

REPERTUAR MUZYKI  
POLSKIEJ WIEKU  
XIX – WYDANIA  
CYFROWE I ANALIZA  
SKOMPUTERYZOWANA

Powstanie artykułu możliwe było dzięki uczestnictwu autora w programie „Mobilność Plus III Edycja” finansowanym przez Ministerstwo Nauki i Szkolnictwa Wyższego. Tekst został przyjęty do wydania w trybie *double-blind review* process.

## 1. Wprowadzenie

Niemal dwadzieścia lat temu Iwona Lindstedt pisała na temat metod informatycznych w analizie muzycznej: „Już pierwsze próby wykazały, iż techniki komputerowe nie tylko doskonale służą liczeniu i tworzeniu tabel cech statystycznych muzyki, dzięki czemu można wnioskować o cechach stylistycznych dzieła, lecz także sprawdzają się w badaniach porównawczych poszczególnych elementów muzycznych, co prowadzić może do zdefiniowania składni muzycznej rządzącej materiałem danego dzieła. Szybko doceniona została też stymulująca rola metod informatycznych w analizie muzycznej, która pobudza do zadawania nowych pytań badawczych, sugeruje nowe podejścia i nowe problemy, nieodczytywalne z oryginalnego, nieprzetworzonego maszynowo materiału”<sup>1</sup>

Obecnie do listy zalet zastosowania technik komputerowych na gruncie muzykologii należałoby dodać także: (1) możliwość prowadzenia badań atrybucji dzieł anonimowych dzięki wykorzystaniu coraz bardziej rozrastających się baz danych zawierających zapisy nutowe w jednym z wielu dostępnych formatów symbolicznych (np. MusicXml<sup>2</sup>, MEI<sup>3</sup>, Humdrum<sup>4</sup> etc.), (2) tworzenie wizualizacji wyników analizy w formie modeli<sup>5</sup>, (3) renderowanie dynamicznych partytur na stronach internetowych na podstawie zapisu symbolicznego. Niestety, pomimo tak obiecujących perspektyw na gruncie muzykologii polskiej w przeciągu ostatnich dwudziestu lat trudno było zaobserwować poważniejsze zainteresowanie możliwościami, jakie daje analiza skomputeryzowana<sup>6</sup>. Wydaje się, że głównym powodem takiego stanu rzeczy jest dość duża hermetyczność zarówno muzykologii jako dziedziny naukowej, jak i informatyki (lub – dokładniej – wiedzy z zakresu programowania), co utrudnia współpracę programistów z muzykologami (na problem ten zwracała uwagę Lindstedt). Tymczasem współpraca taka jest warunkiem koniecznym dla wytworzenia narzędzi pozwalających osobie niemającej głębszej wiedzy w zakresie systemów komputerowych oraz języków programowania przeprowadzać efektywną analizę skomputeryzowaną muzyki. Zauważmy, że w przypadku dwu najważniejszych (a z pewnością oferujących najwięcej możliwości analitycznych) systemów skomputeryzowanej analizy muzycznej, jakimi są *Music21*<sup>7</sup> oraz *Humdrum Toolkit* wraz z *Humdrum Extras*, brak interfejsów

1 Iwona Lindstedt, *Metody informatyczne – nowe perspektywy analizy muzycznej?* „Muzyka”, 4 (179), 2000, s. 63.

2 Oficjalna strona formatu: [www.musicxml.com](http://www.musicxml.com) [dostęp: 1 V 2017].

3 *Music Encoding Initiative*, zob. [music-encoding.org](http://music-encoding.org) [dostęp: 1 V 2017].

4 Format rozwijany aktualnie w Center for Computer Assisted Research in the Humanities na Uniwersytecie Stanforda, zob. [ccarh.org](http://ccarh.org) [dostęp: 1 V 2017].

5 Np. model struktury harmonicznego utworu stworzony za pomocą programu *mkeyscape*, będącego jednym z wielu programów ułatwiających analizę muzyczną wchodzących w skład zestawu narzędzi *Humdrum Extras* autorstwa dra Craiga Sappa, zob. [extras.humdrum.org](http://extras.humdrum.org) [dostęp: 1 V 2017].

6 Oczywiście od takiego stwierdzenia są pewne wyjątki, wśród których na czoło wysuwają się prace Ewy Dahlg-Turek i baza pieśni ludowych zakodowanych w formacie EsAC (zob. [www.esac-data.org](http://www.esac-data.org) [dostęp: 1 V 2017]) oraz badania Iwony Lindstedt.

7 Zob. [web.mit.edu/music21](http://web.mit.edu/music21) [dostęp: 1 V 2017].

8  
Dla wyróżnienia składni poleceń terminala używamy – zgodnie z konwencją przyjętą w publikacjach informatycznych – czcionki o stałej szerokości; ukośnik na końcu wersu „\” oznacza kontynuację zapisu w jednej linii (ze względów edytorskich w wydaniu papierowym takie linie są łamane).

9  
Na potrzeby niniejszego artykułu posługujemy się pojęciem *Humdrum* w co najmniej dwu znaczeniach: (a) syntaksa *Humdrum* oraz (b) zestaw programów do analizy zapisów symbolicznych muzyki *Humdrum Toolkit* oraz *Humdrum Extras*. Ta wieloznaczność pojęcia bywa niejednokrotnie powodem pewnego zamieszania. Np. w pracy Macieja Gołąba czytamy: „[...] Tę ostatnią tendencję widać wyraźnie w uniwersalnym programie informatycznym do wspomaganie badań muzykologicznych HUMDRUM (w skład którego wchodzi podprogramy *Humdrum Syntax* i *Humdrum Toolkit*)” (Maciej Gołąb, *Spór o granice poznania dzieła muzycznego*. Wydawnictwo Naukowe Uniwersytetu Mikołaja Kopernika, Toruń 2012, s. 153). Tymczasem *Humdrum Syntax* oznacza zespół reguł syntaktycznych formatu *Humdrum*, nie zaś oprogramowanie.

10  
Zob. [musescore.org/pl](http://musescore.org/pl) [dostęp: 1 V 2017].

11  
Zob. [lilypond.org](http://lilypond.org) [dostęp: 1 V 2017].

12  
Zob. [science.jkilian.de/salieri/GUIDO/index.html](http://science.jkilian.de/salieri/GUIDO/index.html) [dostęp: 1 V 2017].

13  
Zob. [abcnotation.com](http://abcnotation.com) [dostęp: 1 V 2017].

graficznych umożliwiających wykonywanie operacji bez znajomości składni jednego z języków programowania oraz umiejętności pracy w środowisku tekstowym (wpisywanie poleceń w tzw. terminalu).

Posłużmy się przykładem – aby na gruncie programu *Music21* stworzyć prosty model w postaci tzw. rolki pianolowej najwyższego głosu jednego z chorałów Jana Sebastiana Bacha, należy wpisać następujący kod<sup>8</sup>:

```
from music21 import *
sBach = corpus.parse('bach/bwv57.8')
sBach.getElementById('Soprano').plot \
('PlotHorizontalBarPitchSpaceOffset',
title='Soprano')
```

Analogiczny do powyższego graf uzyskamy na gruncie *Humdrum*<sup>9</sup> za pomocą kodu:

```
extractx -s 4 bach.krn - proll > bach.ppm
```

Wpisanie tego polecenia tworzy graf w formacie .ppm (format zapisu grafiki rastrowej), który można przekonwertować do jednego z popularnych formatów graficznych (.jpg lub .tif).

Przykład 1.

Jan Sebastian Bach, *Christ lag in Todesbanden*, BWV 277, graf programu proll.



W obydwu przypadkach wygenerowanie końcowego modelu w postaci pliku graficznego wymaga nie tylko znajomości samego oprogramowania, języka *Python* dla *Music21* lub powłoki *bash* dla *Humdrum*, lecz także umiejętności pracy w środowisku tekstowym.

Także w przypadku edytorów przeznaczonych do przygotowywania składów nutowych zdecydowanie największą popularnością cieszą się programy graficzne – są to na gruncie polskim przede wszystkim komercyjne produkty, jak *Finale* oraz *Sibelius*. Zdecydowanie rzadziej wykorzystywany jest darmowy, a jednocześnie bardzo zaawansowany i generujący wysokiej jakości skład program *MuseScore*<sup>10</sup>. Niemal zupełnie nieznaną są programy typu *Open Source* mogące wyrenderować co prawda bardzo wysokiej jakości skład nutowy, wymagające jednak umiejętności pisania kodu źródłowego. Wśród nich warto wspomnieć o *LilyPond*<sup>11</sup>, *Guido*<sup>12</sup> czy *ABCplus*<sup>13</sup>.

W kontekście powyższych obserwacji nie sposób powstrzymać się od stwierdzenia pewnego paradoksu – otóż w pracy badawczej

o wiele większe możliwości wszechstronnego wykorzystania zapisanego za pomocą komputera materiału nutowego dają rzadziej używane programy darmowe, oparte na otwartym kodzie źródłowym i posługujące się otwartymi tekstowymi formatami zapisu, niż odpowiadające im zamknięte, komercyjne produkty. Podstawową przewagą formatów otwartych jest bezpośredni dostęp do zapisanego materiału. Duża część oprogramowania *Open Source* wykorzystuje bardzo intuicyjne i zrozumiałe dla osób czytających nuty formaty zapisu. Porównajmy zapis pierwszych czterech taktów głosu wokalnego pieśni *Piast* autorstwa Franciszka Lessla ze zbioru *Śpiewów historycznych* Niemcewicza<sup>14</sup>.

Przykład 2:

Franciszek Lessel, *Piast* [głos wokalny], t. 1–4<sup>15</sup>.



