

ステータス:	終了	開始日:	2010/04/11
優先度:	通常	作業時間の記録:	0.00時間
担当者:			
カテゴリ:			
対象バージョン:			
説明			
譜面モニタの属性表示機構を拡張。連結 & 記号のスラー記述拡張			
関連するチケット:			
関連している Bug # 38: (V5.70)ショートカットでメニュー選択するとメンバーがOFFになってしまう終了			
		2010/04/13	2010/04/13

履歴

#1 - 2014/01/07 20:34 - Redmine Admin

；《Ver5.7 開発後記》2010.04.09

；

； 譜面モニタに関する機能強化を2つと内部処理の改善を2つ、そして文法解釈の

； 変更を1つ行いました。譜面モニタ強化の1つ目は前回予告したテンポ表示の拡

； 張です。数ある属性の中から、譜面モニタ表示でMusingが効率的になるもの、す

； なわち曲中指定の可能性が高いものを厳選し、それをポップアップで選択できる

； ようにしてあります。前回の開発を通して表示スタイルやフィンガー情報ダイア

； ログとの連動等、その完成イメージの骨格は固まっていたのですが、開発は難航

； を極めました。全体属性であるテンポ、メンバー属性である音量、フィンガー属

； 性である強弱指定など、それぞれが値の範囲や±記号に対する記法定義に微妙な

； 差異を持ちます。それらに対応し統一感のある機能としてご提供するには、内部

； データの持ち方やフラグの意味をも改修していく必要がありました。改造はMIDI

； エクスポートまで波及しプログラムを大幅に見直す事になりました。

； そんな苦難の改良でしたが様々な工夫を盛り込みました。今回は単純に2行間を

； 交互に表示するロジックでしたが、テンポ以外の属性表示をしてみるとその仕様

； では重なって判読できなくなる状態が頻発。しかし機械的なシフトで闇雲に行数

； を増やすのは私の美感が許しません。そこで各文字列長を算出し余裕のある場合

； は1行目を優先しつつ、混み合う際には複数の行に分散していくといったインテ

； リジェンスな処理を導入しました。その際、行数限界の妥当性検証に皆さんの優

； れたMuseデータが大いに役に立ちました。充実したMuseデータが蓄積されている

； ということは、そこから得られる音楽の楽しさだけでなく、開発自体に直接貢献

； しています。皆さん本当にありがとうございます。

； 前回のテンポ表示と同様、同じ時刻の属性を1つにまとめて表示する様にしまし

； たが、この集約処理は遅延系属性に対してのみ効果的で遅延の無い属性はむしろ

； 可読性を損なう事に気づきました。そこでグルーヴ属性やペダル属性、そしてア

； クセントは敢えて同時刻でも複数行に展開する様にしています。

； 遅延指定がある場合、音部記号エリアの数値は基本的にその最終値を表していま

； すが、強弱属性は各ノートが担う情報であることから例外的な扱いにしています。

； また強弱は(v)だけでなくアクセント(w)も同格に扱う工夫も施しています。一方

； グルーヴ属性(pq)は表示すべき値が音長となるため、レイアウトの都合上、音部

； エリアの表示は断念しました。ご容赦下さい。

； 操作機構上の工夫も施しました。表示対象となるメンバーやフィンガーは、フィ

； ンガー情報ダイアログの選択行と連動させるだけでなく、譜面モニタでメンバー

； IDのキーを入力すると切替えられる様にしてあります。サクサクと切替えられる

； ので重宝すると思います。

； さて、2つ目の譜面モニタに対する強化は、WAVEコマンドも表示するようにした

； 事です。WAVEコマンドにはレイテンシー(もたつき)が付き物ですが、そのタイミ

； ング調整が格段にし易くなったと思います。そして内部処理の改善は、実はこの

； WAVEコマンド表示の開発過程から派生しました。この開発の最中、WAVEレイテン

； シーが以前のバージョンと比べて悪化していることに気づいたのです。よくよく

； 調べてみると、なんと前々回の改版で実施した同時刻発音ノートの順序最適化処

； 理に絡んでいました。実験の結果、WAVEコマンドは同時刻内の最後尾に配置する

； とレイテンシーが緩和されることが判明。多分、その次の音源コマンドとのイン

； ターバル確保がしやすくなるためだと分析しています。早速、最適な順序処理を

； する様に改善しました。ちなみに今回から、添付のオーディオファイル(拍手)を

； mp3化してみました。

； もう一つの内部処理の改善は、曲尾の判定を占有音長だけでなく実際の発音音長

； も加味し、より長い方を採用するようにしたことです。実はこの改善も積極的な

； ものではなく、後述する文法解釈の変更に伴い必要に迫られて実施しました。以

； 前、掲示板の書き込みでhimajin925さんから、曲の最後に和音を記述した際の曲

； 尾処理に関し疑問を投げかけられたことがあったのですが、今回の改善によって、

； 違和感の無い自然な仕様になったと思います。

； さて肝心の文法解釈変更の内容ですが『スタッカートなどで音長が短くなってい

； る音符に連結&を添えると、たとえ次に来る音符の音程が同一でない場合であっ

； ても、規定音長まで伸ばす』というものです。同一音程で接続される場合がタイ

； とすれば、今回の仕様によってスラーも追加されたと言えます。実は2年半前の

； V5.2において連結&の文法解釈を変更した際に一気にやってしまいたかったので

； すが、プログラムがあまりに複雑に絡み合い、どうしてもまともな動作するコン

； パイルエンジンに仕立て上げる事ができませんでした。今回譜面モニタ強化に伴

； い大域的に構造を見直す機会があったので再度チャレンジし、無事完成に漕ぎ着

； けることが出来ました。無論、データの互換性が少々崩れますが、今回の解釈変

； 更に関してはむしろ意図的にスラーとして記述していた過去データの方が多いだ

； ろうと楽観しています。また、それを補って余りある効能があると判断しました。

； 例えば、スタッカートで演奏している一連の旋律の中で、一部だけスラーにした

； 場合、今までは音長を明に記述して音を規定音長として指定し、更にその次の

； 音符で再びスタッカート指定をしなければなりませんでした。これからはスラー

； にしたい音符に&を付けるだけで済みます。あたかもアクセントコマンドの様に、

； 一時的指定として機能します。コーディングの可読性も格段に増すと思われます。

； 今回の開発後記、少々長くなってしまいました。しかし実際いっぱい開発したの

； ですから仕方が無いですよ(苦笑)。苦勞して開発したこれらの機能、ぜひ有効

； に活用して頂きたいと切に願っています。