

Under utvecklingen av programmet tänkte vi inte så mycket på hur vi skulle kunna använda oss utav konstruktioner specifika för dynamiska språk. Sådär i efterhand kan vi dock inte komma på några tillfällen då det kunde/borde använts. Vi stötte dock heller inte på några problem som vi tror hade kunnat lösas bättre i ett statiskt språk som t.ex. Java.

När det gäller dynamiska features som eventuellt hade kunnat användas fanns det t.ex. inget meningsfullt sätt att använda eval eller metaprogrammering. För det första eftersom programmet är relativt förutsägbart och därför kan vi inte se något behov av att dynamiskt skapa klasser eller metoder. För det andra eftersom det vi inte kan se att det hade varit användbart att t.ex. låta användaren skriva in namnet på en funktion som därefter anropas med hjälp av eval. Det fanns heller inget behov av att använda funktioner vars returvärde kan vara av olika typ.

Detta har lett till att programmet lika gärna hade kunnat gjorts på ett liknande sätt i t.ex. Java om det hade funnits ett "Javify" API.

Något som vi dock kom på hade kunnats göras annorlunda i vårt program var att istället för att lagra information om låtar och spellistor i en dict som består av dicts som består av listor som består av dicts så hade vi kunnat skapa en klass för låt och en klass för spellista vilket hade gjort informationen mer lättöverskådlig. Hade vi gjort programmet i Java hade vi nog kommit på det tidigare då det hade blivit omständligt att deklarerar den datastruktur som vi valde. En direktöversättning av datastrukturen från Python till Java skulle sett ut enligt nedan.

```
Map<String[], Map<String[], List<Map<String, String>>>> tracks = new  
HashMap<String[], HashMap<String[], ArrayList<HashMap<String, String>>>>
```

Eftersom det inte finns tupplar i Java skulle vi behövt använda en array, vilka är mutable, vilket gör det riskabelt att ha som nycklar i en map, även om det i vårt fall inte hade varit ett problem då vi inte modifierar nycklarna utan bara läser dem.

Det var inte tänkt att man skulle behöva hämta ett accesstoken för att köra programmet men det var nödvändigt då vi inte kunde få autensiering till spotify på annat sätt. Man var tvungen att registrera sin applikation och ha en "redirect uri" vilket innebär en hemsida att länka till.