Zapis w formacie .abc (*ABCplus*):

```
L: 1/8
M: 2/4
K: Ab clef=treble
EA GB |
Ac2B/A/ |
GA Bc |
B>G E z~ |
```

Zapis w formacie .ly (*LilyPond*)<sup>16</sup>:

```
\clef "treble" \key as \major \time 2/4
es8 [ as8 ] g8 [ bes8 ]
as8 c4 bes16 [ as16 ]
g8 [ as8 ] bes8 [ c8 ]
bes8. [ g16 ] es8 r8
```

Zapis w formacie .krn (*Humdrum*)<sup>17</sup>:

```
**kern \ *clefG2 \ *k[b-e-a-d-] \ *M2/4
=1- \ 8e-L \ 8a-J \ 8gL \ 8b-J
=2 \ 8a- \ 4cc \ 16b-L \ 16a-J
=3 \ 8gL \ 8a-J \ 8b-L \ 8ccJ
=4 \ 8.b-L \ 16gJk \ 8e- \ 8r
= \ *-
```

Otwartość kodu, w jakim zapisano powyższe przykłady, oznacza nie tylko, że w stosunkowo krótkim czasie, opanowawszy składnię,

14  
Julian Ursyn Niemcewicz,  
*Śpiewy historyczne  
z muzyką i rycinami*,  
Warszawa, w drukarni  
J.C.K. Mości Rządowéy  
1819, 26.

15  
O ile nie zaznaczono inaczej, wszystkie przykłady nutowe w niniejszym artykule zostały wyrenderowane za pomocą programu *Verovio*.

16  
W przypadku kodu .ly ukośniki (\) są jego integralną częścią i w przedstawionym przykładzie nie oznaczają łamania wiersza.

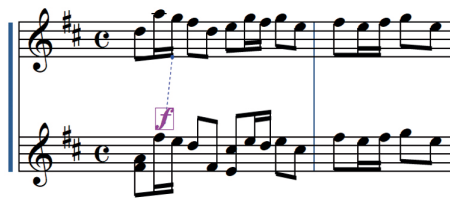
17  
Dla ułatwienia skrócono zapis, stosując znak ukośnika (\) są jego integralną częścią i w przedstawionym przykładzie łamanie wiersza. W składni *Humdrum* każde kolejno następujące „wydarzenie” zapisujemy w nowej linii – w istocie zapis powyższy to kolumna tekstu niedająca się efektywnie zapisać w druku.

można uzyskać pełną kontrolę nad zanotowaną treścią muzyczną, ale także czyni taki zapis całkowicie „przenośnym” – zmian można dokonywać w dowolnym programie do edycji tekstu (np. w popularnym *Wordzie* z pakietu *Microsoft Office*, *Excelu* czy zwykłym notatniku tekstowym). Co więcej, manipulowanie treścią muzyczną w celu wykonania określonych zabiegów analitycznych sprowadza się do manipulacji tekstem. Jest to widoczne zwłaszcza w przypadku składu *Humdrum*, gdzie zapis nutowy w istocie przypomina arkusz kalkulacyjny. Do wykonania podstawowej analizy statystycznej pliku .krm nie jest konieczne wykorzystanie specjalistycznego oprogramowania – np. statystykę wystąpień poszczególnych dźwięków można sporządzić za pomocą prostego wyszukania tekstu (komenda `ctrl+f` „e” w programie *Excel* odnajdzie wszystkie dźwięki „e”, standardowe narzędzia statystyczne programu pozwolą zaś policzyć wszystkie jego wystąpienia).

Jednak brak dostępu do kodu źródłowego ma znacznie poważniejsze negatywne konsekwencje. Analogiczny do powyższych zapis nutowy sporządzony w programach komercyjnych można otworzyć (z założenia) wyłącznie w programie, w którym sporządzono transkrypcję. *Finale* oraz *Sibelius* to programy typu „WYSIWYG”<sup>18</sup>. Oznacza to, że końcowy efekt pracy (skład nutowy) będzie wyglądał dokładnie tak, jak na ekranie monitora. W jakim znaczeniu ta pozorna korzyść stanowi problem? Otóż typowym zabiegiem dokonywanym przez autorów składów nutowych jest ich „upiększanie” poprzez ręczne przesuwanie elementów graficznych. Dla większości takich elementów, jak oznaczenia dynamiczne, artykulacyjne, ozdobniki etc., stosowane jest zakotwiczenie znaku do określonej nuty. Jednak ręczne przesunięcie oznaczenia niezwykle często skutkuje jedynie graficznym przemieszczeniem znaku przy jednoczesnym pozostawieniu niewłaściwego zakotwiczenia. Zjawisko to ilustruje dobrze przykład zapisu nutowego w programie *Finale* (zob. przykł. 3).

Przykład 3.

Zakotwiczenie oznaczenia dynamicznego w programie *Finale*.



Program za pomocą niebieskiej linii przerywanej wyświetla informację o zakotwiczeniu oznaczenia (czego nie widać na późniejszym wydruku), które z muzycznego punktu widzenia (jeśli sugerować się jedynie warstwą graficzną) przynależy raczej do drugiej szesnastki, i to w innym głosie. Tego typu niepożądaną niejednoznaczności

18  
Ang. *What you see is what you get.*

zapisu łatwo uniknąć w większości formatów otwartych, a w sporej części z nich (m.in. *Humdrum*) jest ona wręcz niemożliwa z uwagi na rygorystyczną syntaksę<sup>19</sup>. Dodatkowe problemy wynikające z wykonania transkrypcji w formacie zamkniętym wiążą się ze specyficznymi cechami notacji muzycznej. Zastosowanie nietypowych czcionek, ręczne przesuwanie systemów na stronie, ukrywanie elementów w celu osiągnięcia pożądanego efektu graficznego – te wszystkie zabiegi powodują, że tak przygotowany skład w zasadzie nie może być wykorzystany w innym celu niż do sporządzenia wydruku. W aspekcie funkcjonalnym zatem plik tego typu jest niemal równoważny transkrypcji wykonanej ręcznie na papierze nutowym.

W kontekście powyższych rozważań wydaje się oczywiste, że w przypadku podjęcia próby badań repertuaru muzyki polskiej wieku XIX z wykorzystaniem narzędzi informatycznych niezwykle ważny jest wybór odpowiedniej metody zapisu transkrypcji, która pozwoli na dalszym etapie prac wykorzystać je możliwie wszechstronnie. Artykuł niniejszy stanowi próbę refleksji nad dostępnymi formatami zapisu, metodologiami analizy skomputeryzowanej oraz możliwymi sposobami renderowania partytur repertuaru muzyki polskiej XIX stulecia. W ostatniej części artykułu zaprezentowane zostaną przykładowe transkrypcje oraz podstawowe możliwości analizy skomputeryzowanej.

## 2. Formaty zapisu symbolicznego

Pierwsze próby wykorzystania komputerów do zapisu oraz analizy muzyki podjęto już pod koniec lat 40. XX wieku<sup>20</sup>. W latach 60. wzmożło się zainteresowanie metodami informatycznymi oraz powstały pierwsze prace rozwijające teorie analizy statystycznej<sup>21</sup>. Wypracowano także pierwsze narzędzia umożliwiające wykonywanie analizy muzycznej wspomaganej komputerowo – służyło temu m.in. oprogramowanie autorstwa Johna Sellecka i Rogera Bakemana<sup>22</sup>. Stworzenie dającego się automatycznie przeszukiwać oraz analizować korpusu dzieł wymagało zastosowania specyficznego, standaryzowanego formatu zapisu. Zadanie to okazało się niezwykle trudne z uwagi na wieloaspektowość oraz częstą niejednoznaczność, a także wyjątkowo wysoki stopień skomplikowania notacji muzycznej. Właśnie dlatego pierwsze próby zbudowania baz zakodowanych zapisów muzycznych obejmowały stosunkowo łatwiejszą do zanotowania muzykę monofoniczną, a więc m.in. pieśni ludowe oraz chorał gregoriański. Niektóre z opracowanych w tym celu formatów okazały się niezwykle żywotne – wśród nich na szczególną uwagę zasługują *EsAC* (*Essener Assoziativ Code* oraz *P&E* (*Plaine and Easie Code*).

### 2.1. Wybrane systemy zapisu symbolicznego

Głównym celem twórców formatu *EsAC* było zakodowanie korpusu europejskich pieśni ludowych w możliwie przejrzystej i przenośnej

19 Także w darmowym, graficznym edytorze *MuseScore* problem nie występuje, jeśli bowiem przesuniemy ręcznie dane oznaczenie, jego zakotwiczenie jest aktualizowane i przenoszone do najbliższej nuty.

20 Zob. CCARH, *Music 253* [http://wiki.ccarh.org/wiki/Music\\_253](http://wiki.ccarh.org/wiki/Music_253) [dostęp: 15 V 2017]. Jednym z pierwszych badaczy korzystających z możliwości analizy komputerowej był Bernard Bronson (l. Lindstedt, *Metody informatyczne – nowe perspektywy analizy muzycznej?*, op. cit. s. 63). Warto podkreślić, że maszyny konkurujące o miano pierwszych komputerów zostały zbudowane ledwie kilka lat wcześniej (*ENIAC* w latach 1943–45 oraz *Colossus* w 1943 roku).

21 Zob. Wilhelm Fucks, *Mathematische Analyse von Formalstrukturen von Werken der Musik*, Köln 1963.

22 John Selleck, i Roger Bakeman, *Procedures for the analysis of form: Two computer applications*, „*Journal of Music Theory*”, 9 (2), 1965, s. 281–293.

23  
Zob. *Numbered musical notation*, [https://en.wikipedia.org/wiki/Numbered\\_musical\\_notation#History\\_and\\_usage](https://en.wikipedia.org/wiki/Numbered_musical_notation#History_and_usage) [dostęp: 17 V 2017].

24  
Zob. Wolfram Steinbeck, *The Use of the Computer in the Analysis of German Folksongs*, „Computers and the Humanities”, 5 (10), s. 1976, s. 287–296.

25  
Zob. Helmut Schaffrath, *The „Essen Associative Code”: A Code for Folksong Analysis*, w: *Beyond MIDI. The Handbook of Musical Codes*, red. Eleanor Selfridge-Field, MIT Press: Cambridge, MA, 1997, s. 343–361.

26  
Zob. <http://www.esac-data.org/> [dostęp: 17 V 2017].

27  
Dane ze strony *EsAC Data Homepage*, [www.esac-data.org/](http://www.esac-data.org/) [dostęp: 19 V 2017].

28  
Całość zakodowanego do tej pory repertuaru można pobrać ze strony Instytutu Sztuki PAN, <http://www.ispan.pl/pl/dzialalnosc-badawcza/kolberg-online> [dostęp: 19 V 2017].

