AlgoRhythm

A Library for Algorithmic Music Composition

Joris ten Tusscher, Cas van der Rest, Orestis Melkonian

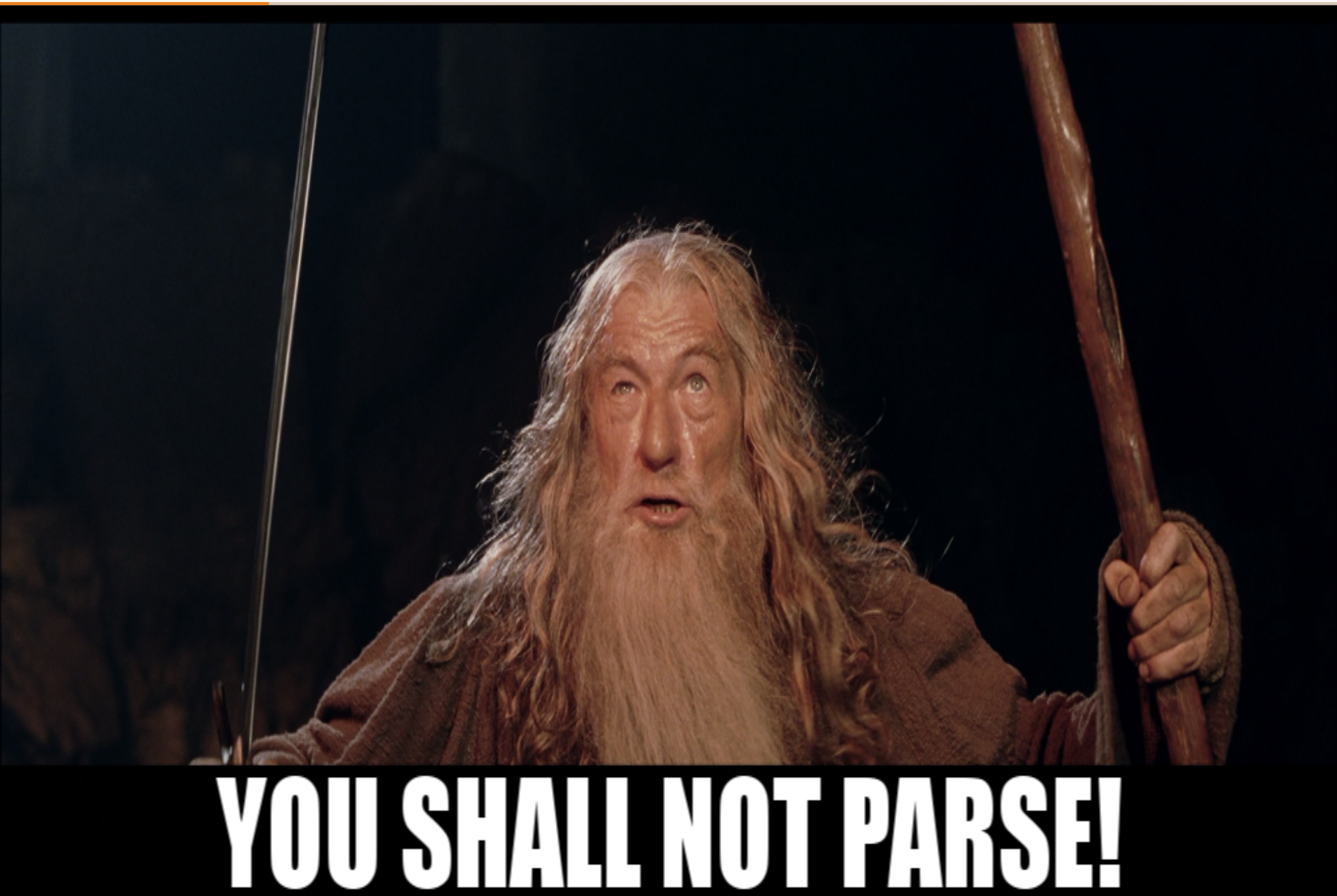
April 5, 2018

Universiteit Utrecht

Music DSL

music representation (Music, MusicCore, Scale, Chord, etc...
music manipulation (transpose, retrograde, time-scale, etc...

