Chapter 3

System design of DAPHNE

3.1 Design principles

For musical information processing, no matter the purpose is analysis or performance generation, the importance of using musical structures has been emphasized in a lot of researches [Yan88] [Des90]. As a computer-assisted system, DAPHNE obtains analysis information like musical structures. Such information can be further used by automatic performance generation system to achieve abundant expressions [Hig99-b].

In DAPHNE, the following analyses can be performed: musical structures, relationships among occurrences of musical structures, structure functions (grouping by upbeat, accent, and desinence), tonal, harmony, and musical forms. These analyses can be done automatically or manually [Liu98-a]. The results can be obtained quickly and easily. Based on the automatically obtained results, users can further analyze some items manually. Useful information for accompaniment system is also provided. All these information can be stored in a performance data file, which can be further used by the performance system. Moreover, the visualization of performance data makes music analysis easier and more accurate. It is possible to compare and evaluate the automatic performance generated by the performance system.

In order to make the analysis process smoother, a flexible WYSIWYG style user interface is adopted. Users are able to analyze and obtain results on actual scores directly. To a greater or lesser extent, users, especially music professionals, won't feel inconvenience or obstacles when using DAPHNE [Liu99-b].

Besides, the "multi-version scores" and "performance replay" utilities are implemented. For one piece with different versions of scores, DAPHNE can generate different analysis results and make comparison between them. These utilities are designed for ensuring the correctness of the analysis.
3.2 Analysis items

DAPHNE provides users the following analysis items by now. Most of the items are important for generating expressive performance [Liu98-b].

3.2.1 Musical structures

Musical structures are the fundamentals of performance [Cro98] [Dra91] [Wal87]. We adopted Hugo Riemann (1849 – 1919)'s Phrasing theory, which says that several levels of structures compose a musical work [Ong81]. The structures include:

- **Motif** - The smallest recurring linear units. Composed by at least one measure. Usually two measures long.
- **Phrase** - The longer units. Composed by at least four measures. Usually two motifs long.
- **Sentence** - The longest units. Composed by at least eight measures. Usually two phrases long.

It can be explained by using an example (Chopin Mazurka Op. 7 No. 3). Figure 2 shows a part of the score and Figure 3 shows its analyzed occurrences of musical structures (Precisely, some occurrences have *Aufakt*, which means upward movement of conductor's baton or hand, especially to indicate beat preceding bar line.).

![Mazurka Op. 7, No. 3 (measure 9-24)](image)

Figure 2. Musical structures on the score (Chopin Mazurka Op. 7 No. 3 m9-m24)
These occurrences are the basic components of the piece. Musical expressions are associated with these structures. In order to generate expressive performance, we need to analyze the musical structures first.

Since there are exceptions in both the size and the meaning of structure units, users can also specify their own structure unit and its occurrence besides the standard units.

For a music analysis system, the structure analysis is the basis. Different systems use different structure analysis methods. Based on the *Phrasing theory*, DAPHNE’s own method was developed. The algorithms of automatic analysis can be logically expressed by using the equation described below.

First, it is necessary to define some notations for the various elements of a piece.

\[ S: \text{Music piece} \]
\[ \sigma : \text{Musical structure} \]
\[ N: \text{Note} \]
\[ \tau : \text{Variable for a set of notes} \]
\[ \tau = <N_{11}, N_{12}, \ldots> \]

A piece is composed by musical structures according to the *Phrasing theory*.

\[ S = <\sigma_1, \sigma_2, \ldots, \sigma_m> \]

And each musical structure is composed of smaller structures.

\[ \sigma_i = <\sigma_{i1}, \sigma_{i2}, \sigma_{i3}, \ldots, \sigma_{in}> \]

The smallest structure is composed by measures.

\[ \sigma_j = <\sigma_{j1}, \sigma_{j2}, \sigma_{j3}, \ldots, \sigma_{jp}> \]

Each note contains information like pitch, volume, etc.

\[ N_n = <\text{PTCH, VOLU, LENG, ITVL, PART, MEAS, NVAL}>_{n} \]
Table 2. Attributes of a note

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>PTCH</td>
<td>Pitch</td>
</tr>
<tr>
<td>VOLU</td>
<td>Volume</td>
</tr>
<tr>
<td>LENG</td>
<td>Length</td>
</tr>
<tr>
<td>ITVL</td>
<td>Interval</td>
</tr>
<tr>
<td>PART</td>
<td>Part</td>
</tr>
<tr>
<td>MEAS</td>
<td>Measure</td>
</tr>
<tr>
<td>NVAL</td>
<td>Note value</td>
</tr>
</tbody>
</table>

Figure 4 depicts the flow chart of the analysis procedure of musical structures.