29  
Oskar Kolberg, *Krakowskie: Dzieła wszystkie*, PWM, Ludowa Spółdzielnia Wydawnicza: Kraków i in. 1962, s. 200.

30  
Informacje dotyczące zasad kodowania oraz znaczenia pól metadanych można odnaleźć w przygotowanej przez Ewę Dahlig-Turek instrukcji. Zob. <http://www.ispan.pl/esac-instrukcja.pdf> [dostęp: 20 V 2017].

formie, pozwalającej na późniejsze wykorzystanie danych. Kod *EsAC* inspirowany jest chińską notacją *Jianpu*<sup>23</sup>. Projekt został zapoczątkowany w Niemczech w latach 70. XX wieku w *Deutsches Volksliedarchiv* na Uniwersytecie Alberta Ludwika we Fryburgu Bryzgowijskim. Autorem pionierskich prac opartych na analizie zakodowanego materiału byli Wolfram Steinbeck<sup>24</sup> oraz wieloletni kierownik projektu Helmut Schaffrath<sup>25</sup>. Od 1994 roku, kiedy zmarł Schaffrath, projekt kontynuowany jest przez Ewę Dahlig-Turek w Instytucie Sztuki Polskiej Akademii Nauk<sup>26</sup>. Aktualnie baza obejmuje repertuar ok. 20 tysięcy pieśni<sup>27</sup> (głównie niemieckich, polskich, chińskich oraz mniejszych kolekcji innych krajów europejskich), z czego zakodowano także 2844 pieśni z *Dzieł* Oskara Kolberga<sup>28</sup>.

Typowy zapis w formacie *EsAC* składa się z licznych pól metadanych zawierających podstawowe informacje o utworze i jego źródłach, następnie pola *KEY* zawierającego dane dotyczące sposobu kodowania (m.in. sygnaturę, podstawową wartość rytmiczną, tonikę oraz metrum) oraz *MEL* z zapisaną melodią.

Przykład 4.

*Slicna panienka*<sup>29</sup>.



Zapis pieśni w formacie *EsAC*<sup>30</sup>:

```
DWOK05
CUT[Slicna panienka]
REG[Krakowskie]
TRD[DWOK05 Krakowskie, s. 200]
KEY[K0003 16 G 3/8]
MEL[5_3_1_ 5__5_ 4_2_2_ 4__4_ 3_1_2_ 355__
4_3_2_ 321_0_ //]
```

Usunięcie wszystkich spacji, podkreślników oraz kropek z pola *MEL* pozwala uzyskać zapis samej melodii wyabstrahowanej z rytmu, dzięki czemu możemy przeszukiwać bazę pod kątem motywów melodycznych. Ponieważ zapis jest relatywny w odniesieniu do toniki zanotowanej w polu *KEY*, przeszukiwanie wzorców melodycznych możliwe jest niezależnie od pierwotnej tonacji utworu.

Niestety, poza wieloma zaletami system ma poważne ograniczenia, z których podstawowym jest niemożliwość zakodowania muzyki innej niż jednogłosowa. Ponadto w formacie nie da się zapisać tekstu pieśni, co zawęża potencjał analityczny do warstwy czysto muzycznej. Pomimo wskazanych problemów *EsAC* stanowi jeden z ważniejszych standardów, baza pieśni ludowych zaś to aktualnie



jedno z głównych źródeł otwierających drogę badaniom tego typu repertuaru. W połączeniu z możliwościami operowania na plikach tekstowych, jakie daje powłoka *bash* (*Bourne Again Shell*), baza plików zakodowanych w formacie *EsAC* pozwala na wykonywanie bardzo zaawansowanych działań analitycznych oraz wizualizację wyników.

Standardem powszechnie przyjętym w procesie katalogowania źródeł muzycznych jest format *Plaine and Easie Code* (*P&E*) wykorzystywany w bazie RISM. Z uwagi na potrzeby największego katalogu źródeł muzycznych, jakim jest RISM<sup>31</sup>, możliwości kodu *P&E* są znacznie większe niż w przypadku *EsAC*. Co ważne, format pozwala na tworzenie niepoprawnych z punktu widzenia ortografii muzycznej zapisów – należy pamiętać, że istotą zanotowanych w RISM incypitów muzycznych ma być dokładne odwzorowanie źródeł, które niejednokrotnie zawierają błędy. Rekordy RISM oprócz incypitów zapisanych w formacie *P&E* wyposażone są w bogate metadane, pozwalające identyfikować źródła muzyczne. Przykładowy incypit muzyczny dla utworu Józefa Elsnera wygląda jak na poniższym przykładzie<sup>32</sup>.

Przykład 5.

Józef Elsner, *Muzyka na wprowadzenie zwłok księcia Józefa Poniatowskiego*.



Zapis w formacie *P&E*<sup>33</sup>:

```
@clef:G-2
@keysig:xFCGD
@timesig:3/8
@data:''2E{8.G3EG}{8.B3GB}/''4.E{6DC}
{'8BB''CD}/4E''\
{8.B6B}{8.G6G}{8.E6E}/4B{8.B6B}{8B''6C''B}
{AGAF}/4E
```

Pomimo tego, że składnia *P&E* pozwala na zakodowanie większej ilości informacji niż *EsAC*, to jednak baza melodii ludowych ma istotną przewagę. Nie sposób bez żadnych zastrzeżeń porównywać dwu baz, z których w jednej zapisano ponad milion rekordów, w drugiej zaś około 20 tysięcy. Należy jednak pamiętać, że w formacie *EsAC* notowane są całe utwory, podczas gdy w RISM zapisuje się jedynie incypity wybranych głosów. Oznacza to, że utworów fortepianowych możemy szukać, wpisując jedynie partię prawej ręki, w przypadku polifonii wokalne zaś będzie to zwykle głos najwyższy. Oczywiście dla każdego z rekordów RISM można zapisać incypity wszystkich głosów w przypadku utworów wielogłosowych, lecz ze względów praktycznych rozwiązanie takie jest stosowane niezwykle rzadko. Zapisanie w bazie RISM kompletnych dzieł muzycznych jest

31  
Według danych z maja 2015 baza RISM zawiera ponad milion rekordów opisujących źródła oraz 30 tysięcy rekordów kompozytorów.

32  
Zob. [opac.rism.info/search?id=300257617](http://opac.rism.info/search?id=300257617) [dostęp: 20 V 2017].



33  
Krótki podręcznik dla kodu P&E odnajdziemy na stronie International Association of Music Libraries, Archives and Documentation Centres, [www.iaml.info/plain-e-easie-code](http://www.iaml.info/plain-e-easie-code) [dostęp: 20 V 2017].

34  
Dokumentacja dla formatu dostępna jest na stronie CCARH pod adresem [www.ccarh.org/publications/reprints/musedata95/musedata95.pdf](http://www.ccarh.org/publications/reprints/musedata95/musedata95.pdf) [dostęp: 20 V 2017].

35  
Prace zespołu kierowanego przez Waltera Hewletta skupiły się na repertuarze muzyki europejskiej z okresu 1700–1825. Kompozytorzy reprezentowani w bazie to: Jan Sebastian Bach, Ludwig van Beethoven, Arcangelo Corelli, Jerzy Fryderyk Händel, Józef Haydn, Wolfgang Amadeusz Mozart, Giovanni Rovetta, Jerzy Fryderyk Telemann oraz Antonio Vivaldi.

36  
Zob. [www.melodic-match.com/](http://www.melodic-match.com/) [dostęp: 22 V 2017].

37  
Zob. [web.mit.edu/music21/](http://web.mit.edu/music21/) [dostęp: 22 V 2017].

niemożliwe nie tylko z powodu ogromu pracy, jaką należałoby włożyć w wykonanie transkrypcji źródeł, ale także z uwagi na ograniczenia formatu P&E. Obserwacja powyższa nie stanowi w żadnym razie krytyki samego przedsięwzięcia – z założenia ma ono inne cele. Niemniej, jedynie repozytoria zawierające kompletne transkrypcje pozwalają na przeprowadzanie analizy skomputeryzowanej oraz wypracowanie wniosków opartych na badaniach statystycznych całych grup repertuaru.

Cele takie realizowane są przez działające przy Uniwersytecie Stanforda Center for Computer Assisted Research in the Humanities. Już w latach 80. XX wieku w CCARH rozpoczęto prace nad stworzeniem bazy zawierającej możliwie szerokie spektrum repertuaru muzyki klasycznej zakodowanej w formacie *MuseData*<sup>34</sup>. Składnię formatu oraz programy pozwalające na analizę zakodowanych utworów opracował Walter Hewlett. Aktualnie repozytorium *MuseData* zawiera niemal 900 zakodowanych kompozycji, z czego większość to duże utwory cykliczne, jak np. kantaty oraz pasje Jana Sebastiana Bacha czy symfonie Ludwiga van Beethovena<sup>35</sup>. Dzięki opracowanemu w CCARH oprogramowaniu możliwe jest renderowanie partytur z zapisu symbolicznego do formatu .pdf. Tak wygenerowane składy cechuje wysoka jakość, choć oczywiście nie są one wolne od niedociągnięć.

Przykład 6.

Jan Sebastian Bach, *Fuga a-moll*, DWK II, t. 5–7.



O użyteczności formatu *MuseData* świadczy to, że właśnie na jego podstawie (oraz na bazie formatu *Humdrum*) opracowano najpowszechniej obecnie używany standard wymiany danych muzycznych, jakim jest *MusicXml*. Niemal każdy z edytorów nutowych (zarówno komercyjnych, jak i udostępnianych na wolnych licencjach) obsługuje ten format. Dostępne jest także oprogramowanie pozwalające wykonywać skomputeryzowaną analizę plików *MusicXml* – jest to m.in. komercyjny program *MelodicMatch*<sup>36</sup> oraz zestaw narzędzi *Music21*<sup>37</sup>. Nie ulega jednak wątpliwości, że aktualnie największe możliwości w zakresie wspomaganego komputerowo analizy muzycznej dają narzędzia *Humdrum*. Także w tym formacie dostępne są najprawdopodobniej największe zasoby zakodowanej muzyki klasycznej, udostępniane przez CCARH.

