is influenced by context. The actual representation of even a single note is determined by the location of that note in the score: i.e. the musical function of that note. Not only the musical context has influence on the BRAILLEMUSIC interpretation. Also the size of paper the Braille music will be printed on and the amount of extra information on that Braille sheet has an impact on transcription decisions. The resources available to print Braille are limited and each individual note requires at least two Braille cells. Every Braille music interpretation rule has as one of it's aims to limit the amount of Braille information actually printed to the sheets. This in order to keep the music readable and not confuse the user with an avalanche of information.

The BRAILLEMUSIC interpretation procedure consists of three steps:

- 1) Retrieve parameters concurring the current object in the score. These parameters are parameters describing the current context of the object. Parameters can have values like: pith number, duration value, boolean values (has the current note a dot?, is the current note a member of a slur?, etc)
- 2) Couple specific combinations of required parameter values to specific actions. These actions represent the Braille music transcription. Which parameters are required by the rule is determined in the definition of the rule. Each rule requests the parameters it requires using the WEDELTOOLKIT interface methods. A BRAILLEMUSIC interpretation rule is associated to the object type of the queried WEDELOBJECTS. In this way specific rules can be applied to specific types of objects.
- 3) Evaluate the parameter value combination associated action. The result of the action evaluation is an addition of specific Braille cells in a buffer. Which Braille cells will be

inserted and in which order is determined by the action definition. The action definition is an 'implementation' of a Braille music rule as described in the *New international Manual of Braille Music Notation*. This buffer is a list in which new Braille cells or combinations of Braille cells can be inserted in specific locations. In this way these module generate a sentence in BRAILLEMUSIC 'grammar'. The content of this buffer will be sent to the Braille printer.

Four *sources* of parameters are defined in The Braille Interpreter:

- 1) The structural musical information, which is available via the WEDELTOOLKIT. The structural information can be obtained by means of methods, which enable the interpreter to query the several properties of a WEDEL object. The interpreter can also query an note object about it's location in the score and retrieve information about it's context. It can, for example, request a reference to it's neighbouring note or other object and query information of that note, by means of the same methods used to query itself.
- 2) The information about paper resources, which is available through the BRAILLEPRINTERSETTINGS object. Examples of parameters provided by this object are: Margins, Page height, Line length.
- 3) The information about additional information printed to the paper, which is available through the DOCUMENTSETTINGS object. Examples of parameters provided by the object are: Print page numbers, Print copyright info, Print first page, etc.

Specific explicit properties of the BRAILLEMUSIC interpretation procedure, which is available through the BRAILLEMUSICINTERPRETERSETTINGS object. Examples of such settings are: interpretation method (bar-over-bar, section-by-section), accuracy (skip rules of a specific priority)

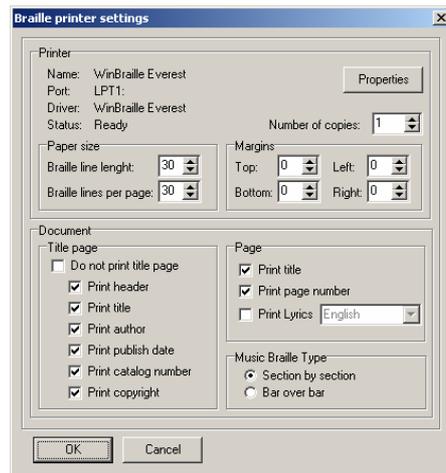
### 9.3.4.6 Interface with WEDELTOOLKIT

In order to provide the functionalities and features described before, the Braille Printing module needs to read through the WEDEL symbolic file (which contains the symbolic music notation). Functions allowing this navigation and the retrieving of information regarding the content of the symbolic file are provided by the WEDEL toolkit, and constitutes a shared Library supporting operation of the various VIP Interface components.

### 9.3.4.7 Implemented Functionality

## Braille Printing

The user can choose menu **File/Print/Braille** for Braille Printing. After that the Braille Printer dialog will bring up, which allows the modification of the contents of the related Settings objects.



Braille printer dialog box.

The Braille printer dialog contains 3 different setting categories:

1. The first section determines the currently selected printer. By selecting the Properties button next to the active printer selection control, the user can edit the settings of the selected Braille printer.  
The Printer region contains also the following regions:
  - Margin settings, which allows the modification of the margin definition of the used Braille paper.
  - Paper size settings, which allows the modification of the paper size of the used Braille paper. Note that the number of characters that can be printed on a line, can influence the Braille Music Interpretation results.
  - The Copies setting, which allow selection of the number of copies printed of the score.
2. The Music Braille Type region allows modification of Braille Music Interpretation properties. There are two possibilities: Interpretation on basis of 'section by section' format and interpretation on basis of 'bar over bar' format.
3. The Document Settings region allows selection of additional information to be printed in the Braille document. The following information can be added to the printed document:
  - a. Print Title page : A title page is inserted at the beginning of the score document on which the title, author, publish date, catalog number and copyright can be displayed.
  - b. Print title : On every page a title is inserted.
  - c. Print page numbers : On every page a page number is inserted.
  - d. Print lyrics : When this option is selected the lyrics will be printed (if available) in the selected language. More than one languages can be available in the WEDELOBJECT. The lyric language selection control will be enabled when the Print lyrics option is selected and reflect the available lyric languages in the WEDELOBJECT.

When user select Ok button, the interpreted Braille Music will be sent to the Braille printer. If the score involves more than one instrument, the user can select the instrument part he wants to print.

### **9.3.5 The Spoken Music Module**

#### *9.3.5.1 Spoken Music Description and Scope*

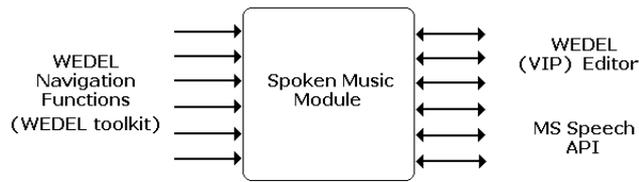
The Spoken Music is a component of the WEDELMUSIC system which performs basically the following task: given a specific score in WEDEL symbolic notation, representing a certain music piece, it converts the (symbolic) music notation into text and then "speaks" the text using the "text-to-speech" module.

Among the main features of the Spoken Music Module are: possible selection of different languages, selection of the segment to be spoken, selection of the desired score, retrieving of information about the piece to be "spoken", navigation through the segments, replay of the last segment.

#### *9.3.5.2 Spoken Music and WEDELMUSIC System*

According to the system general architecture described in the TA, the SPOKEN MUSIC MODULE interacts basically with two other components of the WEDELMUSIC System (see figure 19):

- the WEDEL toolkit libraries (or WEDEL navigator);
- the Interface for Blind people (the WEDELMUSIC VIP EDITOR, in particular).



**Figure 19**

The Spoken Music module uses functions provided by the WEDEL toolkit in order to navigate through the WEDELMUSIC object and in particular to manage components and entities of the music score (reading of single bars, reading of specific instrument parts, reading of score information ...).

The Interface for Blind calls the functions of the Spoken Library provided by the Spoken Music Module, in order to manage the “speaking” of music for blind. Besides, the Interface for Blind provides all feedback and facilities to allow VIP users to exploit the Spoken Music functionalities.

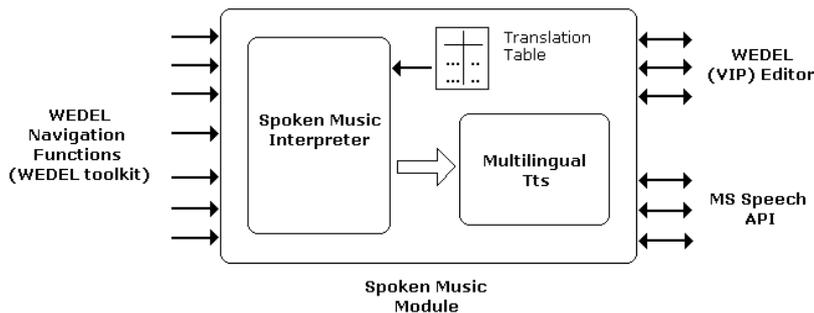
Furthermore, the Spoken Music module uses the functions provided by the standard library MS Speech API, in order to have a common standard interface to Text-to-speech (Tts) engines.

The same Tts engines used for the WEDELMUSIC VIP EDITOR can be used also for the Spoken Music.

### 9.3.5.3 SPOKEN MUSIC MODULE Logical Architecture

The SPOKEN MUSIC MODULE is basically composed of two logical sub-modules (Figure 20):

- the Spoken Music Interpreter,
- the Multilingual Tts.



