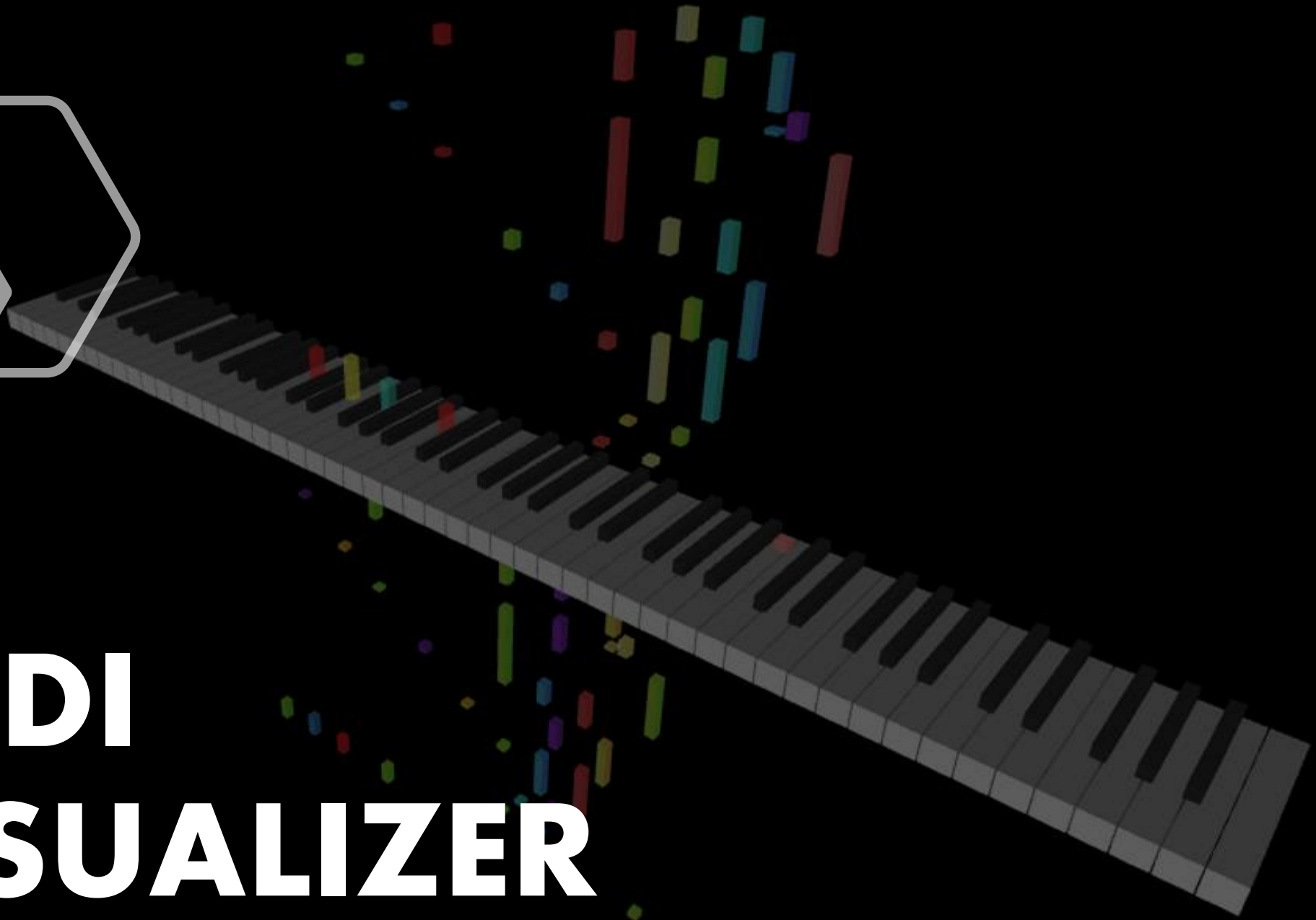
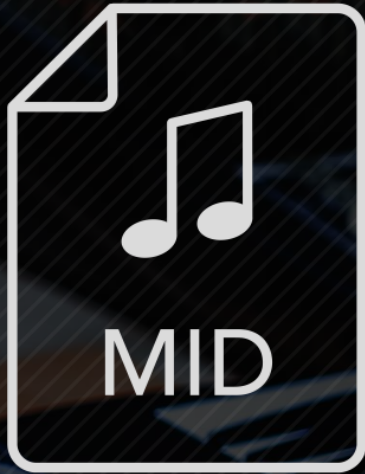