## 2.2. Składnia języka *Humdrum*

Format *Humdrum* opracowany został przez Davida Hurona na Ohio State University w latach 80.<sup>38</sup>. Założeniem twórcy systemu

narzędzi oraz reguł gramatycznych *Humdrum* było stworzenie uniwersalnego zestawu narzędzi pozwalającego zanotować abstrakcyjną informację muzyczną w sposób umożliwiający operowanie danymi w zależności od celów analitycznych. Partytura w formacie .k<sup>39</sup> przypomina arkusz kalkulacyjny, w którym w kolumnach (zwanymi *spine*, ang. kręgosłup) od lewej strony zapisano głosy od najniższego do najwyższego. Na stronie dokumentacji systemu odnajdujemy ilustrację wyjaśniającą zwięźle ideę symbolicznego odwzorowania notacji muzycznej na przykładzie głównego tematu *Kunst der Fuge* Bacha<sup>40</sup>.

Przykład 7.

Jan Sebastian Bach, Temat główny *Kunst der Fuge* zanotowany w formacie .k<sup>39</sup>.

**Start of kern data spine**

**G** clef, 2nd staff line of staff  
key signature (**B-flat**)  
key, **D minor**  
meter, **2/2**  
metric sign (**common time**)  
first bar (styled invisible)  
half note D4, stem up  
half note A4, stem up  
second bar  
half note F4, stem up  
half note D4, stem up  
third bar  
half note C#4, stem up  
quarter note D4, stem up  
quarter note E4, stem up  
fourth bar  
start tie half note F4, su  
8th F4, su, tie end, beam start  
eighth G4, stem up  
eighth F4, stem up  
8th E4, stem up, beam end  
fifth bar  
quarter D4, stem up  
**end of kern data spine**

**\*\*kern**  
\*clefG2  
\*k[b-]  
\*d:  
\*M2/2  
\*met(c)  
=1-  
2d/  
2a/  
=2  
2f/  
2d/  
=3  
2c#/  
4d/  
4e/  
=4  
[2f/  
8f]/L  
8g/  
8f/  
8e/J  
=5  
4d/  
\*-

Syntaksa plików .k<sup>39</sup> jest z jednej strony niezwykle rygorystyczna – większość informacji można zakodować tylko w jeden, precyzyjnie określony sposób. Jest to wyjątkowo ważne, ponieważ gwarantuje, że dane zostaną zinterpretowane przez komputer w sposób jednoznaczny. Z drugiej jednak strony *Humdrum* jest nader elastyczny i pozwala na tworzenie dodatkowych – oprócz uprzednio zdefiniowanych – typów reprezentacji. W oficjalnej dokumentacji zdefiniowano 44 takie

38  
David Huron, *The Humdrum Toolkit Manual. Reference Manual*, Center for Computer Assisted Research in the Humanities, Menlo Park 1994.

39  
Zwyczajowo przyjętym rozszerzeniem dla plików *Humdrum* wywodzącym się od głównego typu reprezentacji danych muzycznych (*kern* – od niemieckiego *Kern* oznaczającego „rdzeń”) jest rozszerzenie .k<sup>39</sup>.

40  
Źródło ilustracji:  
[www.humdrum.org/guide/ch02/](http://www.humdrum.org/guide/ch02/)  
[dostęp: 23 V 2017].

41  
Zob. [www.humdrum.org/rep/](http://www.humdrum.org/rep/)  
[dostęp: 23 V 2017].

typy<sup>41</sup>, wśród których główne to **\*\*kern** (zapis informacji muzycznej, jak wysokość i czas trwania dźwięków) oraz **\*\*text** (zapis tekstu słownego dla utworów wokalnych i wokально-instrumentalnych). Pozostałe z typów reprezentacji mogą zawierać informację analityczną, np. **\*\*harm** służący do zanotowania nazw akordów z perspektywy harmoniki funkcyjnej czy **\*\*cents** do zapisania absolutnej wysokości dźwięków w odniesieniu do dźwięku *c* w oktawie razkreslonej (gdzie *c<sup>l</sup>* jest równe 0 centów). Nowe typy reprezentacji mogą być dodawane także w celu wyodrębnienia informacji ze standardowego zapisu, aby ułatwić jej osobną analizę – np. oznaczenia dynamiczne można zapisać obok symboli nut w kolumnie **\*\*kern**, ale też – stworzyć osobną kolumnę **\*\*dynam**, w której wpisujemy stosowne oznaczenia.

Jak już wspomniano, kolumny zawierające zapis muzyczny oznaczane są jako **\*\*kern**. Tylko tak nazwana kolumna (*spine*) może zostać zinterpretowana przez narzędzia *Humdrum* jako zapis muzyczny. Kolumny oddzielone są pojedynczym znakiem tabulatora, w przeciwnym przypadku program poinformuje nas o błędach składni. Znaki przykluczowe dodaje się poprzez wpisanie ich w sposób następujący: *\*k[]* oznacza brak znaków, *\*k[b-e-]* oznacza bemole „b” oraz „es”, *\*k[#c#]* zaś krzyżyki „fis” oraz „cis” etc. Kreski taktowe dla kolejnych taktów oznaczane są za pomocą znaku =, po którym następuje numer taktu: np. =2. Nazwy dźwięków zapisywane są w konwencji angielskiej, wedle której *C* oznacza dźwięk „c” w oktawie małej, *c* to dźwięk „c” w oktawie razkreslonej, *FF* to „f” w oktawie wielkiej, natomiast *dd* to dźwięk „d” w oktawie dwukreslonej itd. Wartości rytmiczne koduje się za pomocą liczb: np. 2 to półnuta, a 8. to ósemka z kropką<sup>42</sup>.

Niezmiernie istotnym elementem składni *Humdrum* są kropki. Ponieważ w kolejnych liniach kodu notowane są następujące po sobie „zdarzenia” muzyczne, możliwa jest sytuacja, w której w danym głosie pojawia się kilka nut (zdarzeń) przypadających na jedną nutę w pozostałych głosach. W przykładzie 8 takie okoliczności zachodzą kilkakrotnie. Podobnie jak w językach programowania składnia *Humdrum* pozwala nam na wprowadzanie komentarzy, które mogą zawierać cenne informacje (jak np. uwagi edytorskie). Komentarze nanosimy, poprzedzając je symbolem wykrzyknika. Jest to narzędzie szczególnie przydatne, kiedy chcemy uwzględnić w partyturze swoje własne uwagi oraz – co bardzo istotne – uzupełnić plik o niezbędne metadane<sup>43</sup>. Wprowadzanie metadanych do pliku *Humdrum* to praktyka wskazana z kilku powodów. Pierwszym jest zapewnienie niezbędnej porcji informacji dla wszystkich użytkowników, którzy korzystać będą z pliku, w tym np. o prawach autorskich czy licencji, na jakiej udostępniany jest plik. Jednak wpisywanie metadanych ma szczególne znaczenie, kiedy tworzymy bazę danych. Zasady komentowania pliku *Humdrum* są bardzo elastyczne, tzn. można wprowadzać własne elementy metadanych.

42

Z pełnym opisem składni można zapoznać się na oficjalnej stronie *Humdrum*: <http://www.humdrum.org/> [dostęp: 24 V 2017]. Pełną dokumentację w pliku .pdf można też pobrać ze strony CCARH: <http://humdrum.ccarh.org/> [dostęp: 24 V 2017].

43

Charakterystykę typów metadanych *Humdrum* możemy odnaleźć na stronie <http://www.humdrum.org/Humdrum/guide.append1.html>.

Przykład 8.

Jan Sebastian Bach, Kantata BWV 4, aria *Jesus Christus Gottes Sohn*,  
t. 5-7.

Zapis t. 5 przykładu nutowego nr 8 w formacie *.kern*:

**kern	**kern	**text	**kern
*clefF4	*clefGv2	*	*clefG2
*k[f#]	*k[f#]	*	*k[f#]
*M4/4	*M4/4	*	*M4/4
=5-	=5-	=5-	=5-
4EE	4r	.	16eL
.	.	.	16ee
.	.	.	16dd#
.	.	.	16eeJ
8GL	4B	Je-	16eL
.	.	.	16ee
8C#J	.	.	16ddn
.	.	.	16eeJ
8F#L	4A#	-sus	16cc#L
.	.	.	16b
8E	.	.	16a#
.	.	.	16g#J
8D	8BL	Chri-	16f#L
.	.	.	16ff#
8AA#J	8c#J	.	16ee
.	.	.	16ff#J
=	=	=	=
*_	*_	*_	*_