**Figure 20** An overview of the Spoken Music Module.

The **SPOKEN MUSIC INTERPRETER** “translates” the symbolic music notation contained in the symbolic music file into a “speakable” text. This text describes in a suited and concise way, the music as it is written on the score.

Translation is performed according to a set of translation rules, enabling the module to effectively build intelligible phrases on the basis of the symbolic music notation.

The **SPOKEN MUSIC INTERPRETER** makes use of a table (a translation table), in order to translate symbols expressed in the WEDELMUSIC notation into music notation and then into words.

The **MULTILINGUAL TTS MODULE** (not to be confused with the respective component of the VIP EDITOR MODULE) allows to easily and smoothly switch between two different search engines for different languages, among those provided. This module is needed since MS Speech API does not provide this necessary feature.

### 9.3.5.4 Spoken Music Module Data Structure

In the following a list and a brief description of some data structures used by the Spoken Music modules:

### **Translation Table**

A table containing an entry for each music symbol (or the most part of those) provided by the WEDELMUSIC model. This table allows the translation of symbolic music notation into text.

### **Current Position**

This is a structure containing the current position of the Spoken Music description (that is the point currently described by the Spoken Music module, or the point where the description was stopped and from what it will restart). It is composed by two fields which indicated the current measure and the offset within it:

```
typedef struct
{
    int battuta;
    int scarto;
} pos
```

### **Current Segment**

The segment currently described by the Spoken Music module, or the last segment described, if the description is not running.

#### *9.3.5.5 Interface with WEDELTOOLKIT*

In order to provide the functionalities and features described before, the Spoken Music module needs to read and navigate through the WEDEL symbolic file (which contains the symbolic music notation).

Functions allowing this navigation and the retrieving of information regarding the content of the symbolic file are provided by the WEDEL toolkit, and constitutes a shared Library supporting operation of the various VIP Interface components.

In the following are listed the functions which allow the navigation through the WEDEL Object score:

- *SetSegLen(int m)*: sets the length of the segment equals to m measures
- *GoToSeg(int s)*: set the specified segment as the current one
- *GoToSegForward(int d)*: move the current position d segments forward
- *GoToSegBackward(int d)*: move the current position d segments backward

#### *9.3.5.6 Implemented Functionality*

### **Selection of different languages**

The user can choose in which language, among those provided, he wants to listen to the Spoken Music. This choice does not depend on the language used for the Interface for Blind. The user can decide to have the language of the Interface for Blind set to English and listen to the Spoken Music in Italian, if he wants.

### **Retrieving and speaking of information about the piece to be “spoken”**

The user can listen to a description of the music piece corresponding to the WEDEL object currently open. This description is composed of:

- the name (title) of the piece,
- the author(s),
- the length (in bars and segments),
- all instruments involved in the score,
- global characteristics of the piece (timing, key signature, ...),
- every further information provided by the symbolic notation regarding the entire score.

The user listens to this description at the beginning of the “speaking” of the piece and he can also require this information at any moment.

### Selection of the desired instrument

If the score involves more than one instrument, the user has to select the instrument part he wants to listen to. Selection of the desired instrument involves a feedback from the Spoken Music module (or even from the Interface for Blind): all instruments present in the music piece are described (via speech) to the user to allow him to choose.

### Selection of the type of segment of music score to be spoken (number of bars/measures for segment)

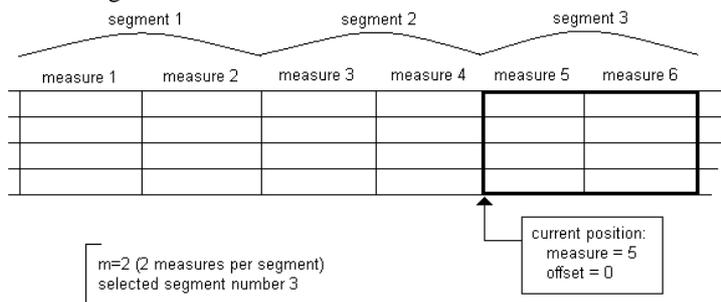
As previously stated the Segment is the basic unit of music score to be spoken (and managed). Music is spoken segment by segment and the user is able to navigate through segments to select the segment (or the sequence of segments) to be spoken.

A segment is defined in terms of number of measures (bars): it is composed of a sequence of  $m$  measures, where  $m$  is user-defined.

### Selection of the segment of music score to be spoken (number of the segment)

Once the segment type is defined ( $m$  is set), the music piece can be regarded as divided in a certain number of segments. In such a way the user can select the segment to be spoken, simply referring to a number (from 1 to the number of segments the music piece is composed of). For instance, the user can choose to listen to the Spoken Music for the segment number 3. The Spoken Music module will speak the third segment of the music piece, starting from the  $[(3*m)+1]^{\text{th}}$  measure to the  $[3*(m+1)-1]^{\text{th}}$  measure included.

Figure 21 shows an example of part of a score (without notes) with  $m=2$  (two measures per segment), where the segment number 3 is selected.



**Figure 21** An example of part of a score (without notes) with  $m=2$  (two measures per segment), where the segment number 3 is selected.

### Navigation through the segments (go to next segment, go to previous segment)

The user can navigate through segments, in order to listen to the description of the desired part(s) of the music piece via the Spoken Music. The navigation can be performed using two high-level commands, such as "Next segment" and "Previous Segment" (the name of them is self-explaining). This two commands returns a feedback to the user (a spoken feedback) stating the current position within the music score.

Since for Spoken Music the basic unit of the score is the segment, position can be given as the number of segment. However, to avoid confusion the feedback regarding the current position can be presented in the same way as for the "current location" in the VIP editor, that is in terms of the pair (current measure, offset within the measure).

Anyway, navigation through the music score for Spoken Music is performed in terms of segments: the user can go to the next segment, to the previous one or to the segment number  $x$ . When the user go to the next segment, for instance, the current position becomes: (first measure of the next segment, offset = 0). See example in the previous figure.

If the user wants to have a high resolution for navigation within the score, the segment length should be fixed equal to one measure ( $m=1$ ).

Navigation can be combined with the speaking of the current segment or can stand alone, having only a feedback regarding the current position.

### **Play (speak) of the entire music piece (for the selected instrument) segment by segment**

This feature allows the user to listen to the description of the selected part/instrument of the music score from the beginning to the end. The score is spoken segment by segment, in order to easily allow the user to keep trace of the current position within the score. At the beginning of each segment the number of the segment is announced to the user.

Besides, before starting the description of the first measure of the score (the first measure of the segment number one), the Spoken Music module “speaks” the description of the entire music piece corresponding to the WEDEL object currently open. This description is composed of:

- the name (title) of the piece,
- the author(s),
- the length (in bars and segments),
- all instruments involved in the score,
- global characteristics of the piece (timing, key signature, tonality).

Then the Spoken Module proceeds with the description of the music notation for the various segments, in sequence.

The spoken description of the music notation continues until one of the following conditions occur:

- the music score for the selected part/instrument ends,
- the user issues the “Stop” command.

In the first case, the current position becomes (measure=1, offset=0) that is, the first segment.

In the second case the current position will be (first measure of the current segment, offset=0).

### **Play (speak) of a single segment (current segment)**

The user can listen to the description of the current segment. This feature can be used in combination with the navigation through segments in order to listen only the desired segments of the music score. The user sets the current position according to his needs then issues the command for the speaking of the current segment.

### **Play (speak) of the music piece from the current position (bar or segment) to the end**

The user can listen to the description of the music score starting from the current position. Again, the spoken description of the music notation continues until one of the following conditions occurs:

- the music score for the selected part/instrument ends,
- the user issues the “Stop” command.

The current position at the end of the operation will be set according to the termination condition. In the first case, the current position becomes (measure=1, offset=0) that is, the first segment. In the second case the current position will be (first measure of the current segment, offset=0).

### **Play (speak) of the music piece from the beginning to the current position (bar or segment)**

In this case the user listens to the description of the music score starting from the beginning until the current position. The Spoken Music description ends if one of the following conditions occur:

- the description of the music score reaches the current segment (included),
- the user issues the “Stop” command.

The current position at the end of the operation will be set according to the termination condition.

It should be noted that, if the Spoken Music description is stopped by the user before reaching its planned end (remind that this operation is Play To current), the current position for the WEDEL VIP editor does not match the current position traced in the Spoken Music module. This is not a problem, since the current position can be easily re-synchronized forcing the current position in the Spoken Music module to assume the value of the current position for the WEDEL VIP editor, using the proper function (see functions description in the following).