![Flowchart](image)

Figure 4. Flow chart of musical structure analysis
Get start measure m_1 and end measure m_2 in range R

\[ m_1 \leq m_2 + 1? \]

Y: motif u_1 = [m_1, m_2] → End

N: motif u_1 = [m_1, m_1 + 1]

Get start note n_1 and end note n_2 of u_1

n_2 is Auftakt?

Y: note n_3 = start of upbeat

N: voice part with melody changed in u_1?

Y: note n_3 = change point

N: tonal changed in u_1?

Y: note n_3 = change point

N: n_3 = 1?

Y: n_2 = end of m_1
   \[ u_1 = [m_{1-1}, m_1 + n_2] \]

N: n_2 = n_3 - 1
   \[ u_1 = [m_{1-1}, m_1 + 1 + n_2] \]

R = R - u_1;

m_1 = m_1 + 2;

R = R - u_1;

m_1 = m_1 + 1;

Find prev motif and next of u_1

Figure 5. Flow chart of motif analysis
3.2.2 Relationships among occurrences of musical structures

Relationships, especially the similarity, among occurrences of musical structures regulate performance expression. Similar occurrences are played similarly, though dynamics and agogics are changed concerning the role of each occurrence in the whole musical piece.

Actually, most of compositions are expressed through repeating and variance of melodies and rhythms. Relationships among occurrences of musical structures affect performance expressions [Bal96] [Hon92]. One of our automatic performances rendering system utilizes performance of an occurrence of structure (s1) to generate performance of other occurrences, which are regarded to be similar referring to s1 [Iya97].

For a musician, with just one look, he or she can judge whether two occurrences are similar or not. But for a computer, it would be much more difficult. For instance, in Figure 6, when comparing motif9 and motif11, it should be considered not only the starting position of the corresponding note (apparently, for note A in motif9, taking A’ or A” as its corresponding note in motif11 will obtain absolutely different results), but also the similarity relations between two motifs whose corresponding notes are totally different. (For example, pitch (motif9) = pitch (motif11) + 4).

![Figure 6. Comparison between motif9 and motif11](image)

The relationships among occurrences of musical structures are described by similarity level, which supposed to be a digit number ranging from 0 to 100. The smaller the similarity level, the more similar the two occurrences should be.

In DAPHNE, for units like motif, the index of “seed” (reference) unit and the similarity level are stored. Thus makes it possible for other performances to refer the performance of the “seed” during performance generation.

Figure 7 shows the analysis result of measure 9–16 in the example score of Chopin Mazurka Op. 7 No. 3 described in Chapter 3.2.1. The first column shows the motif number, the 4th column show the “seed” motif number, and the 5th column shows the similarity level. Since motif 9 is the
"seed" itself, its similarity level is 0. For motif11, it has the same melody as motif9, but its pitch is 4 degrees higher. That means its similarity level is 3. The "seed" motif of motif13 is also motif9. They have the same melody and motif13's similarity is 1.

```
<table>
<thead>
<tr>
<th>Unit NO</th>
<th>Start</th>
<th>End</th>
<th>RefUnit</th>
<th>SimLevel</th>
<th>SimLevel</th>
</tr>
</thead>
<tbody>
<tr>
<td>M10</td>
<td>9</td>
<td>10</td>
<td>9</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>M11</td>
<td>10</td>
<td>12</td>
<td>9</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>M15</td>
<td>13</td>
<td>14</td>
<td>9</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>M15</td>
<td>15</td>
<td>16</td>
<td>9</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>
```

Figure 7. Relationships among structures (Chopin Mazurka Op. 7 No. 3 m9-m24)

The algorithm of similarity level analysis is as follows:

- Function StructBody(σ)
  Get the main body of σ, which is only composed of full measures. The Auftakt is not included.
  \[ σ_1 = \text{Auftakt} ∗ \text{StructBody}(σ) \]
  \[ \text{StructBody}(σ) = \langle μ_1, μ_2, \ldots, μ_n \rangle \]
  When comparing structure σ_1 and σ_2, only the main bodies are considered.
  \[ \text{Compare}(σ_1, σ_2) = \text{Compare} (\text{StructBody}(σ_1), \text{StructBody}(σ_2)) \]