Przedstawiony powyżej przykład fragmentu zakodowanego utworu w formacie *.kern* jest stosunkowo prosty, ponieważ każdy z głosów został zapisany na osobnej pięciolinii. Kod staje się o wiele bardziej skomplikowany, gdy w jednej partii symultanicznie występuje kilka planów muzycznych – może to być *divisi* lub faktura polifoniczna. Komplikacja ta jest szczególnie częsta w przypadku muzyki fortepianowej. Możemy sobie wyobrazić, że dla osoby kodującej zapis muzyczny i mającej odpowiednie przygotowanie najbardziej wydajnym sposobem wykonywania transkrypcji będzie bezpośrednie

zapisywanie kodu *Humdrum*. Zadanie to jest tylko nieco bardziej złożone, kiedy dotyczy utworów wielogłosowych. Jednak już zakodowanie w ten sposób utworów Chopina byłoby w większości przypadków niezmiernie trudne i czasochłonne. W tym celu opracowano konwertery pozwalające automatycznie generować kod pliku .krn z innych formatów, jak np. *MusicXml*<sup>44</sup>. Dzięki wsparciu Narodowego Instytutu Fryderyka Chopina we współpracy z CCARH aktualnie opracowywana jest nowa wersja konwertera *musicxml2hum* umożliwiająca konwersję z formatu *MusicXml* do *Humdrum* bez utraty informacji. Konwerter ten uwzględnia problemy, jakie wiążą się z kodowaniem muzyki fortepianowej wieku XIX, zwłaszcza twórczości Chopinowskiej. Jest to niezwykle istotne, konwerter ten pozwoli bowiem przygotowywać pliki .krn za pomocą powszechnie znanych programów do edycji nut generujących dane w xml oraz – co szczególnie cenne – umożliwi pracę z programami typu OMR (*Optical Music Recognition*). Zarówno programy komercyjne, typu *Finale* czy *Sibelius*, jak i darmowy *MuseScore* zawierają co najmniej kilka interfejsów do wprowadzania danych – te można wpisywać z klawiatury komputera, ale też dzięki klawiaturze midi: „zagrać” utwór, jednocześnie go zapisując. Nader obiecująca wydaje się natomiast perspektywa częściowego zautomatyzowania procesu wprowadzania danych z użyciem programów OMR. Wyniki testów różnego rodzaju programów typu OMR, jakie prowadzono w ostatnim czasie w NIFC, pozwalają z dużym optymizmem patrzeć na tę metodę. W zależności od jakości źródła skuteczność OMR kształtuje się na poziomie 60–90% (dane na podstawie przeprowadzonych prób rozpoznawania tekstu muzycznego skanów pierwodruków wybranych dzieł Chopina ze zbiorów Biblioteki NIFC), co oznacza, że nawet w najbardziej pesymistycznym wariantcie szybkość uzyskiwania zapisu symbolicznego jest znacznie większa niż w przypadku ręcznego wprowadzania (licząc z koniecznymi korektami wyjściowego materiału OMR). Warto zauważyć, że właśnie w ten sposób udało się w stosunkowo niedługim czasie stworzyć ogromną bazę kompozycji renesansowych w projekcie *Josquin Research Project* (JRP) kierowanym przez prof. Jessego Rodina na Uniwersytecie Stanforda. Dzięki metodologii opracowanej głównie przez dra Craiga Sappa w CCARH aktualnie baza JRP obejmuje 760 kompozycji (w tym duże cykle mszalne Josquina i in.), co daje w sumie 1 181 465 zakodowanych nut przekładających się na ponad 76 godzin muzyki i 5 961 stron zapisu nutowego<sup>45</sup>. Oczywiście statystyka powyższa generowana jest automatycznie za pomocą narzędzi *Humdrum*. Jest to zatem imponującej wielkości kolekcja, na której podstawie można przeprowadzać wnioski statystyczne i stylokrtyczne.

Jesse Rodin oraz Clare Bokulich w wystąpieniu prezentowanym podczas spotkania American Musicological Society w Nowym Orleanie w 2012 argumentowali, że tego typu analiza może być niezwykle przydatna<sup>46</sup>. Wykorzystując bazę JRP, autorzy wykazali, że za pomocą narzędzi analizy skomputeryzowanej można opisać różnice

44

Autorem większości programów umożliwiających wielokierunkową konwersję formatu .krn (do formatu i z formatu .krn) jest dr Craig Stuart Sapp z CCARH.

45

Dane ze strony projektu: josquin.stanford.edu/ [dostęp: 25 V 2017].

46

Clare Bokulich i Jesse Rodin, „A 'Large Mass of Facts'”, referat wygłoszony podczas konferencji American Musicological Society, New Orleans 2012.

stylistyczne pomiędzy opracowaniami mszalnymi Ockeghema i Josquina. Rodin i Bokulich zadają pytanie: „Jaki obraz historii muzyki uzyskalibyśmy, gdyby zamiast – jak do tej pory – skupiać się na dziwacznych sposobach użycia *musica ficta*, pojedynczych przypadkach traktowania *cantus firmus* czy nietypowych znakach maturalnych, przyjrzelibyśmy się bliżej przyziemnym, zbyt oczywistym, niewartym wspomnienia, codziennym problemom, jak wzory rytmiczne, konwencjonalne skoki melodyczne, standardowe zakresy głosów” [tł. MK]<sup>47</sup>. Możliwość przeprowadzenia analogicznych do powyższych badań repertuaru muzyki polskiej wieku XIX wydaje się szczególnie atrakcyjna. Stworzenie podobnej do JRP bazy obejmującej omawiany repertuar pozwoliłoby na budowanie wniosków dotyczących stylistyki twórczości kompozytorów polskich na podstawie danych empirycznych uzyskanych dzięki analizie obszernego ilościowo materiału. Wyniki takich badań mogłyby się okazać nadzwyczaj przydatne w odniesieniu do – licznych przecież – źródeł przekazanych anonimowo. W rozważaniach dotyczących problemu atrybucji tego repertuaru badania stylokrytyczne zwykle nie przynoszą jednoznacznych odpowiedzi (zwłaszcza w przypadku dzieł obejmujących określone konwencje stylistyczne), niemniej stanowić mogą jeden z ważniejszych argumentów w takiej dyskusji. Dowodzi tego przykład JRP, gdzie na podstawie analizy całego korpusu Jesse Rodin mógł wyodrębnić grupy kompozycji, które można z pewnością uznać za dzieła Josquina, jak też utwory o wątpliwej atrybucji lub błędnie przypisywane temu autorowi. W przypadku analizy repertuaru muzyki wieku XIX takie wnioskowania oparte na analizie skomputeryzowanej mają jeszcze mocniejsze podstawy, ponieważ język muzyczny twórców tego okresu był bardziej zindywidualizowany niż w czasach polifonii flamandzkiej. Aktualnie w NIFC podejmowane są próby zainicjowania prac mających na celu stworzenie bazy zakodowanych kompozycji, która pozwoli w przyszłości na prowadzenie badań wspomaganych komputerowo. Nie mamy zatem obecnie możliwości badania dużego korpusu dzieł. Niemniej, aby zaprezentować podstawowe narzędzia analityczne systemu *Humdrum*, możemy skupić się na wybranych dziełach<sup>48</sup>. Na potrzeby niniejszej prezentacji wybraliśmy dwa interesujące utwory cykliczne: *Mszę C-dur* Cyryla Gieczyńskiego oraz *Mszę C-dur* op. 26 Józefa Elsnera.

### 3. Analiza skomputeryzowana

Ojciec Cyryl Florian Gieczyński [Goetz] (1793–1866) to pochodzący z Moraw i przybyły na Jasną Górę w 1814 roku kompozytor oraz zakonnik. Święcenia kapłańskie przyjął na Jasnej Górze w 1816 roku, obierając imię zakonne Cyryl. Pod koniec 1819 roku przeniósł się do warszawskiego kościoła św. Ducha (zmienił wtedy nazwisko z Goetz na Gieczyński), by wreszcie od 1821 roku na stałe osiąść

47  
„[...] [We] would like to pose a similar question of Renaissance music – a question musicologists started asking 40 years ago and have now begun to ask again: what would happen if we turned away from bizarre uses of *musica ficta*, singular instances of *cantus-firmus* treatment, and unusual mensuration signs and instead fixed our gaze on the mundane, the all-too-common, the barely-worthy-of-mention – on everyday rhythmic patterns, routine melodic leaps, and standard voice ranges: what sort of music history would we find there?” (C. Bokulich, J. Rodin, op. cit., s. 2).

48  
Warto podkreślić, że analizę opartą na systemie *Humdrum* w odniesieniu do interesującego nas repertuaru z powodzeniem prowadziła Iwona Lindstedt – zob. I. Lindstedt, *Some Remarks on Computer-Assisted Analysis of the Finale of Chopin's Piano Sonata in b flat minor op. 35*, w: *Analytical Perspectives on the Music of Chopin*, red. Artur Szkleener, NIFC, Warszawa 2004, s. 203–223.



w podwarszawskiej parafii w Niegowie. Wiadomo, że w okresie pobytu w Warszawie przyjaźnił się z Józefem Elsnerem.

Obydwa z poddawanych analizie utworów zostały przekazane w źródłach przechowywanych w Archiwum oo. Paulinów na Jasnej Górze, przy czym rękopis mszy Gieczyńskiego jest autografem, msza Elsnera zaś zachowała się w druku z 1823 roku. *Missa in C* Gieczyńskiego napisana jest na duży skład, obejmujący: CATB, 2 fl, 2 cl, 2 fg, 2 clni (zastąpione w wybranych fragmentach przez 2 cor), timp, 2 vn, vla, org. Jeżeli sugerować się wpisem na karcie tytułowej *Sancto Joanne Nepomuceno*, można przyjąć, że utwór został przygotowany na święto św. Jana Nepomucena, którego wspomnienie w kalendarzu liturgicznym przypadało na 21 maja. Niewykluczone zatem, że prawykonanie kompozycji miało miejsce właśnie 21 maja 1816 roku. Dzieło Elsnera opublikowane w 1823 roku powstało trzy lata wcześniej (w 1820 roku), zatem obydwie utwory dzieli niewielki dystans czasowy. Elsner swoją mszę napisał na obsadę CATB, 2 fl, 2 cl, 2 fg, 2 cor, 2 clni, b-trb, timp, 2 vn, vla, org. Obydwaj autorzy związani byli ściśle z tymi samymi środowiskami – Gieczyński jako zakonnik (później, po sekularyzacji i przejściu do kleru diecezji warszawskiej nadal pozostawał w ścisłym kontakcie z macierzystym klasztorem), Elsner wiele ze swoich utworów religijnych dedykował jasnogórskim paulinom, a sam został nawet zaliczony w poczet konfratrów tego zakonu. I Gieczyński, i Elsner współtworzyli aktywnie warszawskie życie muzyczne.

Rozpocznijmy od porównania statystyki „obiektów” w obydwu cyklach<sup>49</sup>:

49  
Dla ułatwienia w przypisie podajemy ciąg komend terminala, za których pomocą uzyskujemy dany wynik. Dla uzyskania statystyki obiektów całego cyklu należy wpisać komendę: `census -k *.krn`.

50  
Dla utworu Gieczyńskiego suma podwójnych kresek taktowych kończących część równa się liczbie części stałych opracowania mszalnego. Elsner natomiast wydziela dodatkowo jako odrębne części muzyczne *Benedictus* oraz *Dona nobis pacem*.

