An Application for Online Music Collaboration (Omni Music)


Sri Lanka Institute of Information Technology, Malabe

Omni Music is the great solution for all the circumstances which have arrival in music industry nowadays. This will drastically reduce all the cost which have to pay when the process of lyrics become a song. And also this will introduce a new business model to music industry by using the methodology of “Service Oriented Architecture”. Composers, accompanists and listeners are benefited from Omni Music web component in several ways. Omni Music provides a software tool to composers to compose music online. They’ll compose music using pre recorded audio samples called ‘Segments’. These segments can be provided by accompanists to partner web services of Omni Music which are distributed in all over the world enabling them to earn extra income by providing music for those segments. Composers use client tool to compose music over the internet and final composed music can be uploaded to Omni Music website for sale. Normal users can purchase those songs from Omni Music store and they let composers and musicians to make money using that way.

Index terms—an Application for Online Music Collaboration using Service-Oriented Architecture (SOA).

I. INTRODUCTION

Producing a song is somewhat complex in its nature. But producing a song over the internet is enormously complicated, because the unpredictable nature of the network bandwidth and due to network latency. Even it is achievable, the quality of sound would be fail to make listeners feel about real symphony is performing.

When considering about current scenario of performing a song composers, accompanists and singers, the three parties behind of this composition have to face many problems. They have to get together in to one place when they have recorded the song by wasting their time and money. Not only that in such situation an artists who is fluent in one quarter have no chance to perform a symphony or producing a song without having the collaboration of others. Omni Music is a web component which can listen as well as compose songs which is a conventional model of music composition. Omni Music will provide a software tool to composers to compose music online. They’ll compose music using pre recorded audio samples called ‘Segments’. These segments can be provided by accompanists over the world enabling them to earn extra income by providing music for those segments. Optimal segments for a given song from all web services have selected by using the technology of web service orchestration. For a better service description this project is used the methodology of web service orchestration and it is used some Semantic Web Features such as Web Ontology Language (OWL-S). The purpose of selecting the best segment with high quality, all the segments are describe by using the language of WSDL and add QoS (Quality of Service). This solution will be a new business model for music industry using the concept called “service oriented Architecture”. According to that concept this will be benefited to all accompanists because they can vend their segments for the highest assessment.

Integration of two or more services into a more complex executable workflow is called “Service Orchestration”.

With the popularity of Web Services, several composition languages for Web Services have emerged in the last few years, such as BPML, XLANG, WSFL [05], WSCDL [06], BPEL4WS [05], GSFL [07] and WS Orchestration. Each specification has its own corresponding workflow engine, however all workflow engines differ in their configuration, deployment specific description, and performance.

II. METHODOLOGY

Omni Music is not a single product and it can’t be accessed by isolating its components. Instead it provides complete software architecture to depict a new business model for the music industry. Basically, Omni Music is composed of following software components or modules

1. OmniWeb – Web interface for the Omni Music

![System Architecture](image-url)
OmniWeb is the web based interface for the Omni Music system. It’s a web application implemented using PHP and hosted under Apache web server. Since this is a web application, any user in the world has access to the Omni Music system through this OmniWeb component. So composers, accompanists and listeners around the world use OmniWeb to access the Omni Music system. OmniWeb is intended to perform following tasks as its day to day operations.

- Provides registration and authentication facilities to its users.
- Provides information about latest music arrivals and top charts.
- Provides a search facility for music.
- Composers create and manage compositions using OmniWeb.
- Accompanists search for various music tracks to perform as their desire.
- Accompanists upload their recorded audio tracks for various compositions.
- Composers publish produced music compositions.
- Listeners can shop for newly produced music.
- Listeners can listen to music.
- Acts as an information portal for partner applications and developers who associated with Omni Music system.
- Provides developer documentations, sandbox certificates and testing environment for developers who develop partner applications for the Omni Music system.