- Function NoteValue(τ)
  Get the value of note set τ. It equals to the summary of the values of all the notes that belong to τ.
  \[ \text{NoteValue}(τ) = N_{1_{NVAL}} + N_{2_{NVAL}} + \cdots + N_{n_{NVAL}} \]

- Function SamePart(i,j)
  Get the common part of τ_i and τ_j.
  \[ \text{SamePart}(i,j) = \langle N_{i_1}, N_{i_2}, \ldots, N_{i_k} \rangle \]
  Where
  \[ τ_i = \langle N_{i_1}, N_{i_2}, \ldots, N_{i_m} \rangle \]
\[ \tau_j = \langle N_{j1}, N_{j2}, \ldots, N_{jk} \rangle \]
\[ N_{jl} = N_{jl} \]
\
... \[ N_{lk} = N_{jk} \]
\[ H \subseteq \{ l_1, l_2, \ldots, l_k \} \]
\[ N_{lh} \neq N_{jh} \]

- **Definition** \( \tau_i \overset{\Delta}{=} \tau_j \)
  
  This means that \( \tau_i \) and \( \tau_j \) are similar. In DAPHNE, if 3/4 of two notes are the same, it is said that they are similar.

  \[ \tau_i \overset{\Delta}{=} \tau_j = \frac{\text{NoteValue}(\text{SamePart}(i, j))}{\text{NoteValue}(\tau_i)} \geq 3/4 \]

- **Definition** \( \text{Stacc}(i, j) \)

  It means that \( \tau_i \) is \( \tau_j \) with staccato.

- **Function** \( \text{SameShape}(i, j) \)

  It means the melodies of two notes set \( \tau_i \) and \( \tau_j \) are the same, however the difference on pitches and the existence of \( \text{Staccato} \) are permitted.

  \[ \text{SameShape}(\tau_i, \tau_j) = \begin{cases} 
  \tau_i & \neg(\tau_i \overset{\Delta}{=} \tau_j) \land \text{LENG}(\tau_i) > 1 \\
  & \land (\text{PTCH}(\tau_i) + m = \text{PTCH}(\tau_j)) \\
  & \lor \text{Stacc}(\tau_i, \tau_j)) \\
  \epsilon & \text{otherwise}
\end{cases} \]

- **Function** \( \text{SameShapePoint}(i, j) \)

  It can be used to obtain the similarity level of two occurrences.

  \[ \text{SameShapePoint}(i, j) = \begin{cases} 
  0 & \text{NoteValue}(\text{SameShape}(i, j)) \leq 1/4 \\
  & \land \text{NoteValue}(\text{DiffPart}(i, j)) > \text{NoteValue}(\text{SameShape}(i, j)) \\
  & \land \text{NoteValue}(\text{DiffPart}(i, j)) > \text{NoteValue}(\text{SameShape}(i, j)) \\
  1 & \text{otherwise}
\end{cases} \]

- **Function** \( \text{DiffPart}(i, j) \)

  This means that \( \tau_i \) and \( \tau_j \) are different. The conditions is
\[
\text{DiffPart} (\tau_i, \tau_j) = \begin{cases} 
\varepsilon & \text{if } \tau_i = \tau_j \\
\tau_i & \neg (\tau_i = \tau_j)
\end{cases}
\]

➤ Function \textit{DiffPoint} \((i, j)\)

The similarity level of two occurrences is obtained by calling \textit{DiffPoint} \((i, j)\). The smaller the return value, the more similar they are. Here's the equation.

\[
\text{DiffPoint} (i, j) = \text{Roundup} \left( \frac{\text{NoteValue} (\text{DiffPart} (i, j)) - \text{NoteValue} (\text{SameShape} (i, j)) \ast 4}{\text{NoteValue} (\sigma i)} \right)
\]

➤ Function \textit{RelaPoint} \((i, j)\)

It is the cardinal number of the similarity level.

\[
\text{RelaPoint} (i, j) = \begin{cases} 
0 & (i = j) \\
1 & (i \neq j)
\end{cases}
\]

➤ Function \textit{SimiLevel} \((i, j)\)

Stands for the similarity level.

\[
\text{SimiLevel} (i, j) = \text{RelaPoint} (i, j) + \text{DiffPoint} (i, j) + \text{SameShapePoint} (i, j)
\]

Where \(1 \leq i \leq j\)

➤ Definition \textit{Similar} \((i, j)\)

Indicates whether two occurrences are similar or not. In DAPHNE, if \textit{SimiLevel} \((i, j)\) is not more than 3, it means similar.

