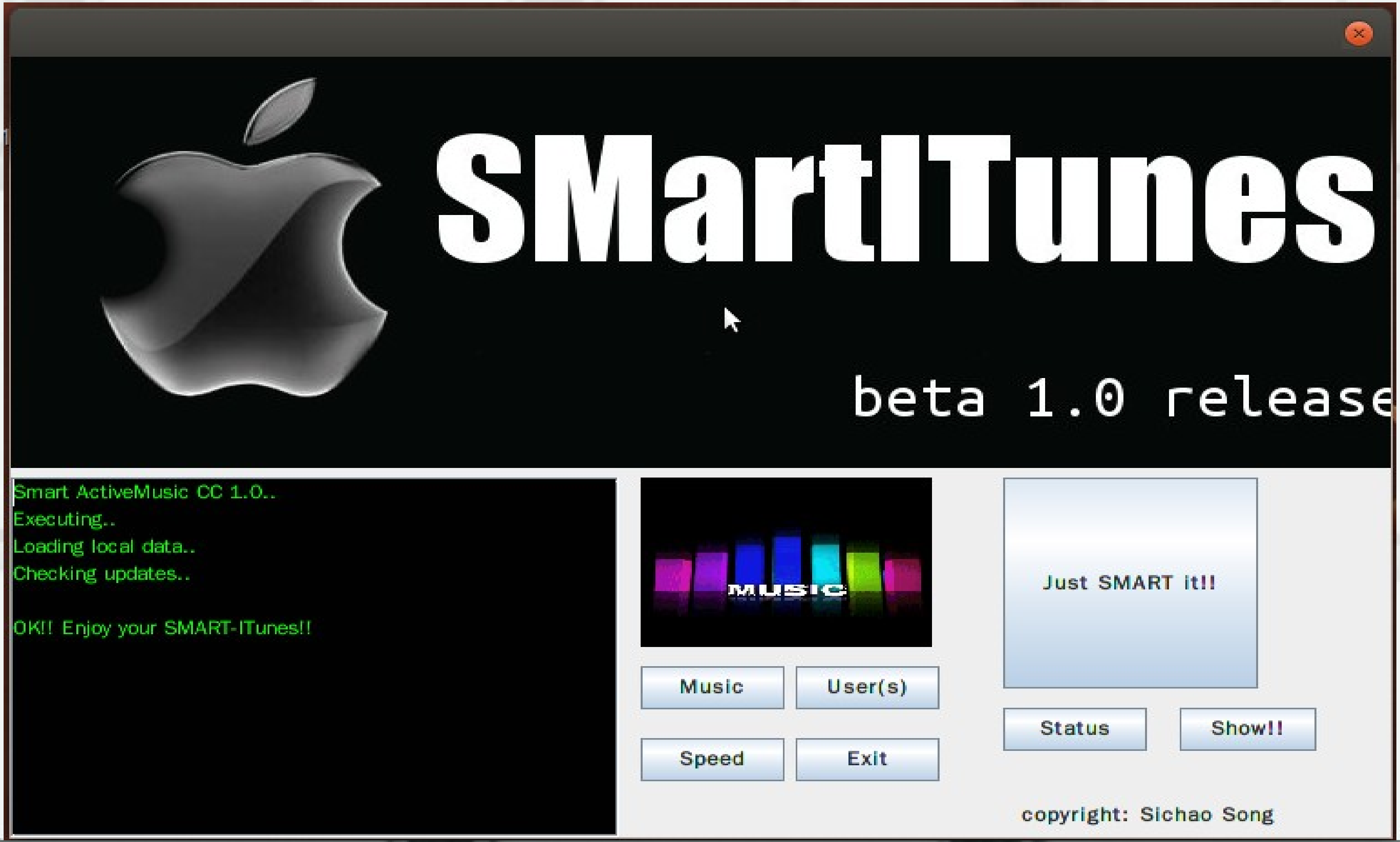
SmartMusic – smart your daily life!



```
Smart ActiveMusic CC 1.0..
Executing..
Loading local data..
Checking updates..

OK!! Enjoy your SMART-ITunes!!
```

Presented by: Sichao Song

Image this senario..

- If you are running while listening to some musics that stored in your mp3..
- When it comes some sad, slow music, you probably have to stop running and taking out your mp3 and chance some other music..