To better explain this concept, the following example can be useful. Assume that the user has navigated through the score with the WEDEL VIP editor, and now the current position (hold in a data structure, in the WEDEL VIP editor) is:

(measure=5, offset=4)

The user now issues the command “Play to current” to listen to the description of the music piece from the beginning to the current position.

The description of the music starts, from the beginning of the music score (measure=1, offset=0) and as the description proceeds, the current position for the Spoken Music, hold in a data structure in the Spoken Music module, increases from (1,0) toward (5,4).

If the user stops the description (issuing the “Stop” command) before the description reaches the position (5,4) then the current position for Spoken Music will be:

(x,y)

where  $1 \leq x < 5$  and  $0 \leq y < 4$

and then:

$(x,y) \neq (5,4)$

that is, the current position for Spoken Music module does not match the current position for WEDEL Vip editor.

To re-synchronize them it is necessary that the WEDEL Vip editor calls the Spoken Music library function SetCurrentPos(5,4), forcing in this way the current position for the Spoken Music module to assume the value (5,4).

The value of the current position for the WEDEL editor remains the same during all the proceeding of the music description and can be changed only inside the WEDEL editor environment.

Figure 22 shows the example described above.

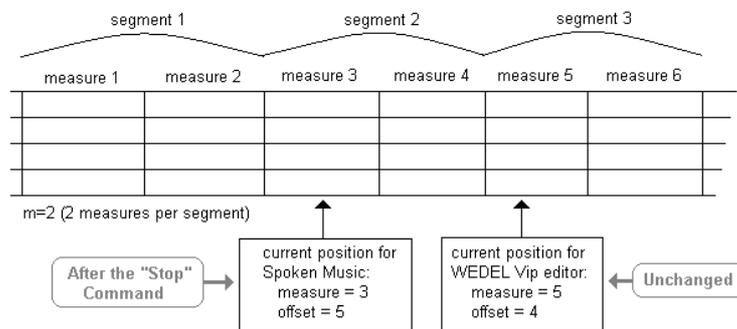


Figure 22.

**Play (speak) of the music piece around the current position (from some bar or segment before to some bar or segment after the current position)**

This feature allows the user to listen to a spoken description of what stands around the current position. Obviously, the meaning of “around of the current position” should be defined in terms of number of measures (or segments) before and after the current position.

This feature can be very useful to give the user a more detailed description of the current position, rather than a simple pair of numbers (current measure, offset).

As a limit, this feature matches the “Play current segment” feature described before.

### Play (speak) of a selected part of the music piece (from bar/segment to bar/segment)

This is a powerful feature because it allows the user to freely choose the part of the music score to be spoken. This operation needs two parameters (generally speaking):

- start position,
- end position.

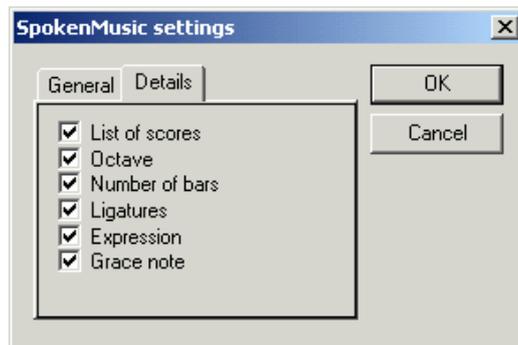
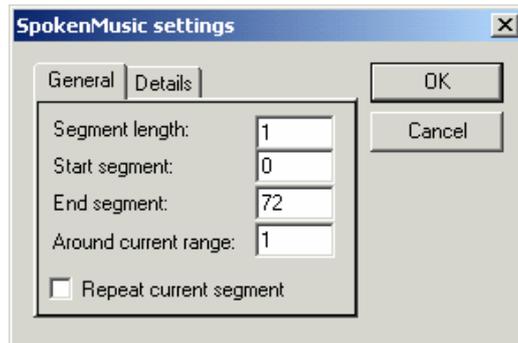
These positions are given according to the specification of position (or location) inside the music score, that is, in the form of a pair (measure, offset inside the measure).

As a limit, this feature coincides from one hand with the play of the entire music piece, from the other with the play of a single segment.

#### 9.3.5.7 The SPOKEN MUSIC User Interface Component

##### Settings

A user can decide the details level of the speech selection choosing the “Spoken Music...” item in the Vip Settings submenu of the Options menu. A new dialog window appears:



By means of the first pane, the user can set the following preferences:

- Segment length: number of bars for each segment.
- Start segment: the segment which the Spoken Music starts to read the score from.
- End segment: the last segment to be read from the Spoken Music.
- Around current range: how many previous and following segments are to be read when the “Around Current” play mode is selected.
- Repeat current segment the user can listen to the repetition of the current segment (until he/she stops the speech) checking this box.

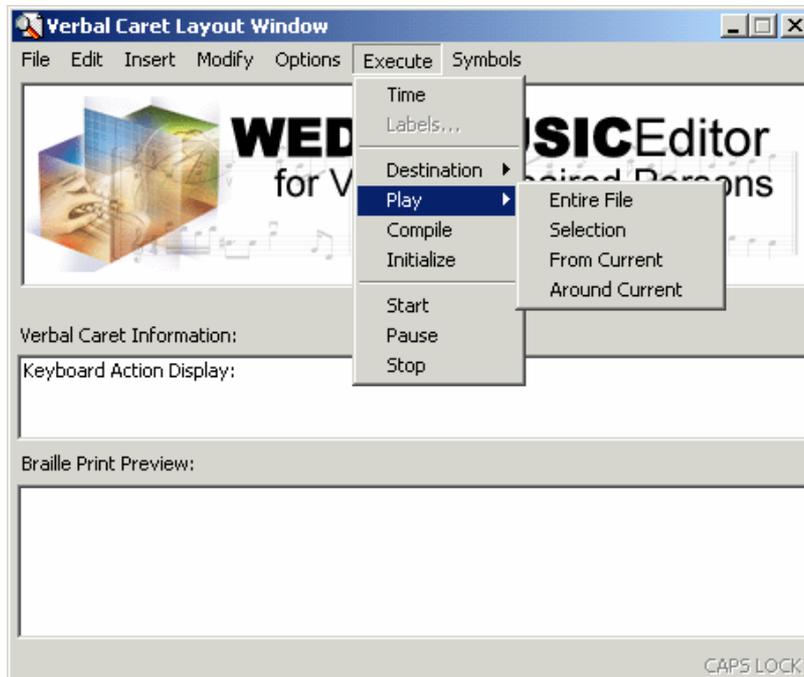
The second pane, instead, allow the user to decide the detail level of the music speaking.

- List of scores: at the beginning of the execution, the SpokenMusic utters the list of the scores which are to be read.
- Octave: for each note, its octave is specified.
- Number of bars: everytime the SpokenMusic read a new bar, its number is uttered.
- Ligatures: the module specify every legature found.

- Expression: the SpokenMusic utters the musical expressions: accento, staccato, tenuto and so on.
- Grace note: the SpokenMusic specify the found abbellimentos: trillo, mordente, glissato and so on.

### Play Range

To set the score range to be played (i.e. which bars are to be played), the user has to select the desired option among the ones listed in the “Play” submenu of the “Execute” menu.



The user can choose one play mode among those listed below:

- Entire file: the score is executed entirely.
- Selection: only the segments in the selection defined in the SpokenMusic settings, that is from the start segment to the end one (see above), are read.
- From Current: the score reading starts from the start segment defined in the SpokenMusic settings (see above) are read.
- Around Current: the score reading starts from “Around current range” segments before the current one to “Around current range” segments after the current one, where the “Around current range” value is defined in the SpokenMusic settings (see above).

### Execution

The user can start, pause and stop the SpokenMusic execution using the matching items in the “Execute” menu.

- Start: the SpokenMusic module begins to read the score according to the preferences set in the “Settings” and “Play range” sections.
- Pause: the SpokenMusic module pauses to read the score; when the execution is started again, it begins from the last segment it was reading.
- Stop: the SpokenMusic module stops to read the score; when the execution is started again, it begins to the first segment according to the preferences set in the “Settings” and “Play range” sections.

## 9.4 Implementation Strategy

## 9.4.1 Conditions Needed for the Development

### 9.4.1.1 Commercial Screen-Reader.

For the development and testing phases, a commercial screen-reader was purchased and installed at the development site(s). This helped in identifying and curing all the problems, special conditions and incompatibilities emerging by the simultaneous execution of the developed module and the screen-reader.

