

# Exhibit 37

---

**From:** jim schrempp [jschrempp@wenet.net]  
**Sent:** Monday, November 29, 1999 5:26 PM  
**To:** Wold Erling (E-mail)  
**Cc:** Vance Ikezoye (E-mail)  
**Subject:** Wired Air Consulting  
**Attachments:** ToMFish1.ppt

Hi,

Attached is a file of two slides that will show you the kind of components I'm thinking of in the new application. In this application I envision a database of 100,000 to 500,000 song fingerprints. Each fingerprint will be the set of feature vectors that describe a 15 second piece of a song. My expectation is that the client software on the user's PC will start a stream analysis in MFish and take out 30 second chunks into a circular buffer.

When the user clicks the "ID This" button the client will transmit the last 180 seconds worth of analysis to a central Wired Air server. That server will invoke a lookup system that will tell us if any of the fingerprints from the database are present in the sample. The server will add the appropriate content (Title, Artist, Link, etc) and return that to the client for display.

I'd like to have MFish build the lookup routine. As I envision it, our code will take out 500,000 fingerprints from the database and load them into some structure. We will initialize the MFish routine by passing in this structure. Our code will accept 180 second samples from the client and pass it to the MFish lookup routine. MFish will return the AudioCBR name of any fingerprints found in the sample. I'd like the routines to return an incorrect identification less than 1% of the time and less than 4% of the time have the routines fail to identify a fingerprint that is in fact in the sample. I thinking that 99% of the lookups occur in less than 5 seconds and no lookup takes more than 15 seconds. We need to specify the system performance we need - how many concurrent lookups can occur on a certain processor and still maintain the performance.

You asked for the first pass idea, this is it. I'd certainly consider you to do other parts of the system, but I think this one plays to your strengths. Brute force won't yield the performance or accuracy needed for this. Your expertise in both the analysis and identification make you the perfect place to implement some kind of clustering of the fingerprints prior to lookup - or some other strategy that optimizes the actual identification.

We could also specify an interface that would allow you to cluster offline. We pass you the fingerprints, you pass back some information that we can store for you. When we next initialize your routine we also pass in this clustering information. This could even be some additional tag field in the AudioCBR object we store.

Let me know your thoughts,  
Jim

-----  
Jim Schrempp  
Wired Air, Inc.  
[j\\_schrempp@wiredair.com](mailto:j_schrempp@wiredair.com)  
408 867-8645

*P/S* Exhibit 1  
Date: 9/4/19 Name: Wold  
Lana L. Lopez, RMR, CRR, CCP, CSR No. 9657