# MIDI VISUALIZER



# MIDI FILE FORMAT SPECIFICATION



---

여러 개의 MIDI Stream

시간에 따라 이벤트를 저장

시퀀스, 트랙 구조, 템포,  
박자표, 음악 등의 정보를 저장

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# MIDI FILE FORMAT SPECIFICATION

## VLQ

Variable Length Quantity

---

가변적인 숫자를 저장하기 위한  
기법  
7비트 단위로 분리하여  
바이트별로 나누어 저장한다.

**MSB가 1**

다음 바이트를 이어서 읽는다.

**MSB가 0**

숫자의 끝을 나타낸다.

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# MIDI FILE FORMAT SPECIFICATION

0xC004

1100 0000 0000 0100

VLQ

1 | 100 0000 | 0 | 000 0100 |

이어짐

끝남

0x2004

0010 0000 0000 0100

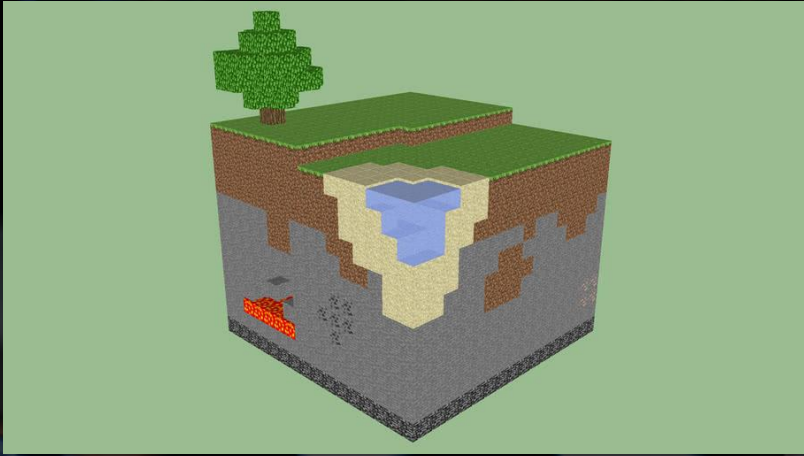
Value

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발



# MIDI FILE FORMAT SPECIFICATION



## Chunk

---

Chunk는 MIDI 파일의 가장  
기본적인 단위이다.

파일 전체에 1개 있는

**Header Chunk**

와, 나머지를 구성하는

**Track Chunk**

로 나누어진다.

# MIDI FILE FORMAT SPECIFICATION

## Chunk

---

Chunk Type (2byte, BE)

Chunk의 종류

Length (2byte, BE)

Chunk 데이터 길이

Data

Chunk 데이터

# MIDI FILE FORMAT SPECIFICATION

## Header Chunk

---

Chunk Type : 'MThd'

Chunk Data :

<Format>

<nTrks>

<Division>



# MIDI FILE FORMAT SPECIFICATION

## Header Chunk Format

- 
- 0 | 파일이 하나의 multi-channel track을 가진다.
  - 1 | 파일이 하나 이상의 simultaneous track을 가진다.
  - 2 | 파일이 하나 이상의 sequentially independent single-track 패턴을 가진다.



# MIDI FILE FORMAT SPECIFICATION

## Header Chunk nTrks

---

nTrks는 이후로 등장하는 Track  
Chunk의 개수이다.

만약 Format의 값이 0이라면  
Ntrks의 값은 항상 1이 된다.

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# MIDI FILE FORMAT SPECIFICATION

## Header Chunk Division

Bit 15	Bits 14 ~ 8	Bits 7 ~ 0
0	4분음표 하나 당 tick 개수	
1	Negative SMPTE	프레임 당 tick

SMPTE 영화 · 텔레비전 기술자 협회

(Society of Motion Picture and Television Engineers, SMPTE)

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# MIDI FILE FORMAT SPECIFICATION

## Track Chunk

---

Chunk Type : 'MTrk'

Chunk Data :

<MTrk Event>+

MTrk Event :

<Delta Time>

<Event>



# MIDI FILE FORMAT SPECIFICATION

## Delta Time

---

Delta Time은 VLQ

이전 이벤트가 시작된 후로부터  
해당 이벤트까지의 시간(tick)

두 이벤트가 동시에 실행된다면  
Delta Time은 0

# MIDI FILE FORMAT SPECIFICATION

## Event

---

Event :

<MIDI Event> or

<Sysex Event> or

<Meta Event>

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# MIDI FILE FORMAT SPECIFICATION

## MIDI Event

MIDI Event는 음원, 특정 악기의 재생,  
가상악기 모듈레이션의 변화 등을 모두  
저장할 수 있다.