### 9.4.1.2 TtS Engines for the Project Languages.

There is an increasing number of TtS engines for a variety of languages that are compatible with Microsoft's SPEECH API. For the multilingual TtS module to be developed, engines for each of the project's language need to be found and probably purchased. It is quite certain that such engines exist for the target languages and it will even be possible to download some for free.

### 9.4.1.3 The WEDELOBJECT.

- The WEDELMUSIC VIP EDITOR's implementation heavily relies on the WEDELOBJECT (Object-Oriented Model), since the EDITOR is actually nothing more than an accessible interface for the functionality implemented in the WEDELOBJECT.

## 9.4.2 Interrelations With Other Modules/Components

### 9.4.2.1 General

**The WEDELOBJECT.** The WEDELMUSIC VIP EDITOR's implementation heavily relies on the WEDELOBJECT (Object-Oriented Model), since the EDITOR is actually nothing more than an accessible interface for the functionality implemented in the WEDELOBJECT. See also the comments on the WEDELOBJECT under paragraph §9.3.1. "*Conditions Needed for the Development.*"

The same is also true for the implementation of the BRAILLE INTERPRETER and the SPOKEN MUSIC INTERPRETER.

The scope of the SPOKEN MUSIC MODULE is clearly not limited to VIP persons. It is a tool that can provide important aid to sighted persons as well. Thus, it should also be accessible from the main WEDELMUSIC EDITOR.

Although Braille interpretation is primarily addressed to VIP persons, it is conceivable that a sighted person makes use of it. For example, in the case that a sighted person needs to pass musical information to a VIP person, it would be quite necessary for a VIP-friendly format to be employed, such as Braille printing format. Thus, the sighted persons are likely to require Braille printing abilities from their main WEDELMUSIC EDITOR.

The VIP EDITOR is probably the only module that needs not be accessible from the main WEDELMUSIC EDITOR since it provides redundant options rather than additional functionalities.

### 9.4.2.2 The Braille Module

We can define the following interrelations with other (internal or external) modules and components:

- Relation and Communication between VIP EDITOR and BRAILLE INTERPRETER MODULE. All external modules can call the public `String interpret(WDFObjectID objID)` or `String interpret(WDFObjectID startObjID, WDFObjectID endObjID)` to invoke the interpretation procedure or send an appropriate Windows Command to this module. The exact types of arguments and return value processing have to be agreed.
- Relation and Communication between VIP EDITOR and BRAILLE PRINTER DEVICE. All external modules can call the public `String print(WDFObjectID objID)` or `String print(WDFObjectID startObjID, WDFObjectID endObjID)` to invoke the interpretation procedure or send an appropriate Windows Command to this module. The exact types of arguments and return value processing have to be agreed.

The BRAILLEPRINTER device will delegate all needed sub-calls to the rest of the relevant Braille Printer module components to execute the interpretation and transmission of data to the printer.

- Relation and Communication between MAIN EDITOR and BRAILLE INTERPRETER MODULE. All external modules can call the public String interpret(WDFObjectID objID) or String interpret(WDFObjectID startObjID, WDFObjectID endObjID) to invoke the interpretation procedure or send an appropriate Windows Command to this module. The exact types of arguments and return value processing have to be agreed.
- Relation and Communication between MAIN EDITOR and BRAILLE PRINTER DEVICE. All external modules can call the public String print(WDFObjectID objID) or String print(WDFObjectID startObjID, WDFObjectID endObjID) to invoke the interpretation procedure or send an appropriate Windows Command to this module. The exact types of arguments and return value processing have to be agreed. The BRAILLEPRINTER device will delegate all needed sub-calls to the rest of the relevant BRAILLE PRINTER MODULE components to execute the interpretation and transmission of data to the printer.

### 9.4.3 Technology Used

#### 9.4.3.1 Screen-Reader

A commercial screen reader has been employed. (see links paragraph, reference [1] for a list of products and pricing)

#### 9.4.3.2 Microsoft SPEECH API

This API (Application Program Interface) is a collection of objects and methods to handle tasks from text-to-speech synthesis to speech recognition and dictation, to command and control, etc. It receives increasing attention in the field becoming a standard to which more and more companies come to conform. Numerous engines are available by third parties for various languages. Most screen-readers are already compliant with SAPI.

Using MS-SAPI ensures that the WEDELMUSIC VIP EDITOR application will benefit from its advanced characteristics and remain on the state-of-art of the TtS technology.

### 9.4.4 Implementation Platform and Development Tools

The WEDELMUSIC VIP EDITOR MODULE was implemented using Visual C. This choice was made taking into account both the advanced features and system-level programming requirements posed by the specific tasks involved in this module and the team's expertise.

Needless to say that the implementation was carried out strictly on the Microsoft Windows environment since it was the only way to utilise the SPEECH API and other system resources indispensable for this module.

## 9.5 Appendix: Index of Signs in Standard Braille Order

Numbers in parentheses are Table numbers. The table reference is followed by a Paragraph reference and, in italics, a Topic in the Index where additional references are located.

In the following diagram the sixty-three braille characters are arranged in seven rows, each sign having a reference number. The Index is divided into paragraphs, each headed by its own number.

Signs with two or more characters will be found in the paragraph headed by the number of the initial character (e.g. ⠠⠠ is under 3; ⠠⠠⠠ is under 31).

Diagram of the 63 Braille Characters

<b>Signs 1-10:</b>	⠠	⠡	⠢	⠣	⠤	⠥	⠦	⠧	⠨	⠩
<b>Signs 11-20:</b>	⠪	⠬	⠭	⠮	⠯	⠰	⠱	⠲	⠳	⠴
<b>Signs 21-30:</b>	⠶	⠷	⠸	⠹	⠺	⠻	⠼	⠽	⠾	⠿
<b>Signs 31-40:</b>	⠼⠠	⠼⠡	⠼⠢	⠼⠣	⠼⠤	⠼⠥	⠼⠦	⠼⠧	⠼⠨	⠼⠩
<b>Signs 41-50:</b>	⠼⠪	⠼⠬	⠼⠭	⠼⠮	⠼⠯	⠼⠰	⠼⠱	⠼⠲	⠼⠳	⠼⠴
<b>Signs 51-56:</b>	⠼⠶	⠼⠷	⠼⠸	⠼⠹	⠼⠺	⠼⠻	⠼⠼	⠼⠽	⠼⠾	⠼⠿
<b>Signs 57-63:</b>	⠼⠼⠠	⠼⠼⠡	⠼⠼⠢	⠼⠼⠣	⠼⠼⠤	⠼⠼⠥	⠼⠼⠦	⠼⠼⠧	⠼⠼⠨	⠼⠼⠩

Index of Signs in Standard Braille Order

Numbers in parentheses are references to the Tables of Signs

Hyphenated numbers are chapter-paragraph references

Italics indicate topics from the main Index

1. ⠠
  - ⠠ First Finger (8A 8B) 8-1 see *fingerings*
  - ⠠ Left Toe (15B) 15-29 see *organ pedals*
  - ⠠ Ring finger *anular* (guitar) (8B) 8-10 see *fingerings*, “*pima*”
  - ⠠ Left hand (percussion) (18B) 18-14 see *percussion*
2. ⠠
  - ⠠ Second Finger (8A 8B) 8-1 see *fingerings*
  - ⠠ Left Heel (15B) 15-29 see *organ pedals*
  - ⠠ Two syllables sung on one note (16B) 16-8 see *vocal text*
3. ⠠
  - ⠠ Slur between two notes or chords (6A) 6-2 see *slurs*
  - ⠠ Syllabic slur (16B) 16-7 see *vocal music*, *slurs*
  - ⠠ Between fingerings, change of fingers on one note (8A 8B) 8-2 see *fingerings*
  - ⠠ Between foot signs, change of pedalling on one note (15B) 15-29 see *organ pedals*
  - ⠠ Syllabic slur for the first language (16B) 16-9 see *vocal text*
  - ⠠ Syllabic slur for the second language (16B) 16-9 see *vocal text*
  - ⠠ Syllabic slur for the third language (16B) 16-9 see *vocal text*
  - ⠠ Phrasing slur (6A 16B) 6-3(a) see *slurs*
  - ⠠ Syllabic slur for the fourth language (16B) 16-9 see *vocal text*
4. ⠠
  - ⠠ C eighth and 128th (1A) 1-1 see *note values*
5. ⠠
  - ⠠ D eighth and 128th (1A) 1-1 see *note values*
6. ⠠
  - ⠠ E eighth and 128th (1A) 1-1 see *note values*
7. ⠠