\[
\text{Similar} (i, j) \quad \Leftrightarrow \quad \text{SimiLevel} (i, j) \leq 3
\]

Figure 8. Depicts the flow chart of similarity level analysis.
Figure 8. Similarity level analysis
Figure 9. Comparison of motifs (to be continued)
\begin{itemize}
\item \( m_1 = \{n_{i1}\}, \quad i = 1,2,\ldots,n_1 \) (notes)
\item \( m_2 = \{n_{2i}\}, \quad i = 1,2,\ldots,n_2 \) (notes)
\end{itemize}

Compare \( n_{i1} \) with \( n_{2j} \)

\begin{itemize}
\item note \( n_{i1} = n_{2j} \)?
\item \( n_{i1} = n_{2i+k} \)?
\item \( n_{i1} = (\text{stacc})n_{2i} \)?
\item \( n_{i1} = n_{1a} + n_{1b} + n_{1c} + n_{1d} \)
\item \( n_{2i} = n_{2a} + n_{2b} + n_{2c} + n_{2d} \)
\end{itemize}

where
\begin{itemize}
\item \( n_{1a} = n_{2a} \)
\item \( n_{1b} = n_{2b} + k(\text{same shape}) \)
\item \( n_{1c} = (\text{stacc})n_{2c} \)
\item \( n_{1d} \neq n_{2d} \) (different)
\item simipoint = simipoint + 
  \( \text{len}(d)^4/\text{len}(N) + \text{len}(b)/\text{len}(N) \) 
  + \( \text{len}(c)/\text{len}(N) \)
\end{itemize}

Get next measure \( m_1, m_2 \)

\begin{itemize}
\item simipoint > 3 ?
\item \( m_2 \) is not similar to \( m_1 \)
\item \( m_2 \) is similar to \( m_1 \)
\end{itemize}

End

Figure 9. Comparison of motifs (continued)
3.2.3 Structure functions

Notes in an occurrence of a musical structure could have different meanings and play their own roles [Ber87]. As described in Table 3, initiative, anacrusis, and desinence are structural functions found in an occurrence of a musical structure [Iga00].

<table>
<thead>
<tr>
<th>Structure function</th>
<th>Expression</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anacrusis</td>
<td></td>
<td>A range where music is getting tenser prior to the initiative.</td>
</tr>
<tr>
<td>Initiative</td>
<td>o</td>
<td>The most tensional.</td>
</tr>
<tr>
<td>Desinence</td>
<td>^^^^^^^^</td>
<td>A range where tension is diminished following the initiative.</td>
</tr>
</tbody>
</table>

The information of structure functions is useful in automatic performance rendering. For structure functions, the factors that affect analysis are numerous and diverse. It’s difficult to summarize analysis rules and accomplish automatic performance. Even for different users, there would have different analysis results. Currently, users specify structure functions manually.

3.2.4 Harmony, chord progression, and tonality

Harmony, among the three elements (rhythm, harmony, and melody) of music, is an important element to form the backbone of music. Most of the music we usually listen is classic or jazz genre. They are based on the “Function Harmony” of the western tonal music. Therefore, without harmony analysis, the music analysis would be unacceptable [Shi87].

The analysis of harmony starts from perceiving chord, deriving tonality, then gives functions and cadence that is clue to performance expression at the end of a structure.

Various problems will emerge when doing harmony analysis. There are occasions that the automatic analysis fails because the “man-made” algorithms could not consider all the aspects. Users should specify the correct harmony for such cases. For that purpose, the Set Tonality and Set Chord sub-windows are designed.

In the Set Tonality sub-window (Figure 10), the user can specify the tonality that controls the whole piece. As previously stated, it will become easier to do function harmony analysis if this tonality is specified. In the dialog, the specified tonality can be previewed in the small window.
Figure 10. The *Set Tonality* sub-window

Figure 11. The *Set Chord* sub-window
In the *Set Chord* window (Figure 11), the user can specify the harmony. Global tonality is displayed on the top of the sub-window. The "local tonality change" can be set in the upper half. "Start" and "End" specify the starting and ending positions of the local tonality change.

The chord can be set in the lower half and confirmed through the preview window. Besides, the additional information like whether has 7 degrees, 9 degrees, root omission, and the chord inversion can also be specified.

By now, the harmony analysis is still under developing.

### 3.2.5 Musical forms

Usually, a music piece reflects the characteristics of a nation, a special period of times, the composer's intention and what he's seeking for. These characteristics are expressed through musical forms. Through the long history of music, some regular forms have been accumulated.

