## Content-based filtering

Content-based filtering is based upon the users previously ranked content. If the user has listened to a certain artist before, and the user has ranked this artist high. Contend-based filtering will find other artists with the same attributes to the artists the user liked. The more a user has ranked artists, the higher is the users stack of attributes the user likes and dislikes.

To find the artists that are similar to what the users like, there needs to be a database with all the artists, and the artists attributes. These artists can be joined by the category. So if the user is likely to rank rap music high. Then the program will recommend other music with the ‘rap music’-attribute.

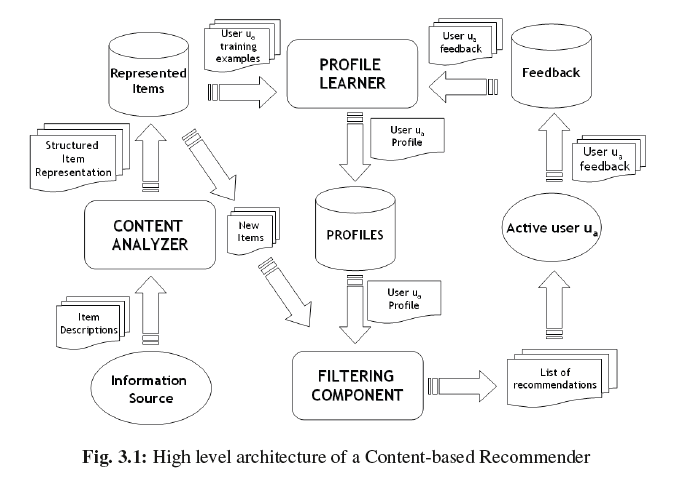
If we look at a high level architecture of a content-based system. We generally see three main components:

**Content analyser:** A way of retrieving data from a source, or from multiple sources. This data will then go into categories, that are easy to read by the other parts of the program. This part of the program will only be called, when there is a change to the data in the source, and the program need to update the database.

**Profile learner:** This part of the program will take the users data, and try to generalise the content to one of the categories. The user data can come in many forms, including a feedback system in the program, or a database that keep track of what kind of music the user listened to the most often. Lastly the user can specify, at the point where they create their profile, what kind of genres they like. With this method they can get recommendations without having ever used the program.

**Filtering component:** This component will take the users content, and the database with everything and compare them. This step should make an ordered list with the music the users has a high likelihood to hear. There are numerous ways this can be done, one of these ways is to create a vector from the users data, on the categories, based on the positive and negative feedback. Then create vectors from the database on the, in this example, artists. And compare the two vectors.

These three main components are joined by smaller components. Now we can connect everything in a diagram seen in figure **X:**



**Caption:** The picture is taken from page 76 in “Recommendation handbook” chapter 3. And illustrates the components of a content-based recommendation system.

**Sources:**

<http://link.springer.com/chapter/10.1007/978-0-387-85820-3_3#page-1>

<https://www.ibm.com/developerworks/library/os-recommender1/>