- 8. F eighth and 128th (1A) 1-1 see *note values*
- 9. G eighth and 128th (1A) 1-1 see *note values*
- 10. A eighth and 128th (1A) 1-1 see *note values*  
Index finger *indice*(guitar) (8B) 8-10 see *fingerings*, “*pima*”
- 11. B eighth and 128th (1A) 1-1 see *note values*
- Fifth finger (8A) 8-1 see *fingerings*
  - Open string (8B 17D) see *string instruments*
  - Natural harmonic (17D) 17-16 see *harmonics*
  - Dotted bar line (9A) 9-2 see *bar lines*, *dotted*
  - Letter O over or under note (winds) (18A) 18-2 see *wind instruments*
  - Change of organ pedalling without indication of toe or heel (15B) 15-29 see *organ pedals*
- 12. Third finger (8A 8B) 8-1 see *fingerings*
  - Right toe (15B) 15-29 see *organ pedals*
  - Three syllables sung on one note (16B) 16-8 see *vocal text*
  - Bar line (9A) 9-1 see *bar lines*
  - Right hand (percussion) (18B) 18-14 see *percussion*
- 13. Measure rest (1A) 1-7 see *rests*
  - Whole and 16th rest (1A) 1-1 see *rests*
  - Middle finger *mayor* (guitar) (8B) 8-10 see *fingerings*, “*pima*”
  - Brevis rest (1A) see *rests*
  - Longa rest (1A) see *rests*
- 14. C half and 32nd (1A) 1-1 see *note values*
- 15. D half and 32nd (1A) 1-1 see *note values*
- 16. E half and 32nd (1A) 1-1 see *note values*
- 17. Thumb *pulgar* (guitar) (8B) 8-10 see *fingerings*, “*pima*”
- 18. F half and 32nd (1A) 1-1 see *note values*
- 19. G half and 32nd (1A) 1-1 see *note values*
- 20. A half and 32nd (1A) 1-1 see *note values*
- 21. B half and 32nd (1A) 1-1 see *note values*
- 22. Half and 32nd rest (1A) 1-1 see *rests*

- Quarter and 64th rest (1A) 1-1 see *rests*
- 23. 
  - Eighth and 128th rest (1A) 1-1 see *rests*
  - Little finger *chiquito* (guitar) (8B) 8-10 see *fingerings*, "*pima*"
- 24. 
  - C whole and 16th (1A) 1-1 see *note values*
  - C Brevis (1A) see *note values*
  - C Longa (1A) see *note values*
- 25. 
  - D whole and 16th (1A) 1-1 see *note values*
- 26. 
  - E whole and 16th (1A) 1-1 see *note values*
- 27. 
  - F whole and 16th (1A) 1-1 see *note values*
- 28. 
  - G whole and 16th (1A) 1-1 see *note values*
- 29. 
  - A whole and 16th (1A) 1-1 see *note values*
- 30. 
  - B whole and 16th (1A) 1-1 see *note values*
- 31. 
  - Natural (3A) 3-1 see *accidentals*
  - End of original segno passage (9C) 9-47 see *braille segno*
  - Star or other indication for pedal up (15A) 15-18 see *pedalling*, *piano*
- Thumb (strings) (8B) 8-8 see *fingerings*
  - Artificial harmonic (17D) 17-16 see *harmonics*
  - Star and ped. under one note (15A) 15-20 see *pedalling*, *piano*
  - Beginning of repeat in cadenza or unmeasured music (9C) 9-32 see *cadenzas*
  - Swell; diverging and converging lines on one note (10) 10-13 see *nuances*
- 32. 
  - Flat (3A) 3-1 see *accidentals*
  - Down bow (17D) 17-24 see *string instruments*, *bowed*
  - Plectrum downstroke (17 D) 17-35 see *string instruments*, *plucked*
  - "Draw" (accordion) (19A) 19-12 see *accordion*
  - Plus sign over or under note (winds) (18A) 18-3 see *wind instruments*
  - Ped. or other indication for pedal down (15A) 15-17 see *pedalling*, *piano*
  - Double bar at end of composition (9A) see *double bar lines*
  - Double bar at end of bar or section (9A) see *double bar lines*
  - Fermata, pause (10) 10-13 see *nuances*
  - Double flat (3A) 3-1 see *accidentals*
  - Separation of rhythmic groups (1A) 4-8 see *rhythmic groups*
  - Distinction of values (1A) 4-8 see *note values*

- Print double bar with dots preceding; end of repeat (9B) 9-5 see *print repeats*
- Print double bar followed by dots; beginning of repeat (9B) 9-6 see *print repeats*
- Low tremolo (19B) 19-6 see *accordion music, registration*
- Full measure in-accord (5B) 5-11 see *in-accords*
- Up bow (17D) 17-24 see *string instruments, bowed*
- Plectrum upstroke (17D) 17-35 see *string instruments, plucked*
- "Push" (accordion) (19A) 19-12 see *accordion*
- 33. ••
- Sharp (3A) 3-1 see *accidentals*
- First string (17A) 17-8 see *string instruments*
- Second string (17A) 17-8 see *string instruments*
- Fifth string (17A) 17-8 see *string instruments*
- Third string (17A) 17-8 see *string instruments*
- Double sharp (3A) 3-1 see *accidentals*
- Fourth string (17A) 17-8 see *string instruments*
- Sixth string (17A) 17-8 see *string instruments*
- High tremolo (19B) 19-6 see *accordion music, registration*
- Seventh string (17A) 17-8 see *string instruments*
- 34. ••
- C quarter and 64th (1A) 1-1 see *note values*
- 35. ••
- D quarter and 64th (1A) 1-1 see *note values*
- 36. ••
- E quarter and 64th (1A) 1-1 see *note values*
- 37. ••
- F quarter and 64th (1A) 1-1 see *note values*
- 38. ••
- G quarter and 64th (1A) 1-1 see *note values*
- 39. ••
- A quarter and 64th (1A) 1-1 see *note values*
- 40. ••
- B quarter and 64th (1A) 1-1 see *note values*
- 41. ••
- Fourth finger (8A 8B) 8-1 see *fingerings*
- Right heel (15B) 15-29 see *organ pedals*
- 42. ••
- Triplet (4) 4-10 see *rhythmic groups*
- 43. ••
- Seventh interval (5A) 5-1 see *intervals*
- Prefix for a chord symbol part (12A) 12-2 see *chord symbols*
- 44. ••
- Turn between notes (11) 11-9 see *ornaments*
- Small circle in chord symbols (12A) 12-14 see *chord symbols*

-  Inverted turn between notes (11) 11-11 see *ornaments*
-  Small circle bisected by line in chord symbol part (12A) 12-14 see *chord symbols*
- 45. 

 Short appoggiatura (11) 11-8 see *ornaments*  
 Variant of measure (sign precedes and follows) (9D) 9-55 see *variant passages*
-  Black note head with no stem (13A) 13-3 see *notes, with unusual shapes*
-  X-shaped note head (13A) 13-2 see *notes, with unusual shapes*
-  Vertical stems that indicate approximate pitch (13A) 13-8 see *notes, with unusual shapes*
-  Diamond- or triangle-shaped note head (13A) 13-2 see *notes, with unusual shapes*
-  Variant of two (or other number) measures (9 D) 9-55 see *variant passages*
-  End of slanting line to designate approximate pitch (13A) 13-9 see *approximate pitch*
- 46. 

 Trill (11) 11-2 see *ornaments*
- 47. 

 Measure or part-measure repeat (9C) 9-15 see *braille repeats*  
 Equals sign in metronome marking (14) 14-17 see *equals sign*  
 Part-measure repeats of different value (9C) 9-19 see *braille repeats*  
 Repeat 4 (or other number) times (9C) 9-36 see *braille repeats*
- 48. 

 Staccato (10) 10-13 see *nuances*  
 Merged text sung on one note (16D) 16-8 see *vocal text*
- 49. 

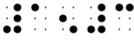
 Fifth interval (5A) 5-1 see *intervals*  
 Repeat text once (16D) 16-13 see *vocal text*  
 Repeat vocal text twice (16D) 16-13 see *vocal text*
- 50. 

 Sixth interval (5A) 5-1 see *intervals*  
 Small triangle in chord symbol part (12A) 12-14 see *chord symbols*  
 Small triangle bisected by line in chord symbol part (12A) 12-14 see *chord symbols*
- 51. 