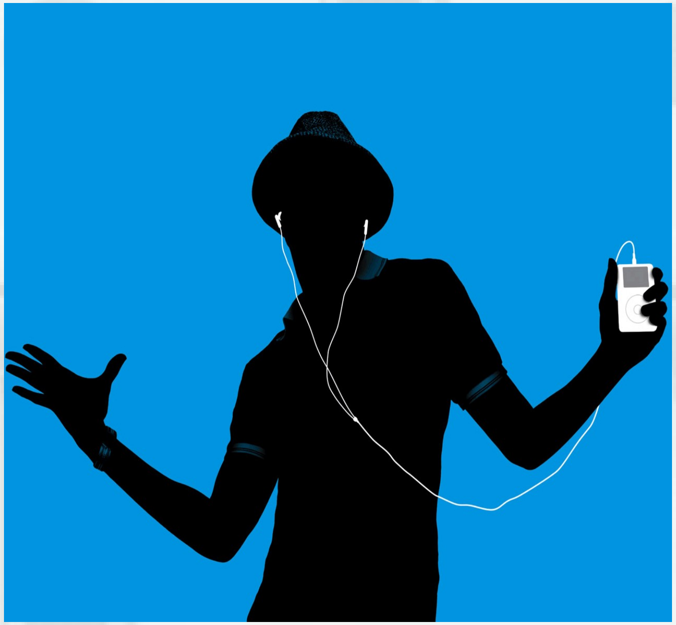
- It drives you CRAZY!!

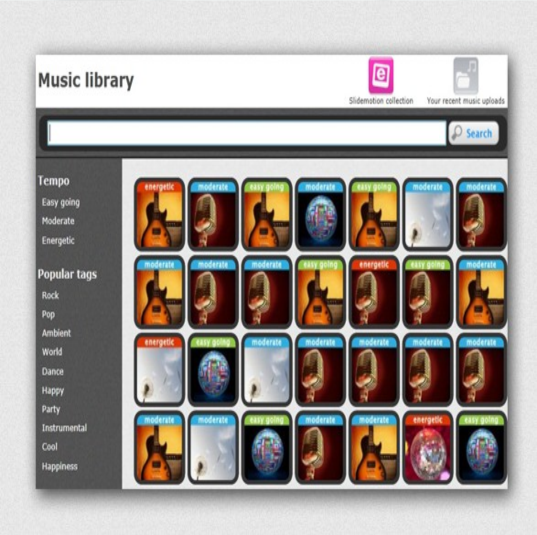
Don't worry!

- Now we have the SmartMusic Itunes beta1.0 release now

Senario

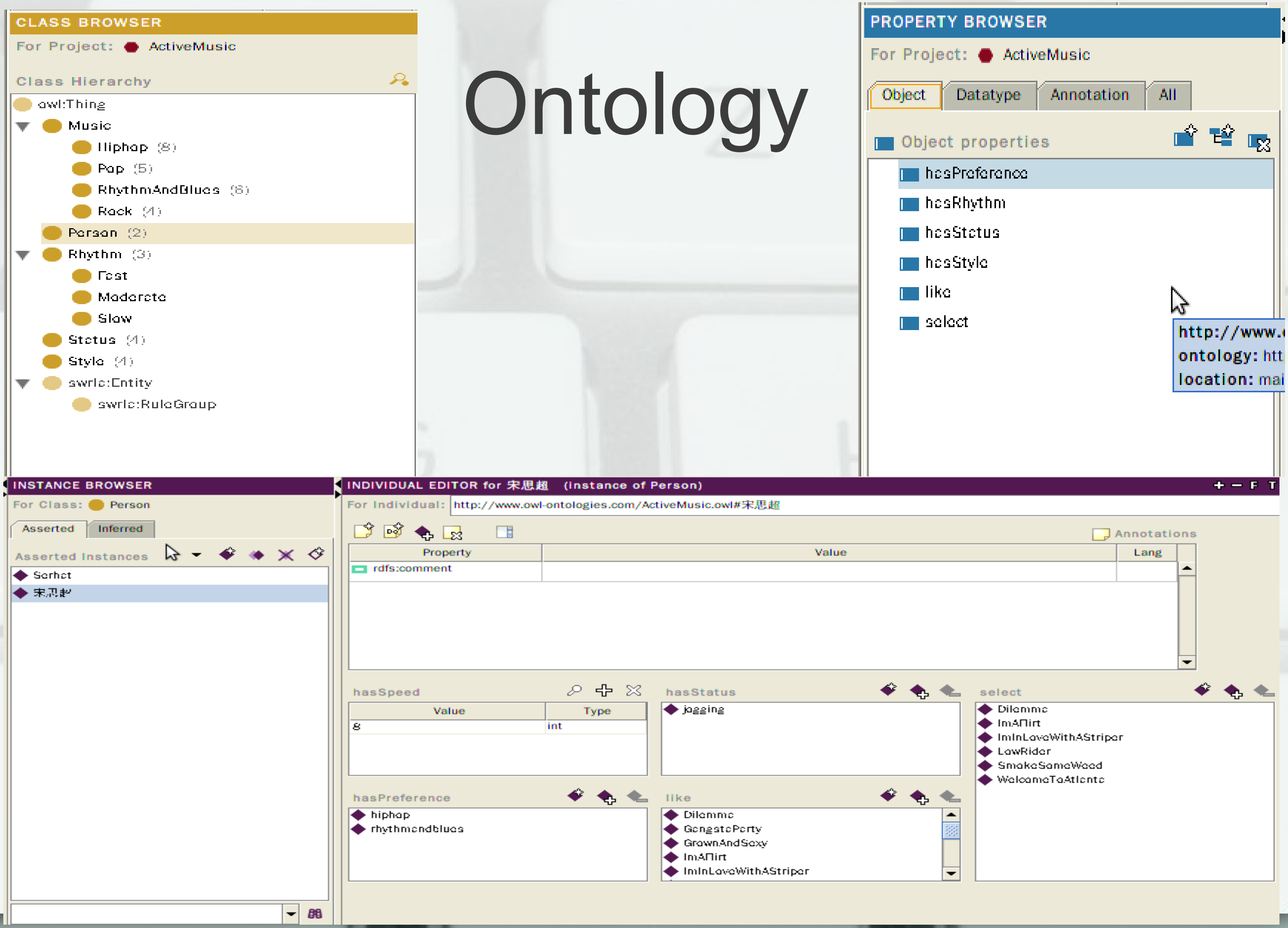
Running
Jogging
..