51  
Różnica pomiędzy liczbą nut (*notes*) a główek nut (*note-heads*) wynika z tego, że `census` liczy osobno także nuty ligowane.

52  
Tabele graficzne przygotowano za pomocą: <https://datawrapper.de/> [dostęp: 27 V 2017]. W tabeli 9 skrót Els-B oznacza partię basu w mszy Elsnera, Gie-B partię basu w mszy Gieczyńskiego etc.

	Elsner	Gieczyński
Number of note-heads:	36064	37706
Number of notes:	35848	37686
Highest note:	aaa-	bbb-
Lowest note:	CC	FF
Number of rests:	10472	10997
Number of single barlines:	674	728
Number of double barlines:	7	5

Msza Gieczyńskiego jest nieco obszerniejsza (w sumie 728 taktów)<sup>50</sup> w porównaniu do utworu Elsnera (674 takty). Aktywność głosów liczona jako średnia liczba nut w ramach taktu<sup>51</sup> (rozpatrywana globalnie) w obydwu kompozycjach jest zbliżona – u Elsnera to średnio 53 nuty w ramach taktu, u Gieczyńskiego niemal 52. Podobnie jest w przypadku odsetka pauz, które wynoszą odpowiednio 6,43% (Elsner) i 6,61% (Gieczyński).

W kolejnym kroku przyjrzymy się statystyce zawartości interwałów melodycznych w głosach wokalnych. Pozwoli to określić sposób prowadzenia tych głosów (bardziej melodyjny, gdy dominują kroki sekundowe)<sup>52</sup>.



## Statystyka interwałów melodycznych głosów wokalnych.

	Els-B	Gie-B	Els-T	Gie-T	Els-A	Gie-A	Els-S	Gie-S
1	652	444	673	532	677	569	562	471
2	493	301	575	362	814	221	858	420
3	157	187	146	117	137	241	149	163
4	110	189	79	44	61	40	66	66
5	110	104	19	27	10	21	28	31
6	14	16	27	10	12	6	25	14
7	2	9	3	0	3	0	5	3
8	27	82	3	1	1	1	3	3
9	0	7	0	0	0	0	0	0
11	0	0	0	1	0	0	0	1

Jak widać na przykładzie 9, w obydwu przypadkach dominujące typy interwałów melodycznych w głosach wokalnych to – zgodnie z oczekiwaniami – repetycje oraz kroki sekundowe<sup>53</sup>. Nie zaskakuje także odpowiednio większy odsetek skoków kwarty oraz kwinty w głosie najniższym, co wynika z jego roli podstawy harmoniczej. Aby wskazać na ewentualne różnice w traktowaniu tej partii przez obydwu autorów, można zbadać relacje interwałowe pomiędzy najniższym głosem wokalnemu a partią organów – w ten sposób dowiemy się, do jakiego stopnia głos wokalny pełni niezależną funkcję. W mszy Elsnera 75,06% współbrzmień pomiędzy wskazanymi partiami to unison, następnie oktawy (8,76%) oraz tercje wielkie (2,38%). Dla kompozycji Gieczyńskiego wartości te to odpowiednio 49,39% unisony, 17,98% oktawy, 7,26% seksty małe, 5,49% kwarty czyste oraz 4,56% tercje wielkie. Niemal 84% współbrzmień unisono oraz oktawy pomiędzy wokalnemu basem a partią organów u Elsnera stanowi znaczny odsetek świadczący o małej samodzielności głosu. W utworze Gieczyńskiego interwały te sumarycznie dają 67,37%, co pozwala postawić tezę, że basowy głos wokalny jest traktowany o wiele swobodniej. Jest to efektem m.in. powierzenia partii basu roli solowej w większej liczbie odcinków niż w przypadku mszy Elsnera. Jeśli przyjrzeć się dokładnie partyturze, okaże się także, że Gieczyński o wiele częściej prowadzi omawiane partie niezależnie, podczas gdy nawet w swoich ustępach solowych partia basu u Elsnera jest zwykle dublowana przez partię organów. Bez wątpliwości natomiast Elsner o wiele bardziej melodyjnie kształtuje partie altu oraz sopranu (zob. przykład 9). I rzeczywiście, melodyka głosów wokalnych mszy Gieczyńskiego ma w wielu momentach charakter deklamacyjny, oparty na repetycji.

53

W zestawieniu nie brano pod uwagę jakości ani kierunku interwału.

Cyryl Gieczyński, *Msza C-dur, Kyrie*, t. 30–32 (CATB).

30

The image shows a musical score for four voices (Soprano, Alto, Tenor, Bass) in 4/4 time, key of C major. The lyrics are: Ky ri - e e - lei - son Ky - ri - e e - lei - son. The score is numbered 30.

Jeszcze innym możliwym sposobem badania melodyki danego głosu jest stworzenie n-gramów wzorców melicznych, co pozwala na wyodrębnienie najczęściej powtarzanych ukształtowań<sup>54</sup>.

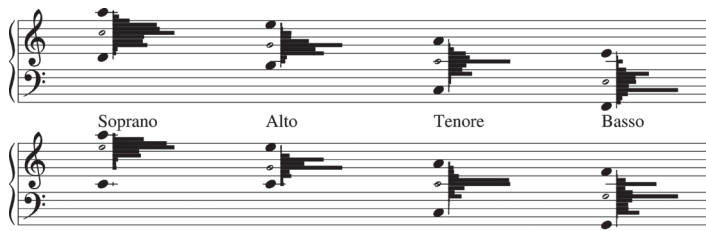
Elsner				Gieczyński			
120	P1	P1	P1	146	P1	P1	P1
28	P1	P1	-M2	25	+m2	P1	P1
28	-M2	P1	P1	24	P1	P1	+m2
27	P1	-M2	-M2	24	P1	-m2	P1
25	+M2	+m2	-m2	23	P1	+m2	P1

W obydwu przypadkach repetycja odgrywa główną rolę w kształtowaniu melodyki najwyższego z głosów wokalnych, niemniej jeśli porównać wyniki, wyraźnie widać szczególną predylekcję Gieczyńskiego do powtarzania dźwięków.

Istotną informacją dotyczącą sposobu budowania danej partii jest jej tessitura. Przykład II ukazuje zakresy głosów wokalnych w połączeniu z histogramem częstotliwości występowania danego dźwięku. Cała nuta ukazuje medianę, co daje pojęcie o wykorzystaniu poszczególnych rejestrów.

54  
 extractx -s 8 \*.krn |  
 mint -s [\\_\ ] | ridx -H |  
 grep -v r | context -n 3 |  
 sortcount

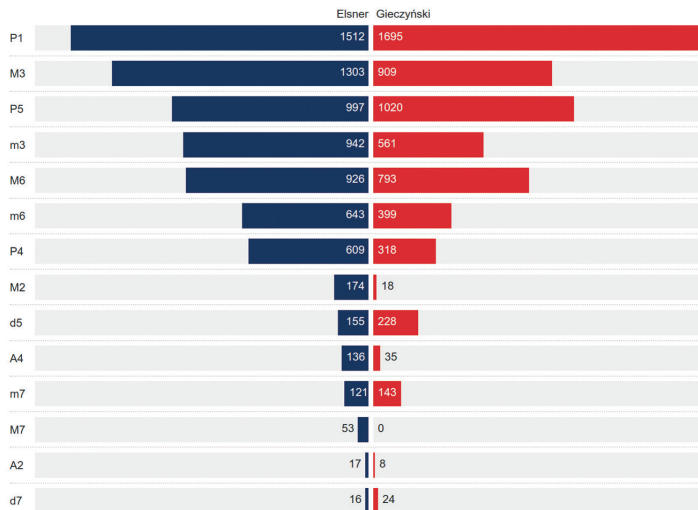
## Zakresy głosów wokalnych.



W zakresie ambitusów głosów wokalnych<sup>55</sup> trudno wskazać na istotne różnice pomiędzy obydwoma utworami. Jedyną cechą wyraźnie różnicującą analizowane kompozycje pod tym względem jest wykorzystanie rejestrów głosu sopranowego. U Elsnera histogram rozkłada się równomiernie, z tendencją do wykorzystania rejestru wygodnego dla śpiewaków, podczas gdy Gieczyński wyraźnie sytuuje sopran w górnym rejestrze.

Przyjrzyjmy się teraz zawartości interwałów harmoniczych w głosach wokalnych. Za pomocą programu *hint* zbadamy częstość ich występowania, przy czym badamy głosy w permutacji (interwały pomiędzy wszystkimi głosami, a nie tylko kolejnymi) oraz sprawdzając interwały większe od septymy do ich przewrotów (zatem oktawa będzie traktowana jak unison, decyma jak tercja itp.).

## Statystyka interwałów harmoniczych dla głosów wokalnych.



55 Wykres górny ukazuje ambitus głosów wokalnych mszy Elsnera, dolny mszy Gieczyńskiego.

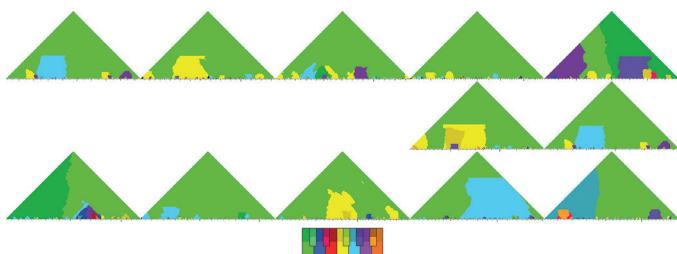
Histogram – przykład 12 – jednoznacznie wskazuje na akordowy sposób prowadzenia głosów – świadczy o tym dominacja współbrzmień tercjowych, kwintowych oraz kwartowych i sektowych. Zastanawia nieco większa przewaga współbrzmień unisonu/oktawy w przypadku utworu Gieczyńskiego. Jeśli dokładniej przeanalizować partyturę, można zauważyć, że wspomniana wcześniej predylekcja do deklamacyjnego kształtowania melodyki głosów wokalnych realizowana jest nie tylko poprzez zastosowanie repetycji, ale także równoległe prowadzenie głosów w unisonach i oktawach (co da się zauważyć m.in. na przykładzie 10, t. 31). Frapujący jest zupełny brak współbrzmienia septymy wielkiej w partiach głosów wokalnych – jednak ten składnik akordu septymowego jest przez Gieczyńskiego wprowadzany w partiach instrumentalnych.