The analysis of musical forms may begin with small-scale or large-scale structures and proceed toward the opposite end. In many cases, standard forms are used, such as Binary, Ternary, Rondo, Sonata, or Theme and Variations. The titles of compositions often indicate what forms are being used. A familiarity with the standard forms is essential and a survey of them and their related genres can be found in any good text on the forms of music. A detailed analysis of form should contain some verbal description of the structure [Kik88].

The basic forms in tonal music are one-part (A), two-part (A-B, or A-A'), and three-part (A-B-A', A-B-C, or A-A'-A). Complex forms, like composite two-part, composite three-part, and more complex forms such as Rondo and Sonata can be understood as outgrowths of these three. In composite ternary, each section has its own internal form, a binary form or a ternary. The basic idea of ternary is statement-digression-restatement (A-B-A). The statement (A) is distinguished and identified by its difference from the digression (B). Therefore contrast is an essential attribute of ternary form. Restatement is dependent on a preceding digression, as well as on its identity with the initial A section. The division into three parts that marks ternary form is brought about entirely by design. The first division is normally due to a strong cadence marking the end of part one; the second is due to the contrasting nature of the material. It is said, the division into three parts is due to the contrasting material of the middle section. In binary form, on the other hand, the B material of the A-B form is similar rather than contrasting.

Formal analysis here means one that examines the overall structure. It includes an examination of the structure, rhythm, variation techniques, and especially the relationships between small and large-scale structures.

The motifs and phrase often form the basis of the small-scale building blocks, or cells, in a composition. Thus, this is a good place to begin. Large-scale divisions are usually clear and easy
to establish; so, they should be outlined near the beginning. Cadences, the "punctuation marks" of music, help to define them.

The last may combine to form even larger units. Thus, the large-scale structure is built from small cells, and the larger structures then echo the form of the small much in the way that a rectangular building resembles the brick from which it is made. This type of aggregated structure is called architectonic form. It is found abundantly in music around the world and resembles the self-replicating structures found in fractals and in nature.

To a greater or lesser extent, DAPHNE can analyze musical forms automatically based on musical structures, relationships among occurrences of musical structures, harmony progression, and tonality.

It is difficult to analyze musical forms only with structure, motif, phrase, sentence that obtained according to the Phrasing theory, because a music piece is constructed with many sentences and musical forms analysis is performed based on sentences and their similarity levels.

Thus higher-level units must be found so as to musical forms can be judged according to the relationships of them. Therefore, new units like two sentences, four sentences, etc. are extended upon sentence. Due to this method, DAPHNE can analyze musical forms automatically.

Figure 12 is the flow chart of musical form analysis.
Analysis of structure, and similarity level on motif, phrase, sentence, harmony, etc.

Number of sentences > 3?

Y

Analysis of structure, and similarity level on paragraph

Y

Binary part?

N

Three part?

Y

Processing
Composite three-part,
Sonata,
Fuga,
Rondo,
Variation.

N

Get range of each part

End

Figure 12. Flow chart of musical form analysis
Figure 13 shows the analysis scene of Chopin's Mazurka No. 7. It is a 3-part form and the brackets show the range of each one.

![Figure 13. Result of musical form analysis](image)

Due to the complexity of musical forms, some pieces still require users to specify forms explicitly.

Except structural functions, all of the five analysis items above are automatically performed to some extent. Analyzed results are saved into the DAPHNE file for other systems.
3.3 Visualization of performance data

Taking expressive performance generation as its ultimate purpose, DAPHNE is different from other simple music analysis systems. Besides analysis, DAPHNE provides the visualized performance analysis and accompaniment information analysis functions according to the results of music analysis. This makes music analysis and performance combined directly.

Visualized performance data based on information of analysis helps users to confirm analysis and to find performance characteristics. It enables users to understand performance objectively and to describe relationships between an expression and performance parameters. Users can select different performance parameters such as volume or velocity and show them on the graph. Using these graphs, as well as confirming analysis, users can analyze performance and make comparisons of different performances; even to find new performance rules.

DAPHNE pays attentions to the structure function of each segment of a piece, and the obtained musical structures and structure functions are applied to the performance analysis. Here, the musicians’ expressions can be obtained. First of all, the analysis is based on the same music theory as the musicians used, which means the analysis had the same musicology viewpoints. And, various performance elements can be displayed simultaneously, thus the relationships among these elements can be extracted [Uru99].