중요한 MIDI Event

Status	Data	Desc
1000nnnn	0kkkkkkk 0vvvvvvv	Note off
1001nnnn	0kkkkkkk 0vvvvvvv	Note on

k: key num    v: velocity

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발



# MIDI FILE FORMAT SPECIFICATION

## Note Number

Oct	Note Num											
	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
-1	0	1	2	3	4	5	6	7	8	9	10	11
0	12	13	14	15	16	17	18	19	20	21	22	23
1	24	25	26	27	28	29	30	31	32	33	34	35
2	36	37	38	39	40	41	42	43	44	45	46	47
3	48	49	50	51	52	53	54	55	56	57	58	59
4	60	61	62	63	64	65	66	67	68	69	70	71
5	72	73	74	75	76	77	78	79	80	81	82	83
6	84	85	86	87	88	89	90	91	92	93	94	95
7	96	97	98	99	100	101	102	103	104	105	106	107
8	108	109	110	111	112	113	114	115	116	117	118	119
9	120	121	122	123	124	125	126	127				

UGRP

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# MIDI FILE FORMAT SPECIFICATION

## Meta Event

---

주요한 Meta Event

**FF 51 03 tttttt Set tempo**

템포를 변경하는 이벤트

**FF 08 04 nn dd cc bb Time**

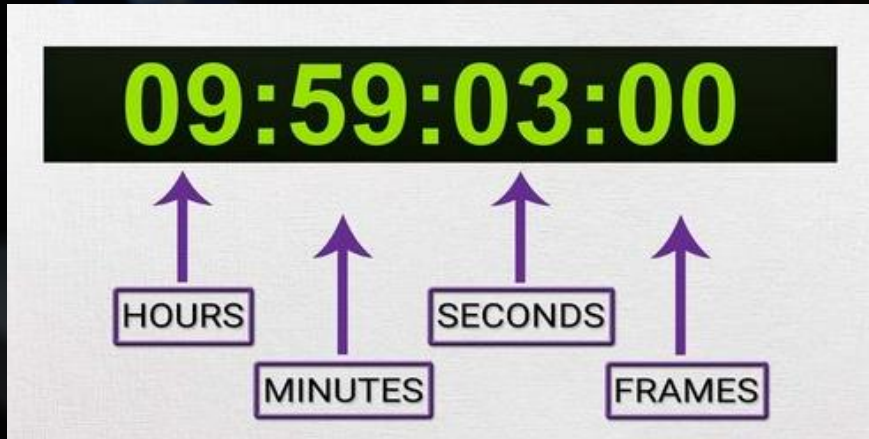
**Signature**

박자표를 변화시키는 이벤트

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# MIDI FILE FORMAT SPECIFICATION



**FF5103**  
**SetTempo**

FF 51 03 tt tt tt

ttttttt라고 쓰인 자리(24bit)에 변화할  
템포를 넣는다.

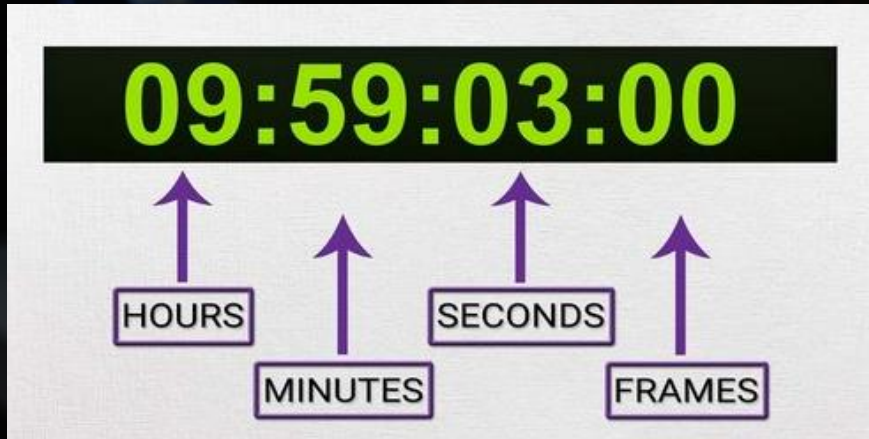
여기에 4분음표의 길이를  
마이크로초로 계산하거나 (최대 16.77s)  
SMPTE Time Code로 표시할 수 있다.