 Second interval (5A) 5-1 see *intervals*
- 52. 

 Third interval (5A) 5-1 see *intervals*  
 Print segno (9B) 9-9 see *print segno*  
 Braille segno (9C) 9-47 see *braille segno*  
 Coda sign (9B 9C) 9-9 see *segno sign*
- 53. 

 Fourth interval (5A) 5-1 see *intervals*  
 A line of continuation in figured bass (12B) 12-23 see *figured bass*

- 
 Two lines of continuation in figured bass (12B) 12-23 see *figured bass*
- 
 Numbered variant, followed by number of measures (9D) 9-56 see *variant passages*
- 
 Second (or other number) variant of measures (9D) 9-56 see *variant passages*
- 
 Repeat four measures (9C) 9-39 see *braille repeats*
- 
 Repeat four measures (9C) 9-39 see *braille repeats*
- 
 First ending (9B) 9-6 see *print repeats*
- 
 Repeat measure 1 (or other number) (9C) 9-42 see *braille repeats*
- 
 Repeat group of measures (9C) 9-42 see *braille repeats*
- 
 Second ending (9B) 9-6 see *print repeats*
- 
 Repeat measure 2 (or other number) (9C) 9-42 see *braille repeats*
- 
 Figured bass numeral (12B) 12-16 see *figured bass*
- 
 Figured bass numerals (12B) 12-16 see *figured bass*
- 
 Oblique stroke in figured bass (12B) 12-27 see *figured bass*
- 
 Blank space in a string of figures (12B) 12-21 see *figured bass*
- 
 Before a figure, any number that is crossed in print (figured bass) (12B) 12-20 see *figured bass*
- 54. 
 Word sign; literary material (10) 10-2 see *nuances*
- 
 End of barré sign when it is not followed by a fret sign (17C) 17-34 see *barrés*
- 
 Bass solo (B.S.) (19A) 19-4 see *accordion*
- 
 Diverging lines (crescendo) (10) 10-5 see *nuances*
- 
 Converging lines (decrescendo) (10) 10-5 see *nuances*
- 
 Arpeggio up (11) 11-20 see *ornaments*
- 
 Arpeggio down (11) 17-36 see *ornaments*
- 
 Register (19A) 19-4 see *accordion*
- 
 Without register; use print initials (19A) 19-4 see *accordion*
- 
 Comma, half breath (10 16C) 16-11 see *breath signs*
- 
 Neither draw nor push, cesur (19A) 19-12 see *accordion*
- 
 End of diverging lines (10) 10-5 see *nuances*
- 
 End of converging lines (10) 10-5 see *nuances*
- 
 Music asterisk (14) 14-16
- 
 Fifth position or fret (17B) 17-8 see *string instruments*
- 
 Sixth position or fret (17B) 17-8 see *string instruments*
- 
 Second position or fret (17B) 17-8 see *string instruments*
- 
 G clef in left hand part (2) 2-5 see *clef signs*
- 
 G clef; treble (2) 2-1 see *clef signs*
- 
 G clef with small 8 above (2) 2-7 see *clef signs*
- 
 G clef with small 8 below (2) 2-7 see *clef signs*
- 
 G clef on first line; French violin clef (2) 2-6 see *clef signs*
- 
 Third position or fret (17B) 17-8 see *string instruments*
- 
 C clef; alto clef for viola or high clef for bass (2) 2-1 see *clef signs*
- 
 C clef on first line; soprano or descant clef (2) 2-6 see *clef signs*
- 
 C clef on second line; mezzo-soprano clef (2) 2-6 see *clef signs*

-  C clef on third line; alto clef for viola or high clef for bass (2) 2-6 see *clef signs*
-  C clef on fourth line; tenor clef (2) 2-6 see *clef signs*
-  C clef on fifth line; baritone clef (2) 2-6 see *clef signs*
-  Fourth position or fret (17B) 17-8 see *string instruments*
-  16 ft. (19B) 19-6 see *accordion music, registration*
-  4 ft. (19B) 19-6 see *accordion music, registration*
-  4 ft. 16 ft. (19B) 19-6 see *accordion music, registration*
-  4 ft. 8 ft. (19B) 19-6 see *accordion music, registration*
-  4 ft. 8 ft. 16 ft. (19B) 19-6 see *accordion music, registration*
-  Tremolo (19B) 19-6 see *accordion music, registration*
-  Double tremolo (19B) 19-6 see *accordion music, registration*
-  8 ft. 16 ft. (19B) 19-6 see *accordion music, registration*
-  8 ft. (19B) 19-6 see *accordion music, registration*
-  F clef in right hand part (2) 2-5 see *clef signs*
-  F clef; bass (2) 2-1 see *clef signs*
-  F clef on third line; tenor clef (2) 2-6 see *clef signs*
-  F clef on fifth line; sub bass clef (2) 2-6 see *clef signs*
-  First position or fret (17B) 17-8 see *string instruments*
-  Half position (17B) 17-8 see *string instruments*
-  End of first line of continuation (10) 10-16 see *nuances*
-  End of second line of continuation (10) 10-16 see *nuances*
-  Eighth position or fret (17B) 17-8 see *string instruments*
-  Ninth position or fret (17B) 17-8 see *string instruments*
-  Tenth position or fret (17B) 17-8 see *string instruments*
-  Eleventh position or fret (17B) 17-8 see *string instruments*
-  Twelfth position or fret (17B) 17-8 see *string instruments*
- 55. 

  -  After note or rest; dotted note or rest 1-2 see *note values*
  -  Omission of second fingering when two sets are given (8A) 8-5 see *fingering*
  -  After note or rest; double dotted note or rest 1-2 see *note values*
  -  Beginning of first line of continuation (10) 10-16 see *nuances*
- 56. 

  -  Octave interval (5A) 5-1 see *intervals*
  -  Parallel motion (9C) 9-50 see *parallel motion*
  -  Sequence abbreviation (9C) 9-52 see *sequence abbreviation*
  -  Separation of signs (figured bass) (12B) 12-25 see *figured bass*
  -  Beginning of second line of continuation (10) 10-16 see *nuances*
- 57. 

  -  First octave (1B) 1-8 see *octave marks*
  -  First row of buttons (19A) 19-2 see *accordion, rows*
  -  Vertical bracket barré (17C) 17-33 see *barrés*
  -  Glissando (6A 17B) 17-15 see *string instruments*
  -  Shift to new position (17B) 17-13 see *shift lines*
  -  Beginning of shift (17B) 17-14 see *shift lines*
  -  Single-note tie (6B) 6-13 see *ties*
  -  Foot crosses in front (organ) (15B) 15-30 see *organ pedals*



-  Fermata on bar line (10) 10-13 see *nuances*
-  3/4 step alteration of a sharp (3A) 13-16 see *accidentals*
-  32nd stem (5D) 5-31 see *stems*
-  Rhythmic grouping (4) 4-10 see *rhythmic groups*
-  Agogic accent (10) 10-13 see *nuances*
-  Prefix for left hand part (15A) see *keyboard music*
-  Left hand pizzicato (17D) 17-26 see *string instruments*
-  Left hand part with intervals reading down (15A) 15-11 see *keyboard music*
-  Whole stem (5D) 5-31 see *stems*
-  Single-note tie between in-accord parts (6C) 6-22 see *ties*
-  Chord tie between in-accord parts (6C) 6-23 see *ties*
- 60. 
-  Fourth octave (1B) 1-8 see *octave marks*
-  Fourth row of buttons (19A) 19-2 see *accordion, rows*
-  Hyphen at end of unfinished measure (14) 14-8 see *dot 5*
-  Prefix for added signs in braille (14) 5-15 see *dot 5*
-  Slur from one staff to another (6A) 6-7 see *slurs*
-  Prefix for editorial markings (6A) 14) 14-15 see *editorial markings*
-  Pedal up as soon as chord is struck (15A) 15-22 see *piano music, pedalling*
-  Half pedal (15A) 15-21 see *piano music, pedalling*
-  Fermata between notes (10) 10-13 see *nuances*
-  Part-measure in-accord (5B) 5-17 see *in-accords*
-  Prefix for pagination or print page turn (14) 14-9 see *page turn*
-  Appoggiatura (11) 11-7 see *ornaments*
-  Short trill (11) 11-13 see *ornaments*
-  Mordent (11) 11-15 see *ornaments*
-  Mezzo-staccato (dot and line) (10) 10-13 see *nuances*
-  Repeat back to segno A (or B, etc.) (9C) 9-47 see *braille segno*
-  Solo part to be accompanied (15A) 15-5 see *keyboard accompaniments*
-  Upward arpeggio through two or more staves (11) 11-20 see *ornaments*
-  Downward arpeggio through two or more staves (11) 11-20 see *ornaments*
-  Vertical brackets surrounding notes or features (12C) 12-40 see *brackets*
-  Single-note tie between staves (6C) 6-23 see *ties*
-  Chord tie between staves (6C) 6-23 see *ties*
- 61. 
-  Fifth octave (1B) 1-8 see *octave marks*
-  Fifth row of buttons (19A) 19-2 see *accordion, rows*
-  Alternation in 64ths (7B) 7-6 see *alternation-type tremolo*
-  Alternation in 8ths (7B) 7-6 see *alternation-type tremolo*
-  Chord tie (6B) 6-17 see *ties*
-  C "common" time (3B) 3-7 see *time signatures*
-  Measure-division for in-accord (5B) 5-17 see *in-accords*