Focus on Generation, Ignore Analysis



Generation

genState, selectors, diatonic improv, etc...

Dynamic Performance

k-means, etc...

Grammars

grammar description

Grammars: Tabla Rhythm

```
tabla :: Grammar () Syllable
tabla = S |:
  [ S |--> TE1 :-: XI
  , XI |--> TA7 :-: XD
  , XD |--> TA8
  , XG |--> TB2 :-: XA
  ...
  , TE4 |--> Ti :-: Rest :-: Dha :-: Ti
  , TC2 |--> Tr :-: Kt
  , TB3 |--> Dha :-: Tr :-: Kt
  , TD1 |--> Rest
  ...
  ]
instance ToMusicCore Syllable where
    ...
```

Grammars: Jazz Improvisation

```
melody :: Grammar () NT
melody = MQ |:
  [ -- Abstract Rhythm { MQ ~> Q }
    (MQ, 1, (== qn))      |-> Q%:qn
  , (MQ, 25, (> (hn^.)))  :-> \t -> Q%:hn :-: MQ%:(t - hn)
  ...
    -- Concrete Rhythm { Q ~> MN }
  , (Q, 47, (== wn)) |-> MN%:qn :-: Q%:hn :-: MN%:qn
  , (Q,  6, (== hn)) |-> HT%:(qn^^^) :-: HT%:(qn^^^) :-: HT%:(
  ...
  -- Abstract Melody { MN ~> N }
  , (MN, 1, (== wn)) |-> MN%:qn :-: MN%:qn :-: MN%:qn :-: MN%:
  , (MN,  1, (== qn)) |-> HT%:(en^^^) :-: HT%:(en^^^) :-: AT%:
    ...
  -- Concrete Melody { N ~> NT }
  , (N, 50, (== qn)) |-> ColorTone%:qn
  , (N, 45, (== qn)) |-> Rest%:qn
  , (N,  1, (== en)) |-> ApproachTone%:en
  ...
  ]

mkSolo :: Music SemiChord -> Music NT -> IO Melody
```

Grammars: Tonal Harmony

```
harmony :: Grammar Modulation Degree
harmony = I |:
  [ -- Turn-arounds
    (I, 8, (> wn)) :->
      \t -> Let (I%:t/2) (\x -> x :-: x)
  , (I, 6, (> hn) /\ (<= wn)) :->
      \t -> II%:t/4 :-: V%:t/4 :-: I%:t/2
  , (I, 2, (> hn) /\ (<= wn)) :->
      \t -> V%:t/2 :-: I%:t/2
  , (I, 2) -|| (<= wn)
    -- Modulations
  , (V, 5, (> hn)) :-> \t -> Modulation P5 $: I%:t
  , V -| 3
  , (II, 2, (> hn)) :-> \t -> Modulation M2 |$: I%:t
  , II -| 8
  ]

instance Expand Degree Modulation SemiChord where
    ...

voiceLead :: Music SemiChord -> IO (Music Chord)
```

Demo: Code

```
orientalAlgebras = do
  let ?harmonyConfig = HarmonyConfig
    { basePc   = A
    , baseOct  = Oct3
    , baseScale = arabian
    , chords   = equally allChords
    }
  let ?melodyConfig = defMelodyConfig
    { scales   = equally allScales
    , octaves = [(20, Oct4), (15, Oct5), (5, Oct6)]
    , colorWeight = 0
    , approachWeight = 10
    }
  let ?midiConfig = MIDIConfig (6%5) [Piano, Sitar, Tabla]
  let ?tablaBeat = sn

  (back, fore) <- integrate (12 * wn)
  rhythm <- runGrammar tabla (12 * wn) ()
  writeToMidiFile "out.mid" (dyn (back :=: fore :=: rhythm))
```

Demo: Music score