Figure 14 shows the example of displaying an oval graph. In the graph, a circuit stands for a unit like a motif or a measure (Figure 14 is motif9 of Mazurka. 7-3.). Two overlapped motifs with different length can be compared easily.

![Figure 14. Visualization of performance data (Mazurka 7-3 motif9)](image-url)
A structure function is displayed as a supplement line. The *anacrusis* part is drawn with solid lines, the *desinence* part with dashed lines and the oval, which stands for the sound of the *initiative*, with bold lines.

A diversified, overall analysis of the performance expression can be done by an oval graph based on musical structures. Moreover, the characteristics between similar structures can be obtained. In addition, the feature of one level performance becomes comprehensible by showing structure functions.

### 3.4 Information for accompaniment system

The performance for the accompaniment part is analyzed and synthesized referring to musician's rehearsal performance. An accompaniment part does not necessarily always follow musicians' performance. Since the beginning of a musical structure has more meanings, one of our accompaniment systems gets the information of musical structures and the fitting positions before a session starts [Iga93].

In example Sicilienne, the starts of phrases where a soloist should lead performance are specified as fitting points as Figure 15.

![Figure 15. Information for accompaniment system (Sicilienne)](image)
3.5 User interface

We expect DAPHNE to be used not only by computer scientists interested in music but also by music professionals. In order to make the system sophisticated in the sense of both music and computer system, DAPHNE's user interface should be user friendly especially for music professionals. Musical analysis with scores and performances is available in DAPHNE as if users analyze with actual scores (music sheets), even with those of different editions, and performance (by CDs or their own play) on their desks or pianos in the real world. Not only analyzed information is prompted on the score image, but also performance is visualized using the information. Since expected users do not necessarily get used to computer systems, it attached importance to DAPHNE's user interface; let music professionals who are not familiar with computers feel few obstacles on learning and using DAPHNE [Liu99-a].

User can analyze music items by any or a mixture of the following three methods.

3.5.1 Command line input from main window

![Figure 16. The main control window of DAPHNE](image)

The command is input according to the synopses shown as follows. Since DAPHNE judges the input only by the keywords, the order of the input words in a sentence does not matter. Even if the
user does not input anything, the system can partially supplements the available inputs.

3.5.2 Input from the sub_window of each analysis

Because DAPHNE offers sub_windows which corresponds to the command for each analysis item, the user can analyze by selecting some necessary values on each sub_window.

![Figure 17. The sub-window of structure analysis](image)

3.5.3 Analysis on musical scores

Score sheets are required for musical analysis in the real world. Thus DAPHNE provides users the score images that can be manipulated through a popup menu (Figure 18). Therefore, users can analyze and get results on actual scores directly.

For example, in spite of the frequent appearance, it is not simple to represent the position of the second note of a triplet with the information of a measure’s number and the offset from the beginning of the measure. Not only for specifying but also for confirmation, using score images is the user friendly way for music analysis. DAPHNE has a set of line commands that correspond to the WYSIWYG analysis. Figure 19 depicts the analysis window of Mazurka 7-3.
Figure 18. The popup menu on the score images

Figure 19. The analysis window of Mazurka 7-3
3.6 Other design features of DAPHNE

3.6.1 Performance replay
During an analysis session, users can listen to the performance (by different players as far as there are performance data in MIDI) of a piece. Using a tool of PSYCHE project for playing MIDI data, users can listen to the performance with a specified range.

3.6.2 Visualization

➤ Visualization of analysis - DAPHNE provides users the score images that can be manipulated. Therefore, users can analyze and get results on actual scores directly. Moreover, in order to display multiple sheets of scores, users are able to use two or more monitors to display scores at the same time.

➤ Visualization of performance data - Using the analysis information, performance is visualized for confirming analysis or, conversely, visualization helps analysis. Visualized figures are not only simple measure graph type but also new types of figures as described in [Uru99].

3.6.3 Variety of analysis
Moreover, the database is expanded to keep variety and the accuracy of the analysis. Since there are many editions of score sheets for a musical piece, these editions are displayed simultaneously for comparing diversification and for confirming the correct analysis. Moreover, the differences of different versions’ analyses can be maintained [Liu00-b].

Figure 20 depicts both Zenon and Paderewski versions of Maz. No. 7, along with their own analysis results. The ranges of motif 9 are different since in the Zenon version, there is one more accent than in the Paderewski version. The window below shows the UNI data with analysis results.

Giving prominence to the difference of different versions’ results, the user can collate the scores.
Figure 20. Compare the analysis results of Zenon and Paderewski