## 9.6 Appendix: Index of Signs in Standard Braille Order

### 9.6.1 TABLE 1 - Basic Signs

#### 9.6.1.1 A. Notes and Rests

Throughout this work:

- whole note or rest = semibreve
- half note or rest = minim
- quarter note or rest = crotchet
- 8th note or rest = quaver
- 16th note or rest = semi-quaver (etc.)

C	D	E	F	G	A	B	Rest
							Wholes or 16ths
							Halves or 32nds
							Quarters or 64ths
							8ths or 128ths
							Prefix for 256th notes, i.e.,  (etc.)
							Distinction of values
							Separation of rhythmic groups
							Larger values; 8ths and larger
							Smaller values; 16ths and smaller
							Brevis, i.e.,  (etc.)
							Brevis rest
							Longa, i.e.,  (etc.)
							Longa rest
							Notes printed in large type
							Notes printed in small type

(For modern notation, see Table 13A)

#### 9.6.1.2 B. Octave Marks

	First octave C
	Second octave C
	Third octave C
	Fourth octave C
	Fifth octave C
	Sixth octave C
	Seventh octave C
	A below first octave
	C above seventh octave

#### 9.6.2 Table 2 - Clefs

	G clef; treble
	G clef in the left hand part
	F clef; bass
	F clef in the right hand part
	C clef; alto clef for viola or high clef for bass
	G clef with small 8 above



## 9.6.5 Table 5 - Chords

### 9.6.5.1 A. Intervals

••	Second	••••	Sixth
•••	Third	•••••	Seventh
••••	Fourth	••••••	Octave
•••••	Fifth		

••••••••	Ex. of Unison
••••••••••	Ex. of Tenth (interval larger than octave)

(For tone clusters, see Table 13 B)

### 9.6.5.2 B. In-accords

••••••	Full-measure in-accord
•••••	Part-measure in-accord
••••	Measure division for part-measure in-accord

### 9.6.5.3 C. Moving Notes

•••••	Moving-note sign for one interval
••••••	Moving-note sign for two or more intervals

### 9.6.5.4 D. Stem Signs

••••••••	Whole "stem"	••••••••••	8th stem
••••••	Half stem	••••••••••••	16th stem
•••••	Quarter stem	••••••••••••••	32nd stem

## 9.6.6 Table 6 - Slurs & Ties

### 9.6.6.1 A. Slurs

•••••	Slur between two notes or chords
••••••••	Phrasing slur over more than four notes or chords
••••••••••	Phrasing slur over more than four notes or chords
••••••••••	Beginning and end of phrasing slurs on one note
•••••	Beginning and end of short slur on one note
•••••	Slur from one in-accord part to another
•••••	Slur from one staff to another
•••••	Straight line between staves for voice leading
•••••	End of straight line
•••••	Slur added by an editor in print
•••••	Slur that does not end on a note
•••••	Slur for short appoggiatura; "grace-note slur" in some countries
•••••	Slur that does not come from a note
•••••	Half phrase
•••••	Glissando

### 9.6.6.2 B. Ties

•••••	Single-note tie
•••••	Chord tie (two or more voices are tied)

⠠⠠ Accumulating arpeggio

### 9.6.6.3 C. More Slurs and Ties for Use in Section-by-section Formats

⠠⠠⠠ Slur from another in-accord part  
 ⠠⠠⠠ Slur from another staff  
 ⠠⠠⠠ Single-note tie between in-accord parts  
 ⠠⠠⠠ Single-note tie between staves  
 ⠠⠠⠠ Chord tie between in-accord parts  
 ⠠⠠⠠ Chord tie between staves

To clarify that the sign is “from”

⠠⠠⠠⠠ Single-note tie from another in-accord  
 ⠠⠠⠠⠠ Single-note tie from another staff  
 ⠠⠠⠠⠠ Chord tie from another in-accord  
 ⠠⠠⠠⠠ Chord tie from another staff

## 9.6.7 Table 7 - Tremolos

### 9.6.7.1 A. Repetition

⠠⠠ In quarters  
 ⠠⠠ In 8ths  
 ⠠⠠ In 16ths  
 ⠠⠠ In 32nds  
 ⠠⠠ In 64ths  
 ⠠⠠ In 128ths

### 9.6.7.2 B. Alternation

⠠⠠ In 8ths  
 ⠠⠠ In 16ths  
 ⠠⠠ In 32nds  
 ⠠⠠ In 64ths  
 ⠠⠠ In 128ths

## 9.6.8 Table 8 - Fingering

### 9.6.8.1 A. Keyboard Instruments

⠠ 1 First finger                      ⠠ 4 Fourth finger  
 ⠠ 2 Second finger                   ⠠ 5 Fifth finger  
 ⠠ 3 Third finger  
 ⠠ Between fingering, change of fingers on same note  
 ⠠ Omission of first fingering when 2 sets are given  
 ⠠ Omission of second fingering when 2 sets are given

### 9.6.8.2 B. String Instruments

#### 9.6.8.2.1 1. Left Hand

⠠⠠ Thumb                              ⠠ 3 Third finger  
 ⠠ 1 First finger                      ⠠ 4 Fourth finger  
 ⠠ 2 Second finger                   ⠠ 0 Open string

- 
- ⠠ Between fingering, change of fingers on same note

#### 9.6.8.2.2 2. Right Hand

- 
- Thumb *pulgar*
- Index finger *indice*
- Middle finger *mayor*
- Ring finger *anular*
- Other single letters; braille as printed
- Little finger *chiquito* if ch is printed

### 9.6.9 Table 9 - Bar Lines & Repeats

#### 9.6.9.1 A. Bar Lines

- (space) Bar line
- Braille bar line for special uses
- Dotted bar line
- Double bar at end of composition
- Double bar at end of bar or section

#### 9.6.9.2 B. Print Repeats

- Double bar followed by dots; beginning of repeat
- Double bar with dots preceding; end of repeat
- First, second ending (volta)
- Print segno sign
- Print coda sign

#### 9.6.9.3 C. Braille Repeats

- Full- or part-measure repeat
- Separation of repeats with different values, i.e., •• ••
- Repeat beginning in fifth (or other number) octave
- Repeat four (or other number) times
- Beginning of repeat in cadenza or unmeasured music
- Count back and repeat measures
- Repeat the last four (or other number) measures
- Repeat the last four (or other number) measures
- Repeat specific measures
- Repeat specific measures from a numbered section  
(i.e., repeat measures 9-16 from Section 2)
- Braille segno A (or B, etc.)
- Repeat back to segno A (or B, etc.)
- End of segno music to be repeated
- Coda sign
- Parallel motion
- Sequence: Continue the pattern

#### 9.6.9.4 D. Variants

- Variant of measure (sign precedes and follows)

- Variant of two (or other number) measures
- Second (or other number) variant of measure
- Numbered variant, followed by number of measures  
(i.e., Variant 1, for 3 measures)

**9.6.10 Table 10 - Nuances**

- Staccato (dot)
- Staccatissimo (pear-shaped dot)
- Mezzo-staccato (dot & line)
- Agogic accent (short line)
- Accent (horizontal "V" pointing to the right)
- Reversed accent (horizontal "V" pointing to the left)
- Martellato (thick "V")
- Comma, half breath
- Full break or breath
- Fermata
- Fermata on bar line
- Fermata between notes
- Fermata with square shape
- Fermata with tent shape
- Diverging lines (crescendo) ••••• End of diverging lines
- Converging lines (decrescendo) ••••• End of converging lines
- Swell, diverging & converging lines on one note
- Beginning of first line of continuation
- End of first line of continuation
- Beginning of second line of continuation
- End of second line of continuation
- Word sign; literary material