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발



# MIDI FILE FORMAT SPECIFICATION



**FF0804**  
**Time signature**

FF 08 04 nn dd cc bb

nn : 박자표의 분자

dd : 박자표의  $\log_2(\text{분모})$

cc : 4분음표를 24로 두었을 때  
메트로놈이

몇 clock에 한 번 울릴지 설정 (\*)

bb : 32분음표 몇 개를 4분음표로 둘 지  
설정 (\*\*)

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# MIDI FILE FORMAT SPECIFICATION

Meta event:  
Status: 0x51  
Data: 0x0a2c2b

Note:  
ID: 56  
start: 168647  
end: 168874  
duration: 227

Note:  
ID: 58  
start: 168887  
end: 169114  
duration: 227

File format: 1  
Number of tracks: 2  
Time division: 480

0x59 0x03 0x0a2c2b :

0.666667초 / 4분음표

Time Division :

480tick / 4분음표

Note : 227tick + 13tick(rest)

실제 길이 :  $240 * 0.666667 / 480 =$

0.333s

BPM : 60bpm

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# JSON

---

JSON (JavaScript Object Notation)

경량의 데이터 교환 형식

사람이 읽고 쓰기 용이함

언어에 완전히 독립적

Name-Value pair

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발



# JSON

```
{
  "orders": [
    {
      "orderno": "748745375",
      "date": "June 30, 2088 1:54:23 AM",
      "trackingno": "TN0039291",
      "custid": "11045",
      "customer": [
        {
          "custid": "11045",
          "fname": "Sue",
          "lname": "Hatfield",
          "address": "1409 Silver Street",
          "city": "Ashland",
          "state": "NE",
          "zip": "68003"
        }
      ]
    }
  ]
}
```

JSON (JavaScript Object Notation)