Semantic
Technology

Ontology



CLASS BROWSER

For Project: ● ActiveMusic

Class Hierarchy

- owl:Thing
- Music
  - Hiphop (8)
  - Pop (5)
  - RhythmAndBlues (8)
  - Rock (4)
- Person (2)
- Rhythm (3)
  - Fast
  - Moderate
  - Slow
- Status (4)
- Style (4)
- swrlc:Entity
  - swrlc:RuleGroup

PROPERTY BROWSER

For Project: ● ActiveMusic

Object | Datatype | Annotation | All

Object properties

- hasPreference
- hasRhythm
- hasStatus
- hasStyle
- like
- select

http://www.
ontology: htt
location: mai

INSTANCE BROWSER

For Class: ● Person

Asserted | Inferred

Asserted Instances

- Serhat
- 宋思超

INDIVIDUAL EDITOR for 宋思超 (instance of Person)

For Individual: http://www.owl-ontologies.com/ActiveMusic.owl#宋思超

Annotations

Property	Value	Lang
rdfs:comment		

hasSpeed

Value	Type
8	int

hasStatus

- jogging

select

- Dilemma
- ImAFlirt
- ImInLoveWithAStripper
- LowRider
- SmokeSomeWeed
- WelcomeToAtlanta

hasPreference

- hiphop
- rhythmandblues

like

- Dilemma
- GangstaParty
- GrownAndSexy
- ImAFlirt
- ImInLoveWithAStripper

SWRL rules

- For example..
  - Person(?p) ∧ hasSpeed(?p, ?s) ∧ swrlb:greaterThanOrEqual(?s, 10) → hasStatus(?p, running) --to determine whether a person is RUNNING
  - Person(?p) ∧ hasPreference(?p, ?s) ∧ Music(?m) ∧ hasStyle(?m, ?s) → like(?p, ?m) --which songs the person likes?
  - Person(?p) ∧ hasStatus(?p, sitting) ∧ Music(?m) ∧ hasRhythm(?m, slow) ∧ like(?p, ?m) → select(?p, ?m) --select songs for the person accordingly

Protege-OWL API

- In this project I choose to use Protege 3.4.8 and Protege-OWL API.
- More information about Protege-OWL API: (1) http://protege.stanford.edu/plugins/owl/api/
- (2) http://protegewiki.stanford.edu/wiki/ProtegeOWL_API_Programmers_Guide

So, why Protege-OWL API?

- I decided to choose Protege 3.4.8 instead of Protege 4.2:
  - I feel more comfortable with Protege 3.4.8, and it supports RDF. However, Protege 4.2 only supports OWL 2.0.
- Protege-OWL API is suitable with Protege 3.4.8, and supports Jena; OWL API is suitable with Protege 4.2.

What it can do for you?

- Intelligently selects musics that suit you (based on your preference and current status, or user's PROFILE)