Co oczywiste, zaprezentowane powyżej typy analizy można zastosować także do partii instrumentalnych, w efekcie uzyskując wyczerpujący opis statystyczny dzieła, pozwalający na stworzenie swoistego modelu stylokrytycznego. Tylko użycie większej liczby różnorodnych narzędzi prowadzi do osiągnięcia takiego efektu – większość z wyników analizy skomputeryzowanej wymaga dalszej interpretacji. Siłą narzędzi *Humdrum* polega jednak właśnie na możliwości analizowania dużych korpusów dzieł. Niezwykle ważna jest też opcja tworzenia modeli wizualizacji wyników analizy. Wartość heurystyczna takich właśnie modeli jest ogromna, pozwala bowiem stosunkowo szybko obserwować pewne cechy charakterystyczne i podobieństwa. Przykładem może być analiza wykonana za pomocą programu *mkeyscape*.

Generowany w formie trójkąta model ukazuje wyniki analizy tonalnej, gdzie poszczególne kolory oznaczają daną tonację. Odcienie jaśniejsze zarezerwowane są dla trybu durowego, ciemniejsze – molowego. Kolory odpowiadają tonacjom, co uwidoczniło na legendzie. U podstawy trójkąta znajdują się kreski oznaczające kolejne takty. Model należy interpretować następująco. Najniższy poziom prezentuje bezpośrednie następstwo kolejnych akordów. Przenosząc się o poziom wyżej, uzyskujemy obraz tonacji pełniących strukturalnie ważniejszą funkcję w utworze. Na szczycie diagramu powinna być odzwierciedlona tonacja zasadnicza utworu. Analizę z wykorzystaniem programu *mkeyscape* można przeprowadzać, definiując różne profile ważenia oparte na wybranych koncepcjach analizy tonalnej. Wykresy tonalne sporządzane są przez algorytm „Krumhansl–Kessler” obliczający współczynnik korelacji liniowej Pearsona pomiędzy histogramem klas wysokości dźwięków a zdefiniowanymi uprzednio profilami dla trybów durowego i molowego<sup>56</sup>.

56

Carol L. Krumhansl, Jamshed J. Bharucha, Edward J. Kessler, *Perceived Harmonic Structure of Chords in Three Related Musical Keys*, „Journal of Experimental Psychology”, 8 (1), 1984, s. 24–36.

Modele harmoniczne *mkeyscape*.

Modele ukazane na przykładzie 13 nie tylko pozwalają zaobserwować plan tonalny kompozycji, ale także śledzić ewentualne powtórzenia materiału. Na górnej części grafu widzimy modele dla kolejnych części opracowania *ordinarium missae* Elsnera (z części *Sanctus* kompozytor wyodrębnił *Benedictus*, podobnie jako osobną część potraktował fragment *Agnus Dei – Dona nobis pacem*). Już pobieżny rzut oka na modele tonalne pozwala zauważyć, że są one identyczne w mszy Elsnera dla części *Kyrie* oraz *Dona nobis pacem*. To efekt zastosowania częstego w opracowaniach mszalnych epoki klasycyzmu zabiegu powtórzenia materiału muzycznego pierwszej części mszy w części końcowej (zwykle nie rozpisywano nawet osobno *Dona nobis*, ograniczając się do wpisania w partyturę oznaczenia w rodzaju *Dona ut Kyrie* i pozostawiając śpiewakom kwestię odpowiedniego podłożenia tekstu słownego). Bardzo wyraźnie widać też u Elsnera tendencję do budowania planu tonalnego części zgodnie z założeniami stylistyki klasycznej, według których po fragmencie początkowym następuje modulacja do tonacji dominanty lub subdominanty, a następnie powrót to tonacji zasadniczej. Modele *mkeyscape* mszy Elsnera uwidaczniają nam także dużą stabilność tonalną – jedynie w części *Agnus Dei* kompozytor odważnie porzuca tonację główną. Potwierdza to również analiza tonacji za pomocą programu *key*, który wyłącznie w tym przypadku określa stopień trafności na poziomie niższym niż 50%.

Estimated key: C major	(r=0.9417)
confidence: 69.5%	
Estimated key: C major	(r=0.9651)
confidence: 90.8%	
Estimated key: C major	(r=0.9556)
confidence: 100.0%	
Estimated key: C major	(r=0.9584)
confidence: 72.1%	
Estimated key: C major	(r=0.9759)
confidence: 90.3%	
Estimated key: C major	(r=0.8571)
confidence: 26.0%	
Estimated key: C major	(r=0.9417)
confidence: 69.5%	

Obserwowanych u Elsnera klasycznych reguł kształtowania planu tonalnego brak u Gieczyńskiego, co niekoniecznie musi oznaczać większy poziom komplikacji harmoniczných. Odmiennie są jedynie zasady budowania planu, natomiast kolejne części opracowania tego autora cechuje właściwie jeszcze wyraźniejsza stabilność tonalna. Na najniższym poziomie grafu daje się zaobserwować dość duża zmienność harmoniczną u Elsnera, której – nie licząc części *Kyrie* – brak u Gieczyńskiego.

Zaprezentowany powyżej drobny wycinek możliwości, jakie daje analiza z wykorzystaniem systemu *Humdrum*, pozwala przekonać się o potencjalnej przydatności tego typu narzędzi do badania repertuaru muzyki polskiej wieku XIX. Należy podkreślić, że dzięki podejmowanym przez Narodowy Instytut Fryderyka Chopina działaniom coraz bardziej realne wydaje się stworzenie bazy transkrypcji utworów Chopina w formacie .krn, co umożliwi ich badanie z nowej, niewykorzystanej dotąd perspektywy. Chopin ma szansę stać się pierwszym znaczącym kompozytorem, którego komplet utworów zostanie opracowany i udostępniony w ten sposób. Docelowo baza transkrypcji powinna też zostać uzupełniona zwłaszcza o repertuar twórców mniej znanych i przekazany anonimowo, co z pewnością ułatwiłoby rozstrzygnięcie problemu atrybucji. To jednak nie jedyne szanse, jakie pojawiają się w związku z zastosowaniem omawianych narzędzi. Pamiętajmy także o pracy redaktorów przygotowujących wydania krytyczne. Jeżeli kompozycja zostaje zapisana w formacie tekstowym, możliwe jest automatyczne lub półautomatyczne generowanie komentarza źródłowego na podstawie porównania dwu plików: jednego zawierającego dokładne odwzorowanie tekstu źródłowego i drugiego, w którym naniesiono korekty redakcyjne. Tego typu rozwiązania stosowane są z powodzeniem w kilku przedsięwzięciach, m.in. przez zespół programistów przygotowujących składy nutowe w programie *LilyPond*. Zapisanie materiału nutowego w formacie .krn daje także nowe możliwości renderowania partytur, czyli przygotowywania składów nutowych nowego rodzaju – wydań cyfrowych.

#### 4. Renderowanie partytur

Jednym z największych wyzwań, jakie aktualnie stoją przed twórcami systemów umożliwiających zapisywanie muzyki w formacie tekstowym, jest wypracowanie narzędzi do renderowania partytur. Jest to zadanie niezwykle trudne z uwagi na wysoki stopień skomplikowania notacji muzycznej oraz jej wielowarstwową strukturę logiczną (konieczne jest jednoczesne zapisanie i wyświetlanie różnych typów informacji). Podstawową funkcjonalnością, jaką powinna cechować się baza zakodowanych utworów – oprócz możliwości wyszukiwania i wieloaspektowej analizy – jest wyświetlenie samego materiału muzycznego w postaci zrozumiałej dla odbiorcy. Najwygodniejszą

formą takiego zaprezentowania np. efektów przeszukania bazy pod kątem występowania danej melodii jest wyświetlenie fragmentów partytury zawierających relewantne wyniki. Możliwości takie daje już w tej chwili wspomniana uprzednio baza JRP.

Przykład 14.

Wynik wyszukania wzorca „d a b g a b d” w bazie JRP – wynik wyróżniony kolorem.



Osiągnięcie zadowalającego wyniku, jak na przykładzie 14, jest stosunkowo proste w przypadku renderowania partytury zawierającej polifonię wokalną – zazwyczaj są to składy, w których na jednej pięciolinii znajduje się jeden plan muzyczny. Znacznie trudniejsze jest automatyczne wygenerowanie poprawnego zapisu, gdy chodzi o muzykę fortepianową. Szczególnie kłopotliwe są sytuacje – liczne w twórczości Chopina – kiedy w partii danej ręki pojawia się efemeryczny motyw, rozpoczęty w połowie taktu, urwany równie niespodziewanie. *Humdrum* z uwagi na rygorystyczną składnię „nie rozumie” takiego zapisu i uzna go za błędny. Aby tego uniknąć, należy dla owego dodatkowego planu wprowadzić tzw. *subspine* (podkolumnę) oraz uzupełnić ją o brakujące pauzy (mogą one zostać ukryte i nie wyświetlać się w partyturze), tak aby zapis sumował się do aktualnego metrum. Zauważmy, że działanie takie w pewnym sensie może wypaczać intencję kompozytora. Dodatkowym problemem jest znaczne zaciemnienie kodu – w przypadku dodania większej liczby podkolumn staje się on w zasadzie nieczytelny. Co gorsza, aktualnie dostępne otwarte oprogramowanie do renderowania nut bardzo słabo radzi sobie z tą trudnością. W przypadku programów komercyjnych problem jest mniej istotny, ponieważ ich głównym celem jest zapisanie informacji graficznej, a nie danych logicznych, zatem możliwe jest zanotowanie (a właściwie narysowanie) najbardziej odbiegającego od zasad ortografii muzycznej tekstu. Obecnie właściwie tylko dwa programy oparte na symbolicznym zapisie nutowym pozwalają na wykonywanie takich operacji: *LilyPond*<sup>57</sup> oraz *Verovio*<sup>58</sup> (przykłady 15 i 16).