경량의 데이터 교환 형식

사람이 읽고 쓰기 용이함

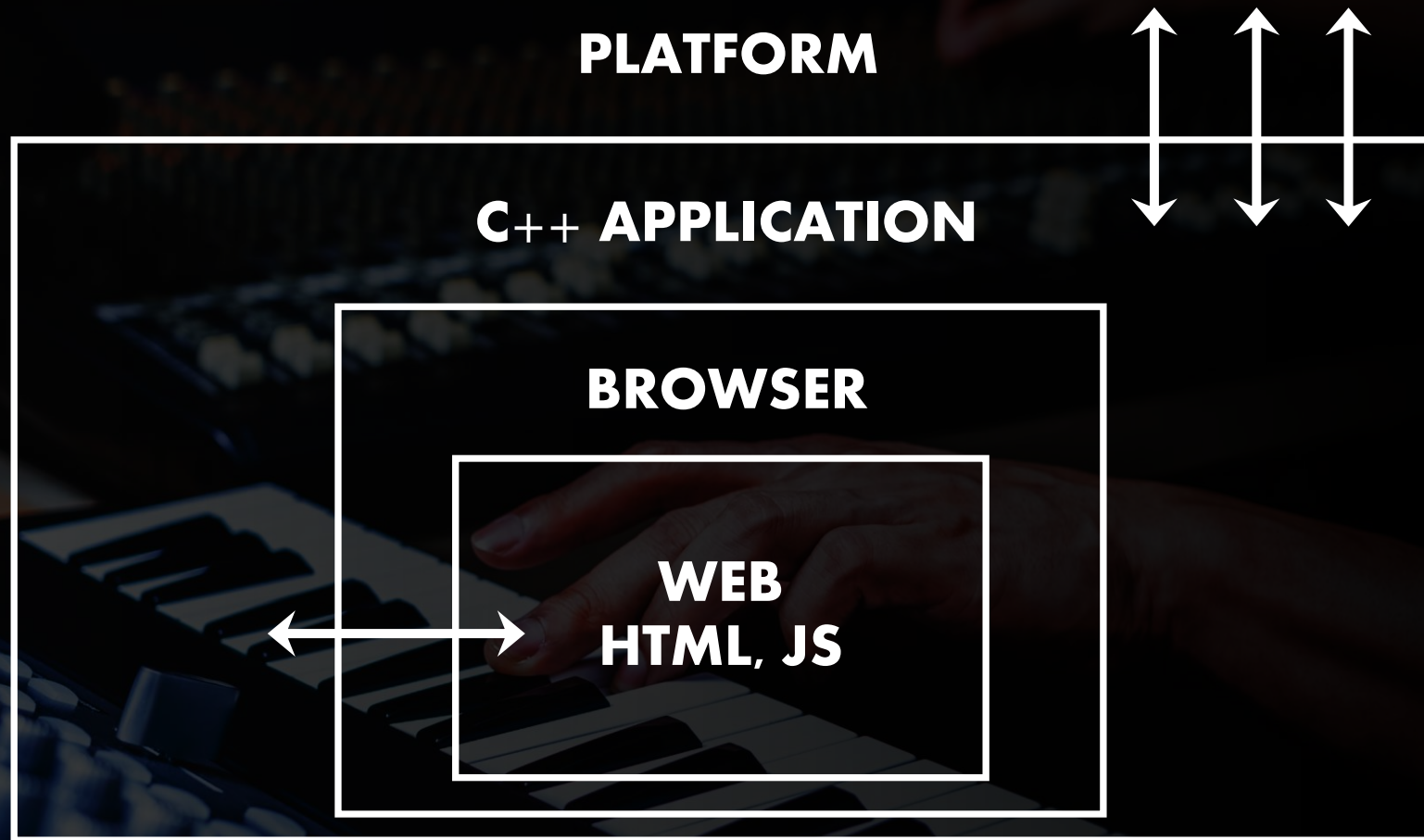
언어에 완전히 독립적

Name-Value pair

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

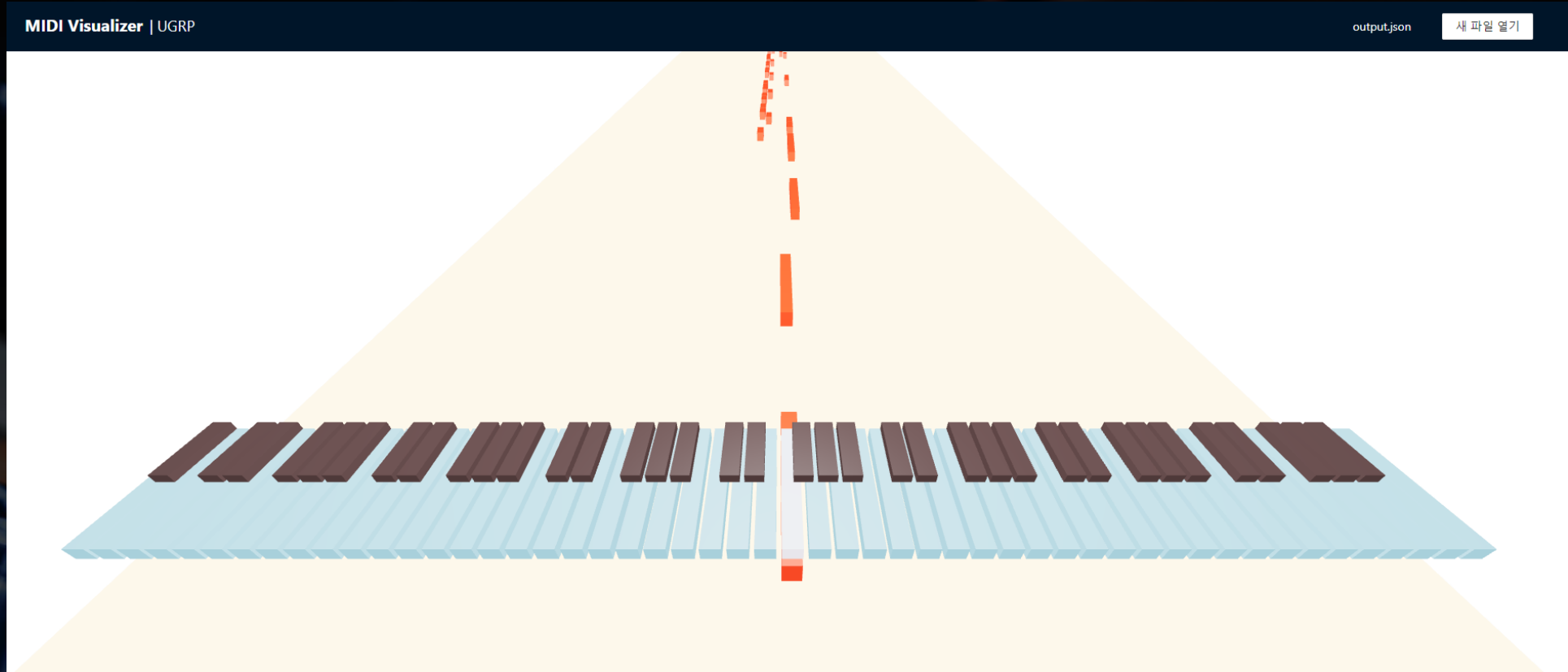
# APPLICATION STRUCTURE



**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# DEMONSTRATION



**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발





# **GUIDO**

# **LIBRARY**

# GUIDO LIBRARY

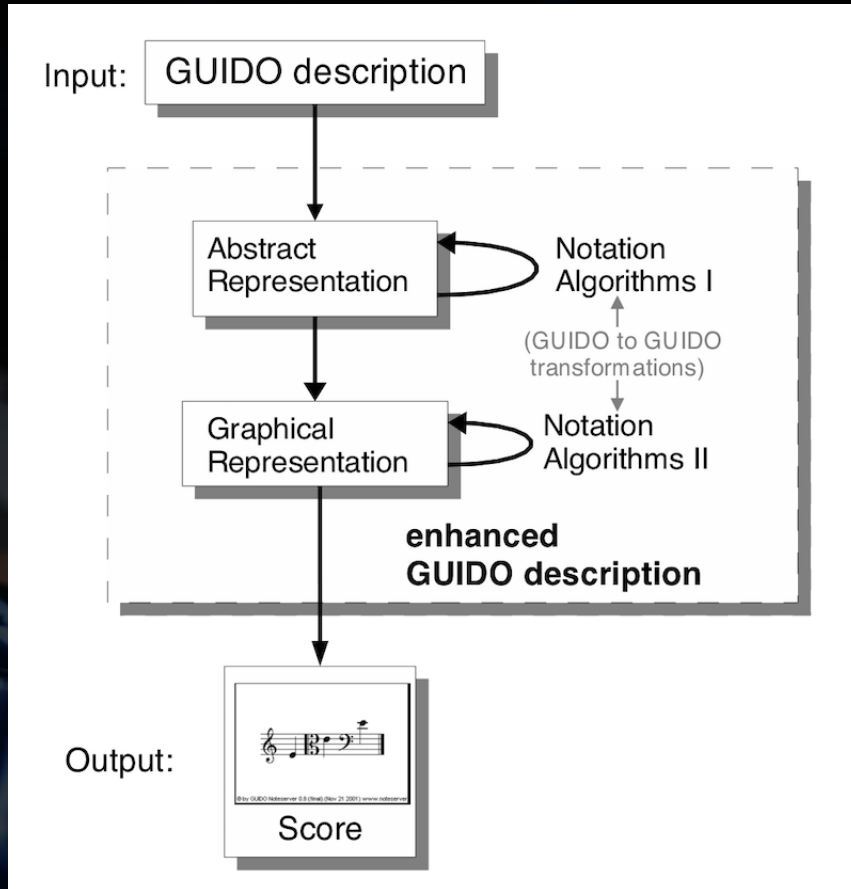
## The Guido Project



**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# GUIDO LIBRARY



## GUIDO Library Structure

**AR**(Abstract Representation)

**GR**(Graphical Representation)

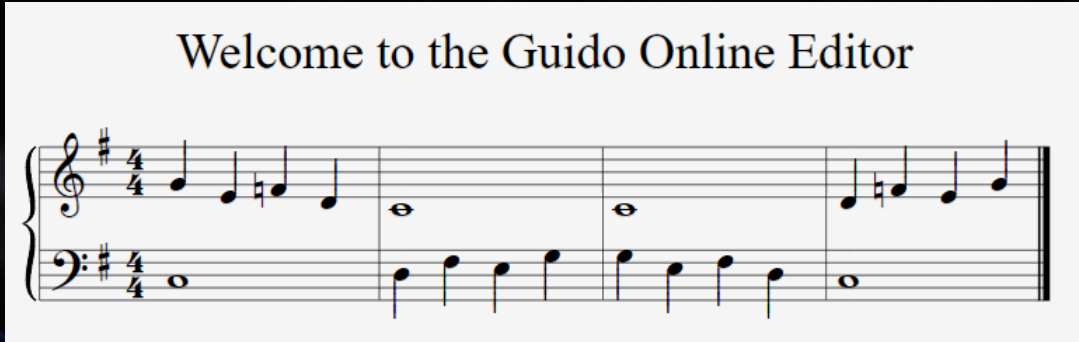
**SVG**(Scalable Vector Graphics)

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발



