

# clj as art

@kn1kn1

Sapporo.clj

# Sapporo.clj

Sapporo.cljは偶数月に開催しています。

今月は6/25(土) 13:30-の開催です。

Sapporo.clj #11

みなさまのご参加をお待ちしております！

# tl;dr

- Overtone  
ClojureからSuperColliderを操ることができます
- Quil  
ClojureからProcessingを操ることができます

SuperColliderって何だろう？

Processingって何だろう？

# Agenda

- Me
- Clojure
- SuperCollider
- Overtone
- Processing
- Quil
- Demo

# 資料など

- 本日の資料はこちらにupしてあります
- 質問は時間が余ったらセミナー時間中にお答えするかもですが、基本的にはセミナー終了後、個別にお伺い致します m( )m

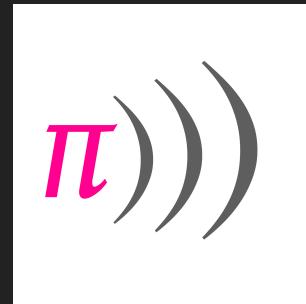
# About Me

- Kenichi Kanai
- a programmer in Sapporo
- Sapporo(3)<-Tokyo(15)<-Sapporo(6)<-Gumma(18)

※OSCは参加するのも発表するのも初めてです

- twitter: [@kn1kn1](#)
- github: [kn1kn1](#)
  - contributor of Sonic Pi
  - main committer of Haskap Jam Pack

# Sonic Pi



- <http://sonic-pi.net/>
- "The Live Coding Synth for Everyone"
- Sam Aaronが開発
- Rubyベース
- 当初はバグフィックスでcontribute
- SIAFラボに合流し、翻訳でcontribute
  - SIAF: 札幌国際芸術祭(Sapporo International Art Festival)

# Haskap Jam Pack



- <https://github.com/siaflab/haskap-jam-pack>
- "an extension package for Sonic Pi"
- Sonic Piでジャムセッション！
- SIAFラボで開発
  - 元々はアプリケーションとして札幌大谷大学講師の小町谷さんが開発されたものを、拡張パックとして@kn1kn1が再開発
- 技術的にはGoでパケットキャプチャしたりとかの話もあります → 別な機会にお話しします

# Clojure



- JVM上で動作するLISP系言語
- 2007～
- Rich Hickey

- code as data: 「データとしてのプログラムコード」
  - LISP系言語なので当 ( r y
- Javaの資産がほぼそのまま使える
- ソースコード(hello-clj.clj)で説明します

- 参考
  - プログラミングClojure 第2版
    - Clojureに関するほとんどのトピックスは網羅されている
    - それでも言及されていないものもある(->演算子(マクロ)とか)
  - Functional programming in Clojure
    - 関数っぽい話題で、"Little Schemer"的なノリ

# SuperCollider



- <https://superollider.github.io/>
- リアルタイム音響合成とアルゴリズミック・コンポジションに特化したプログラミング言語
- 1996年にJames McCartneyによって開発・発表

- SynthDefでシンセ(音源)を定義
- Synthで定義したシンセをトリガー
- Pbindでシンセ、 音程、 持続時間などを結びつけて音を鳴らす
- ソースコード(hello-sc.scd)で説明します

- 参考
  - The SuperCollider Book
    - Wilson, Scott, David Cottle, and Nick Collins. The SuperCollider Book. The MIT Press, 2011.
  - A Gentle Introduction To SuperCollider

# SuperColliderベースの音響合成環境

- hsc3
  - Haskell
- Tidal
  - Haskell
- Overtone
  - Clojure
- Sonic Pi
  - Ruby
- LNX\_Studio
  - これは言語ベースでなくDAW

# Overtone



- <https://overtone.github.io/>
- ClojureからSuperColliderの音響合成を行うライブラリ  
デイニング環境
- Jeff Rose, Sam Aaron等が開発

- 開発・実行環境
  - Emacs Live
    - Overtoneプロジェクトで開発している
    - EmacsからOvertoneを使うのに必要なClojure環境が揃う
    - ~/.emacs.dを書き換える(~/.emacs-live-old-configに退避する模様)ので注意が必要
  - LightTable
    - Quilでウィンドウが2つ出てしまったりするが、それ以外は大丈夫そう？

- 主な機能
  - SuperColliderのシンセサイザーエンジンに対するAPI
  - タイミングやメトロノームに関する関数群(at, apply-at等)
  - freesound.orgのサンプルを利用可能

- 最新版は2016年4月にリリースされた0.10.1
  - freesound.org API v2に対応するなど
  - sample関数に変更が入ったもののバグが入っている  
ようなので、ローカルでは0.9.1のコードにrevertしている→暇を見てIssueかPR出しておきます
  - Sam AaronがSonic Piに専念しているため、残念ながら開発はアクティブでない

- ソースコード(hello\_overline.clj)

- 参考
  - 公式のgetting\_started
  - Piotr Jagielski氏によるovertone-workshop
  - 公式のdocs/sc-book
    - The SuperCollider Book のコードをOvertoneに移植している

# Processing



- <https://processing.org/>
- Casey Reas, Benjamin Fryが開発
- ビジュアルデザインのためのプログラミング言語と統合開発環境

- Javaベース
  - 当初はJavaアプリケーションを生成する環境として
  - Androidアプリケーション(apk)も開発できようになり(2012年くらい)
  - processing.js, p5.js もよく使われている

- スケッチ
- setupで前処理
- drawで描画処理
- ソースコード(hello\_processing.pde)で説明します

- 参考
  - ジェネラティブ・アート -- Processingによる実践ガイド
    - <http://www.bnn.co.jp/books/3894/>
  - Processing -- ビジュアルデザイナーとアーティストのためのプログラミング入門
    - <http://www.bnn.co.jp/books/7770/>
  - Generative Design -- Processingで切り拓く、デザインの新たな地平
    - <http://www.bnn.co.jp/books/8012/>

# Quil



- <http://quil.info/>
- ClojureからProcessingを扱うことができる
- Roland Sadowski, Sam Aaron, Nikita Beloglazov等が開発
  - "I've started clj-processing project which later, with Sam Aaron's help, became Quil." -- <http://rosado.cc/>

- Clojure/ClojureScriptで実行可能
- Clojure版は、REPLで実行することにより動的にスケッチを変更することが可能  
→本家Processingには無い機能

- defsketch マクロ
- setup 関数
- update-state 関数
- draw-state 関数

# Hello Quil

```
$ lein new quil hello-quil
$ cd hello-quil
$ emacs -e cider-jack-in src/hello_quil/core.clj
```

- 参考
  - 公式のAPI
  - Generative Art - Quil Translations
    - ジェネラティブ・アート のコードをQuilに移植している

デモ