As in any enterprise application, core business logic is implemented in this component. Basically, OmniEnterpriseServer is a Java based enterprise application which resides on JBOSS application server it makes critical business decisions related to the Omni Music system. This component acts as the backend and also a centralized data repository to the system while other application components consume services and write data back and forth using XML web services. Mainly Acts as the data repository for the system. That means other application components read and write data to repository built right into this component. They use XML web services as data transfer methodology. Provide centralized decision making process for the entire Omni Music system. It means OmniEnterpriseServer is the core module of entire system which is handling all other components. As well as provide semantic matchmaking and ranking services to OmniWindowsClient component. By using this service, composer is able to pick the optimal audio tracks for his compositions without worrying about the technical aspects. So by this way composer is shielded from internal technical artifacts which are used to construct the orchestration. Make available registration, authentication and authorization services to all application components. This is also called Identity Management.

This is the client side tool for the music composers. By using this tool, composers could be able to compose new compositions using audio tracks provided by various accompanists.

Facilitate the composer to compose music. This tool has features to rearrange the provided audio tracks by accompanists and enhance them for better performance. Optimal segments according to published composition note will be listed to the composer through the OmniWindowsClient component. Then the composers have only to drag and drop them according to a correct order of composing. As well as this component make possible the composer to produce the audio output of the composition. This means, after creating the composition, it can be outputted as a WAV file. Output can be further encoded into desired format like Windows Media Audio format or MP3 (MPEG Layer 3) format. Finally this will be facilitated the composer to upload the produced music into OmniWeb so that people can purchase it. This mobile client will facilitate the clients to browse latest album releases using his mobile phone when they are moving here and there. And also facilitate the client to download music to his/her mobile.
phone. Not only will that facilitate the client to listen to streaming music which is streamed from OmniWeb.

The whole system is working properly by satisfying all end users and requirements of the project. When a composer going for a composition after downloaded OmniWindowsClient it will be listed all optimal segments according to the given song. After rearranging or modifying them the composer can upload it as a new song to the Omni Web.

III. RESULTS AND DISCUSSION

We have successfully built all the components of Omni Music and successfully demonstrate it.

The entire idea is trying to make it easier for music lovers, composers and accompanists to listen music and form music products by using the technology of Semantic Web Services without going concert halls by wasting their time and money. By considering the main goal of Omni Music it can easily say that it has successfully reached the winning point by introducing a web application to produce as well as purchase music products. This process is going to minimize the human effort which will want to produce a song or any other music product. And also this product will considerably reduce the cost of producing a song.

IV. CONCLUSION

According to the plan OmniWeb, Omni Enterprise Server and OmniWindowsClient have completely designed and future plan is to develop the OmniMobileClient by giving enhanced features to Omni Music.

### TABLE I

<table>
<thead>
<tr>
<th>Segments(m p3)</th>
<th>Capacity of segment</th>
<th>Buffered Time(seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part1.mp3</td>
<td>1.71MB</td>
<td>3s</td>
</tr>
<tr>
<td>Part2.mp3</td>
<td>1.71MB</td>
<td>2s</td>
</tr>
<tr>
<td>Part3.mp3</td>
<td>1.71MB</td>
<td>2s</td>
</tr>
<tr>
<td>Part4.mp3</td>
<td>1.71MB</td>
<td>1.5s</td>
</tr>
<tr>
<td>Part5.mp3</td>
<td>1.71MB</td>
<td>1.5s</td>
</tr>
<tr>
<td>All segments</td>
<td>8.55MB</td>
<td>11s</td>
</tr>
</tbody>
</table>

### TABLE II

<table>
<thead>
<tr>
<th>Segments(m p3)</th>
<th>Capacity of segment</th>
<th>Buffered Time(seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part1.wav</td>
<td>18.8MB</td>
<td>5s</td>
</tr>
<tr>
<td>Part2.wav</td>
<td>18.8MB</td>
<td>3s</td>
</tr>
<tr>
<td>Part3.wav</td>
<td>18.8MB</td>
<td>2s</td>
</tr>
<tr>
<td>Part4.wav</td>
<td>18.8MB</td>
<td>2s</td>
</tr>
<tr>
<td>Part5.wav</td>
<td>18.8MB</td>
<td>2s</td>
</tr>
<tr>
<td>All segments</td>
<td>94.0MB</td>
<td>14s</td>
</tr>
</tbody>
</table>

REFERENCES