# GUIDO LIBRARY



# GMN

## UGRP

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

## Guido Music Notation

```
(*  
  Welcome to the Guido online editor  
*)  
{  
  [  
    \title<"Welcome to the Guido Online Editor">  
    \barFormat<"system">  
    \clef<"g"> \key<1> \meter<"4/4"> g e f d c/1 c d/4 f e g  
  ],  
  [  
    \clef<"f"> \key<1> \meter<"4/4"> c0/1 d/4 f# e g g e f# d c/1  
  ]  
}
```

Try it online :

<https://guidoeditor.game.fr>

<https://guidodoc.game.fr/>

## Note names

There are different systems of note names:

- diatonic: `c d e f g a h/b`
- chromatic: `c cis d dis e f fis g gis a ais h/b`
- solfege: `do re me fa sol la si/ti`

## Accidentals

Accidentals is an arbitrary sequence of the symbols `#` and `&` (for sharp and flat); in particular,

`##` is a double sharp, `&&` a double flat.



# GUIDO LIBRARY

## Examples

<code>{c1/4., e&amp;1/4., g1/4.}</code>	<code>% c minor triad , ascending order</code>
<code>{c1/4., e&amp;,g}</code>	<code>% same as above, abbreviated notation</code>
<code>{c#1/4, e#, g#}</code>	<code>% C-sharp major triad, ascending order</code>
<code>{c2, dis1, a0, fis}</code>	<code>% four-voice chord, descending order</code>



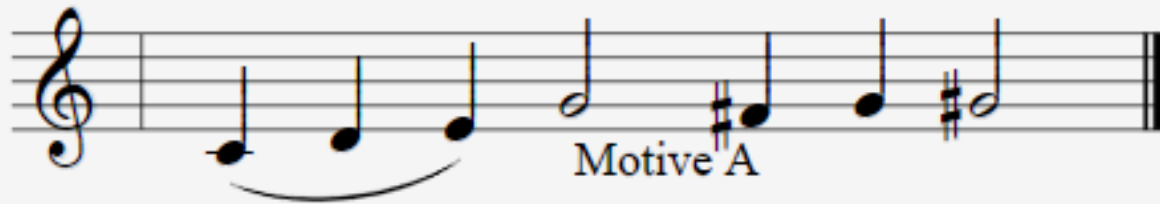
UGRP

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# GUIDO LIBRARY

## Examples

```
\bar                                % barline
\clef<"g2">                        % treble clef
\slur(c1/4 d e)                    % slurred group
\text<"Motive A">(g1/2 f#/4 g g#/2) % text
```



UGRP

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# GUIDO LIBRARY



같은 음이지만 첫 번째 음표에는  
임시표가 붙고, 두 번째 음표에는  
임시표가 붙지 않음.