**9.6.11 Table 11 - Ornaments**

-  Trill [It. trillo; F. cadence, tremblement; G. Triller; Sp. trino]
-  Appoggiatura [It.] [F. port de voix; G. Vorschlag; Sp. apoyadura]
-  Short appoggiatura (formerly: grace note)
-  Turn [It. fioritura; F. double, cadence, double cadence; G. Doppelschlag; Sp. grupito] between notes
-  Turn above or below a note
-  Inverted turn between notes
-  Inverted turn above or below a note
-  Before an accidental, inflected lower note of ornament  
Ex. ••••• •••••
-  Short trill, [short shake, pralltriller, inverted, i.e., upper,

	mordent]
	Extended short trill, shake or pralltriller
	Mordent [It. mordente; F. mordant (pincé, pincement); Sp. mordiente]
	Extended mordent
	Arpeggio up
	Upward arpeggio through two or more staves
	Arpeggio down
	Downward arpeggio through two or more staves
	Bebung

### 9.6.12 Table 12 - Theory

#### 9.6.12.1 A. Chord Symbols

	Prefix for a chord symbol part
	Small circle
	Small circle bisected by line
	Small triangle
	Small triangle bisected by line
	Italicized 7 for a specialized seventh chord

Use national signs of country for plus, minus and oblique stroke

#### 9.6.12.2 B. Figured Bass and Harmonic Analysis

	Prefix for a figured bass part
	Numbers
	Isolated accidentals
	Omitted figure in a string of figures
	A line of continuation
	Two lines of continuation
	Figure (any number) that is crossed in print
	Oblique stroke
	Separation of signs
	Italicized 7 for a specialized seventh chord

Use national signs of country for upper- and lower-case letters as well as upper- and lower-case roman numerals

#### 9.6.12.3 C. Brackets

	<b>Music parentheses (round brackets)</b>
	Vertical brackets surrounding notes or features
	<b>Square bracket above the staff</b>
	Square bracket above staff with unclear ending
	<b>Dotted square bracket above the staff</b>
	<b>Square bracket below the staff</b>
	Square bracket below staff with unclear ending
	<b>Dotted square bracket below the staff</b>

### 9.6.13 Table 13 - Modern Notation

(To be used only in modern music)

#### 9.6.13.1 A. Notes with Unusual Shape

- Black note head with no stem
  - X-shaped note head
  - Vertical stems that designate quasi-notes
  - Diamond or triangle-shaped note head
  - End of a slanting line to designate approximate pitch (quasi-note)
- When no specific note value appears, the value of an 8th note is used. i.e., ••••••••

#### 9.6.13.2 B. Tone Clusters

(placed between a note and its interval sign)

- Tone cluster with natural sign
- Tone cluster with flat sign
- Tone cluster with sharp sign
- Tone cluster with no accidentals specified
- i.e., Play every note in this octave as a chord

#### 9.6.13.3 C. "Fan-shaped" Rhythmic Groups

- Accelerando within rhythmic group
- Ritardando within rhythmic group
- Steady rhythm
- End of rhythmic group as shown in print

### 9.6.14 Table 14 - General Organization

- Prefix for music; music parentheses
- Prefix for literary material such as vocal text, instructions, or other literary information
- Coincidence of notes in more than one part
- Hyphen for unfinished measure
- Prefix for added signs in braille
- Prefix for print pagination or print page turn
- Prefix for editorial markings
- Music asterisk
- Equals sign in a metronome marking

### 9.6.15 Table 15 - Keyboard Music

#### 9.6.15.1 A. Piano

- Right hand part
- Left hand part
- Solo part to be accompanied
- Right hand part when intervals read up
- Left hand part when intervals read down
- Ped. or other indication for pedal down
- Star or other indication for pedal up
- Star and Ped. under one note

- Half-pedal
- Pedal up as soon as chord is struck

### 9.6.15.2 B. Organ

- Organ pedal part
- Start of passage when pedal and left hand parts are written on same print staff
- Return to left hand only; pedal drops out
- Left toe (wedge ^ beneath a note)
- Left heel (u or o beneath a note)
- Right toe (wedge ^ above a note)
- Right heel (u or o above a note)
- Between foot signs, change of pedalling on one note
- Change without indication of toe or heel
- Foot crosses in front (dash \_ above toe or heel sign)
- Foot crosses behind (dash \_ beneath toe or heel sign)

### 9.6.16 Table 16 - Vocal Music

#### 9.6.16.1 A. Prefixes for Parts

- Vocal text
- Soprano
- Alto
- Tenor
- Bass
- First Soprano (or other voice part)
- Second Soprano (or other voice part)

#### 9.6.16.2 B. Slurs

- Syllabic slur
- Syllabic slur for first & second language
- Syllabic slur for third & fourth language
- Syllabic slur variation for single verses in one language.
- Beginning and ending of phrase
- Two syllables sung on one note
- Three syllables sung on one note

#### 9.6.16.3 C. Breath Signs

- Half breath
- Full breath

#### 9.6.16.4 D. Text Signs

- Merged text sung on one note
- Repeat text once i.e., ••••
- Repeat text twice
- or •••• Repeat text three (or other number) times

**9.6.17 Table 17 - String Instruments**

**9.6.17.1 A. String Signs**

- 1st string      •••• 5th string
- 2nd string    •••• 6th string
- 3rd string    •••• 7th string
- 4th string

**9.6.17.2 B. Positions or Frets**

- 1st position/fret      •••• 7th position/fret
- 2nd position/fret     •••• 8th position/fret
- 3rd position/fret     •••• 9th position/fret
- 4th position/fret     •••• 10th position/fret
- 5th position/fret     •••• 11th position/fret
- 6th position/fret     •••• 12th position/fret
- Half position
- Glissando or shift to a new position
- Beginning of shift line
- End of shift line

**9.6.17.3 C. Barrés**

- Grand or full barré
- Half or partial barré
- Vertical bracket barré
-       
-        

**9.6.17.4 D. Other Signs**

- Natural harmonic or open string
- Artificial harmonic
- Down bow or down stroke
- Up bow or up stroke
- Left hand pizzicato

**9.6.18 Table 18 - Winds & Percussion**

**9.6.18.1 A. Wind Instruments**

- Letter O over or under note
- Plus sign over or under note

**9.6.18.2 B. Percussion**

- Right hand
- Left hand

**9.6.19 Table 19 - Accordion**

**9.6.19.1 A. General signs**

- Prefix for accordion music

Rows of buttons

- First row (dash below the note)
- Second row (no indication)
- Third row (1 or M)
- Fourth row (2 or m)
- Fifth row (3, 7 or S)
- Sixth row (4 or d)
- Seventh row (varying indications)
- Eighth row (varying indications)
- Row zero (varying indications)
- Draw (V pointing left)
- Push (V pointing right)
- Cesur (Neither draw nor push)
- Bass solo (B.S.)
- Register
- Without Register (use printed initials)

9.6.19.2 B. Registration Signs

		8 ft.; circle with a dot between the 2 cross-lines
		16 ft.; circle with a dot below the 2 cross-lines
		4 ft.; circle with a dot above the 2 cross-lines
		4 ft. 8 ft. 16 ft.; circle with a dot above, one between and one below the 2 cross-lines
		8 ft. 16 ft.; circle with a dot between the 2 cross-lines and one below
		4 ft. 8 ft.; circle with a dot above the 2 cross-lines and one between.
		4 ft. 16 ft.; circle with a dot above the 2 cross-lines and one below
		Tremolo; 2 horizontal dots between the 2 cross-lines
		Double tremolo; 3 horizontal dots between the 2 cross-lines

		"High tremolo"; a dot to the right between the 2 cross-lines or to the right of the center dot
		"Low tremolo"; a dot to the left between the 2 cross-lines or to the left of the center dot.
		Example of combination of more tremolos

## 9.7 Abbreviations

Abbreviation	Meaning
TtS	Text-to-Speech
MS-SAPI	Microsoft's SPEECH API

## 9.8 References

- (LAZZARO, 1999) JOSEPH J. LAZZARO, "Helping the Web Help the Disabled," *IEEE Spectrum*, March 1999.
- (KROLICK, 1996) KROLICK, "The Braille Music Subcommittee World Blind Union," *New International Manual of Braille Music Notation*, 1996.