

# The Humdrum Toolkit

David Huron

Presented by Yu Fang

# What is Humdrum?

A set of inter-related command-line tools designed to assist *computational* music analysis.

Originally designed by David Huron in the '80s, and programmed by Tim Racinsky and Kyle Dawkins. Now extended and maintained by Craig Sapp.

Craig Sapp also ported a Humdrum viewer and editor to the web. See [Verovio Humdrum Viewer](#).

# What is Humdrum?

Consists of

- ▶ File syntax: tab-separated spreadsheet.
- ▶ Representations: `**kern`, `**pitch`, `**deg`, `**MIDI`, `**freq`, etc.
- ▶ Tools: analyze, manipulate, translate between representations.
  - ▶ Mainly written in AWK and C, but are language agnostic
  - ▶ Heavily influenced by the UNIX philosophy (modularity)

# Background: UNIX Pipes

In: 1 \$ echo 'abcd'

Out: ⇒ abcd

In: 1 \$ echo 'abcd' | sed 's/a/A/'

Out: ⇒ Abcd

Tab-separated spreadsheet.

- ▶ Vertical: sequential data
- ▶ Horizontal: parallel data
- ▶ Column: representation of data, can be pitch, duration, metadata, etc.

```
!!!OTL: Title
!! global comments
**note **pitch **duration
1      C4      quarter
2      B3      eighth
3      G4      eighth
!      !      ! local comments
4      F4      eighth
5      G4      eighth
6      A4      quarter
7      G4      eighth
8      Ab4     quarter
*-     *-     *-
```

Data representation is arbitrary, but there are several standard representations supported by the toolkit.

`**pitch` is one of the standard representations.

<code>**pitch</code>	<code>**heart</code>
C4	55
B3	56
G4	55
F4	57
G4	54
A4	58
G4	56
Ab4	56
*-	*-

# Command: extract

To get rid of metadata, a specific column can be selected using the `extract` command.

In: 1 `$ extract -i '**pitch' file`

Out: ⇒ `**pitch`  
`C4`  
`B3`  
`G4`  
`F4`  
`G4`  
`A4`  
`G4`  
`Ab4`  
`*-`

<code>**pitch</code>	<code>**heart</code>
<code>C4</code>	<code>55</code>
<code>B3</code>	<code>56</code>
<code>G4</code>	<code>55</code>
<code>F4</code>	<code>57</code>
<code>G4</code>	<code>54</code>
<code>A4</code>	<code>58</code>
<code>G4</code>	<code>56</code>
<code>Ab4</code>	<code>56</code>
<code>*-</code>	<code>*-</code>

# Command: mint

The output can be piped into other commands for further processing.

The `mint` command, for example, can convert `**pitch` into `**mint` representation, which describes melodic intervals.

```
In: 1 $ extract -i '**pitch' file | mint
```

```
Out: => **mint  
      [C4]  
      -m2  
      +m6  
      -M2  
      +M2  
      +M2  
      -M2  
      +m2  
      *-
```

**pitch	**heart
C4	55
B3	56
G4	55
F4	57
G4	54
A4	58
G4	56
Ab4	56
*-	*-



# Command: pitch

It can also be converted to frequency representation using the `freq` command.

```
In: 1 $ extract -i '**pitch' file | freq
```

```
Out: => **freq  
261.63  
246.94  
392.00  
349.23  
392.00  
440.00  
392.00  
415.30  
*-
```

```
**pitch **heart  
C4      55  
B3      56  
G4      55  
F4      57  
G4      54  
A4      58  
G4      56  
Ab4     56  
*-      *-
```

# Command: trans

The trans command can be used to transpose pitches.

```
In: 1 $ extract -i '**pitch' file | trans -d1 -c3
```

```
Out: => **pitch  
      *Trd1c3  
      D#4  
      C##4  
      A#4  
      G#4  
      A#4  
      B#4  
      A#4  
      B4  
      *-
```

Augmented second (1 diatonic, 3 chromatic)

# \*\*kern Representation

The core pitch/duration representation of Humdrum.

**kern	**kern
*clefF4	*clefG2
*k[f#c#]	*k[f#c#]
*d:	*d:
*M4/4	*M4/4
=1	=1
1D	4d
.	4f#
.	4a
.	4dd
=2	=2
1AA	4ee
.	4cc#
.	4g
.	4b
=3	=3
1D;	1a;
==	==
*-	*-



# \*\*kern Representation

In: 1 \$ mint score.krn

Out: ⇒ \*\*mint \*\*mint  
\*clefF4 \*clefG2  
\*k[f#c#] \*k[f#c#]  
\*d: \*d:  
\*M4/4 \*M4/4  
=1 =1  
[D] [d]  
. +M3  
. +m3  
. +P4  
=2 =2  
-P4 +M2  
. -m3  
. -A4  
. +M3  
=3 =3  
+P4 -M2  
== ==  
\*- \*

```
**kern **kern
*clefF4 *clefG2
*k[f#c#] *k[f#c#]
*d: *d:
*M4/4 *M4/4
=1 =1
1D 4d
. 4f#
. 4a
. 4dd
=2 =2
1AA 4ee
. 4cc#
. 4g
. 4b
=3 =3
1D; 1a;
== ==
*- *
```

KernScores: a large library of musical scores encoded in `**kern` format (Sapp [2005](#)).

“Meme hunting” using the `patt` and `pattern` command (Jan [2004](#), [2017](#)).

- Huron, David. 1999. *Music Research Using Humdrum: A User's Guide*.
- Huron, David. 2002. "Music Information Processing Using the Humdrum Toolkit: Concepts, Examples, and Lessons." In *Computer Music Journal*, 26:11–26. The MIT Press.
- Jan, Steven. 2004. "Meme Hunting with the Humdrum Toolkit: Principles, Problems, and Prospects." In *Computer Music Journal*, 28:68–84. The MIT Press.
- Jan, Steven. 2017. *The Memetics of Music: A Neo-Darwinian View of Musical Structure and Culture*. Routledge.
- Sapp, Craig Stuart. 2005. "Online Database of Scores in the Humdrum File Format." In *Proceedings of the International Society for Music Information Retrieval*, 664–665.