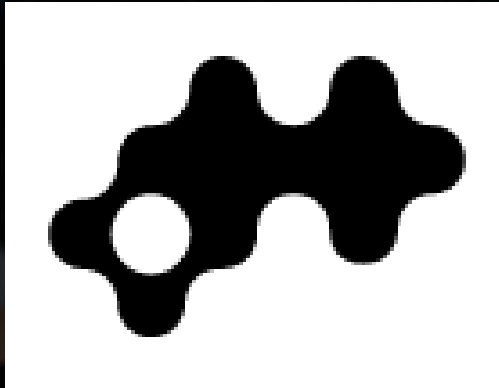
추상화된 데이터를  
실제 그려질 데이터로 변환 필요

## UGRP

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발



# LIBRARY INTERNAL



---

NODE MODULE

WASM

EMSCRIPTEN

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# LIBRARY INTERNAL



---

**Node.js**

자바스크립트 실행 환경

**Node Module**

**NPM**(Node Package Manager)

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# LIBRARY INTERNAL



WA

---

## WASM(Web Assembly)

최신 브라우저에서 작동하는 언어  
C++, RUST 등의 저급 언어를  
컴파일 가능

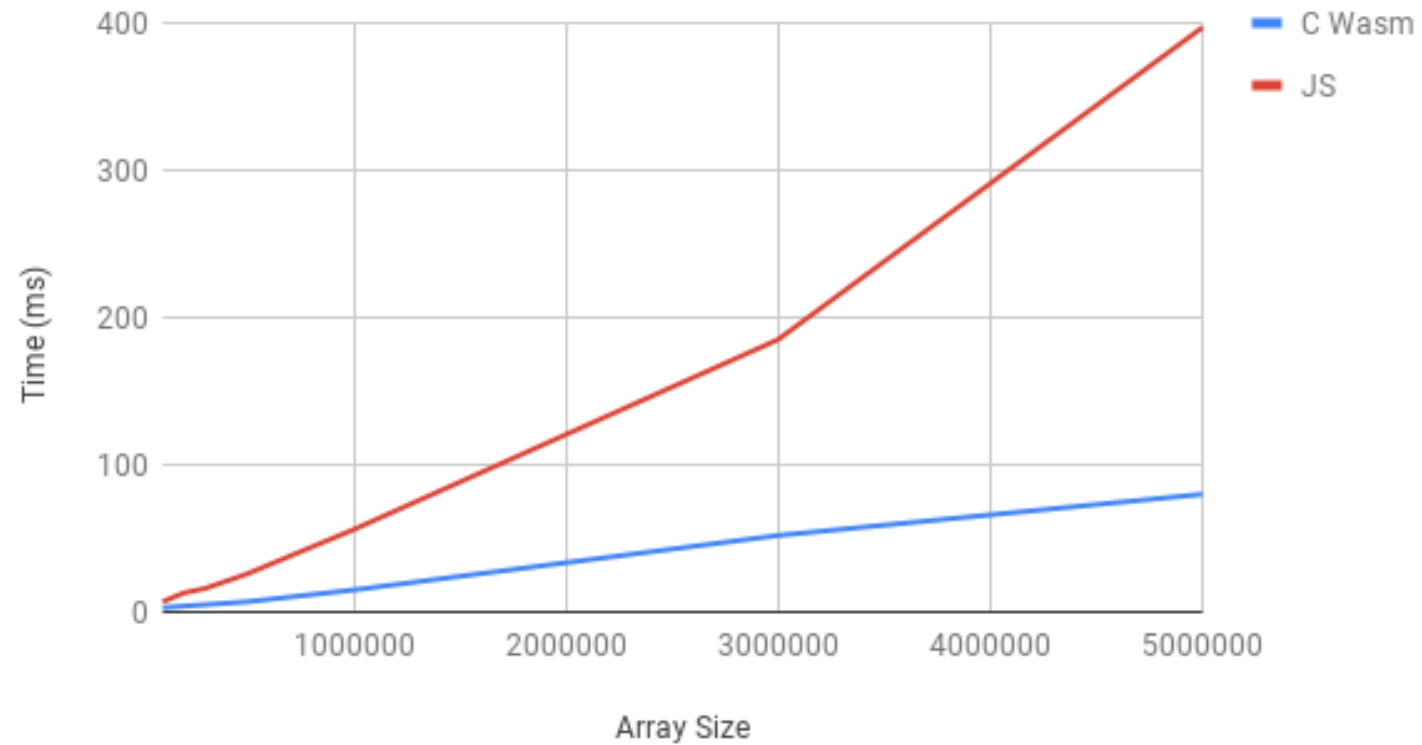
### UGRP

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발



# LIBRARY INTERNAL

Uint8\_t Array Reverse



는 언어  
어를

## UGRP

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발

# LIBRARY INTERNAL



---

Emscripten

자바스크립트에서 WASM을  
사용하기 위한 소스 대 소스  
컴파일러

**UGRP**

| 음원의 악기별 분리 및 음원과 악보 간의 유사도 계산 기술 개발