*LilyPond* to projekt rozwijany już od wielu lat przez grupę programistów (w tym także polskich). Głównym celem, jaki przyświeca twórcom tego oprogramowania, jest zautomatyzowanie procesu renderowania nut w celu uzyskania najwyższej jakości wyniku, bez konieczności licznych ingerencji człowieka. Należy zauważyć, że rzeczywiście jakość składów generowanych przez *LilyPond* jest na bardzo wysokim poziomie, sam program zaś daje ogromne

57  
Oficjalna strona projektu:  
<http://lilypond.org/>  
[dostęp: 30 V 2017].

58  
Strona projektu: <http://www.verovio.org> [dostęp: 30 V 2017].



możliwości w zakresie tworzenia dodatkowych elementów graficznych oraz poprawiania efektu końcowego. Pliki formatu .ly są plikami tekstowymi o niezwykle intuicyjnej składni (por. wcześniejszy przykład kodu). Pomimo tego można wskazać na co najmniej kilka powodów, dla których zapisywanie większego korpusu dzieł wyłącznie w tym formacie byłoby kłopotliwe. Pierwszym z problemów jest brak konwerterów umożliwiających „tłumaczenie” zapisu .ly na inne formaty. W zasadzie jedyną ewentualnością jest bądź napisanie kodu w formacie .ly, bądź jego konwersja z formatu .xml. Niestety, nie ma możliwości wyeksportowania zapisu z formatu .ly do innego, poza formatem .midi, w którym ogromna część istotnej informacji jest tracona. Innymi słowy, pliki takie mają funkcjonalność ograniczoną do tworzenia składu, bez opcji innego wykorzystania zapisanych danych. Ponadto zapis w formacie .ly nie jest tak jednoznaczny jak w przypadku składni *Humdrum*, stąd brak jakichkolwiek narzędzi umożliwiających wykonywanie analizy komputerowej tego typu plików. Osobną kwestię stanowi prędkość działania *LilyPonda*. Po wielu latach rozwoju kod programu jest niezwykle rozbudowany i jednocześnie oparty na kilku językach programowania. W efekcie wygenerowanie prostej partytury utworu na mały skład i stosunkowo krótkiego nie nastęcza większych trudności. Problem dotyczy większych kompozycji, w których przypadku renderowanie składu może trwać nawet kilka lub kilkanaście minut. Jakkolwiek efekt zwykle jest zadowalający, to długość procesu oznacza, że nie sposób wykorzystać *LilyPonda* do pracy w przeglądarce internetowej, w której wyrenderowanie wyników wyszukiwania całej bazy kompozycji trwałoby zbyt długo.

Z zagadnieniem tym znacznie lepiej radzi sobie *Verovio*, którego twórcą jest szwajcarski programista i muzykolog, Laurent Pugin. Napisany i rozwijany przez niego program jest aktualnie najbardziej obiecującym tego typu projektem. *Verovio* potrafi bardzo szybko stworzyć wysokiej jakości składy, spełniające standardy najlepszych wydań tradycyjnych. Co więcej, przyjmuje różne formaty wejściowe (głównym jest MEI) z uwagi na wbudowane konwertery. Niestety, wysoka jakość, o której mowa powyżej, dotyczy raczej muzyki epok wcześniejszych niż nas interesująca, ponieważ sam program został napisany głównie z myślą o muzyce dawnej. Aktualnie trwają jednak prace wspierane przez NIFC, mające na celu dostosowanie *Verovio* do wyświetlania muzyki Chopina. Co szczególnie ważne, dzięki pracom dra Craiga Sappa powstał konwerter *VerovioHumdrumViewer*, który umożliwia renderowanie partytury bezpośrednio z kodu *Humdrum*<sup>59</sup>.

59  
Z możliwościami programu można zapoznać się na stronie <http://verovio.humdrum.org/> [dostęp: 30 V 2017].

Józef Elsner, *Msza C-dur, Kyrie*, t. 1-7, wyrenderowane w programie *Verovio*.

The image shows a musical score for the Kyrie section of the Mass in C major by Józef Elsner. The score is rendered in Verovio and includes staves for various instruments and vocal parts. The instruments listed are Flauto I, Flauto II, Clarinetto I in C, Clarinetto II in C, Fagotto I, Fagotto II, Corno I in C, Corno II in C, Clarino I in C, Clarino II in C, Trombone Basso, Timpani in C G, Soprano, Alto, Tenore, Basso, Violino I, Violino II, Viola, and Organo. The vocal parts (Soprano, Alto, Tenore, Basso) have lyrics in Latin: "Ky ri e Ky ri e Ky ri e e - lei son Ky ri e e - lei". The score is in 6/8 time and C major.

Przykład 16.

Józef Elsner, *Msza C-dur, Kyrie*, t. 1-7, wyrenderowane w programie *LilyPond*.

REPERTUAR MUZYKI POLSKIEJ WIEKU XIX – WYDANIA CYFROWE I ANALIZA SKOMPUTERYZOWANA

Flauto I

Flauto II

Clarinetto I  
in C

Clarinetto II  
in C

Fagotto I

Fagotto II

Corno I  
in C

Corno II  
in C

Clarino I  
in C

Clarino II  
in C

Trombone Basso

Timpani  
in C G

Soprano  
Ky-ri-e Ky-ri-e Ky-ri-e e-lei-son Ky-ri-e e-lei-

Alto  
Ky-ri-e Ky-ri-e Ky-ri-e e-lei-son Ky-ri-e e-lei-

Tenore  
Ky-ri-e Ky-ri-e Ky-ri-e e-lei-son e-lei-

Basso  
Ky-ri-e e-lei-son Ky-ri-e e-lei-son e-lei-

Violino I  
*pp*

Violino II  
*pp*

Viola

Organo  
*p*

## 5. Wnioski

W świetle przedstawionej argumentacji wydaje się oczywiste, że korzyści płynące ze stworzenia dużej bazy danych, obejmującej repertuar muzyki polskiej wieku XIX (a docelowo wykraczającej poza te ramy czasowe), są niewątpliwe. Możliwości, jakie daje współczesna technologia, pozwalają prowadzić badania, których przeprowadzenie było do tej pory nierealne. Szczególnie obiecującym tego przykładem jest baza JRP tworzona przez stanfordzkich naukowców. Kończąc nasze rozważania, warto wskazać na kilka postulatów, których spełnienie umożliwiłoby opracowanie analogicznej do JRP bazy obejmującej repertuar muzyki polskiej.

Niezwykle ważne jest instytucjonalne wspieranie rozwoju narzędzi takich jak *Humdrum* czy *Verovio*. Należy pamiętać, że oprogramowanie typu *Open Source* zwykle budowane jest społecznościowo, tempo prac rozwojowych zależy zaś od możliwości czasowych i finansowych osób mających specjalistyczne kompetencje. Właśnie dlatego inicjatywa podjęta przez NIFC wydaje się tak ważna – w Instytucie udało się doprowadzić do współpracy czołowych programistów reprezentujących najlepsze ośrodki badawcze, dzięki czemu bardziej realne stają się perspektywy ewolucji wspomnianego oprogramowania. Jednak narzędzia to nie wszystko – najważniejsze są dane muzyczne, których pozyskanie jest niezwykle pracochłonne i czasochłonne. Dlatego wydaje się, że pożądanym rozwiązaniem byłoby nawiązanie szerszej współpracy z polskimi instytutami muzykologii oraz uczelniami muzycznymi. To właśnie w tych ośrodkach w ramach prac prowadzonych przez studentów tworzy się ogromne ilości danych, które mogłyby stać się podstawą dla postulowanej bazy muzyki polskiej.

---

**ABSTRACT**

*Polish music of the nineteenth century: digital editions and computer-aided analysis*

This article represents the author's reflection on available formats for the notation of a score, methodologies for computer-aided analysis and the possible ways of rendering scores of nineteenth-century Polish music. The author will also present sample transcriptions and outline the basic capacities of computer-aided analysis. There are undeniable benefits resulting from the creation of a large data base of nineteenth-century Polish music (the ultimate aim is to go beyond that timeframe). This concerns not only critical style analysis, but also research into the attribution of anonymous works. The possibilities offered by modern technology enable us to conduct research that was not previously feasible. One promising example of this is the data base of the Josquin Research Project, created by Stanford academics.

---

**ABTRAKT**

Artykuł niniejszy stanowi próbę refleksji nad dostępnymi formatami zapisu, metodologiami analizy skomputeryzowanej oraz możliwymi sposobami renderowania partytur repertuaru muzyki polskiej XIX stulecia. W tekście zaprezentowano też przykładowe transkrypcje oraz podstawowe możliwości analizy skomputeryzowanej. Korzyści płynące ze stworzenia dużej bazy danych, obejmującej repertuar muzyki polskiej wieku XIX (a docelowo wykraczającej poza te ramy czasowe) są niekwestionowane. Dotyczy to nie tylko przeprowadzania analizy stylokrytycznej, ale także badań w zakresie atrybucji utworów anonimowych. Możliwości, jakie daje współczesna technologia, pozwalają prowadzić badania, których podjęcie było do tej pory nierealne. Szczególnie obiecującym tego przykładem jest baza Josquin Research Project tworzona przez stanfordzkich naukowców.

---

**SŁOWA KLUCZOWE**

skomputeryzowana analiza muzyczna, humdrum, verovio, muzykologia cyfrowa, digital humanities, music encoding initiative

---

DR MARCIN KONIK

Absolwent filozofii i muzykologii, kierownik Biblioteki Narodowego Instytutu Fryderyka Chopina w Warszawie. Doktorat uzyskał na podstawie rozprawy poświęconej wątkom kosmologicznym w średniowiecznych traktatach z zakresu teorii muzyki. Odbił staże w prestiżowych ośrodkach naukowych: Jacob's School of Music na Indiana University w Bloomington oraz na Wydziale Muzyki Uniwersytetu Stanforda. W swoich badaniach skupia się na problematyce skomputeryzowanej analizy muzycznej, muzycznych bazach danych, projektach digitalizacyjnych. Specjalizuje się w badaniach źródłoznawczych muzyki polskiej okresu XVII–XIX wieku.