International Journal of Multimedia and Ubiquitous Engineering
Vol.8, No.5 (2013), pp.201-212
http://dx.doi.org/10.14257/ijmue.2013.8.5.19

The Implementation of a Web System for the Remote Management of IPTV Contents

Jaegeol Yim and Gyeyoung Lee

Dongguk University at Gyeongju Korea

{yim, lky }@dongguk.ac.kr Abstract

Abstract

According to analysis at SNL Kagan, the market share of IPTV is rapidly growing whereas the market share of cable TV is slowly shrinking. This paper introduces an 'Open IPTV System' which allows a subscriber to easily open his or her own IPTV broadcasting station. In the 'Open IPTV System', operators must be present at the office to access the content management system (CMS). We implement a web CMS so that the operators can access the CMS anytime and anywhere as long as they are connected to Internet.

Keywords: *IPTV, contents management, web system*

1. Introduction

The IPTV market share is rapidly growing whereas the cable TV market is shrinking. According to an analysis done by SNL Kagan, the number of IPTV subscribers was 40 million, which is 6.1% of all pay broadcast subscribers in the world, at the end of 2010; the IPTV revenue in 2010 was $12.9 billion US, 142% of the IPTV revenue from 2009. In the light of the prosperity of the IPTV business, we have created the 'Open IPTV System'. It is said to be open because it allows subscribers to upload their own content and easily open their own IPTV broadcasting stations.

One of the disadvantages of the system is that operators must be sitting in a chair in front of the desktop computer in the computer room in order to run the contents management system (CMS). The main purpose of this paper is to implement a web CMS so that an operator can run the CMS anytime and anywhere as long as he is connected to the Internet.

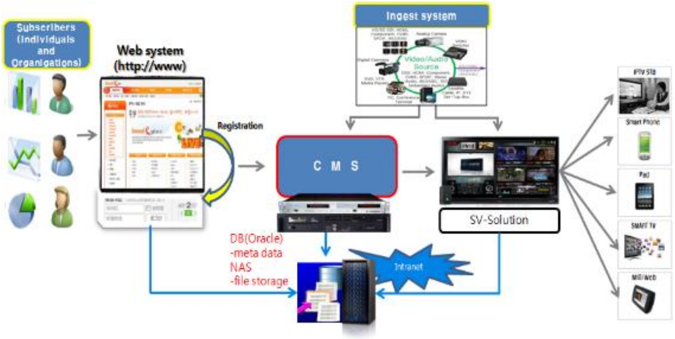


Figure 1. The 'Open IPTV System'

ISSN: 1975-0080 IJMUE

The ‘Open IPTV System’ is described in Figure 1. The main components of the system are the Ingest system, CMS, SV-Solution, and the ‘DB and NAS’ system. The Ingest system is used to encode and transcode the source contents into the IPTV format. For example, an analog video tape needs to be encoded into a digital code before being transmitted through the Internet. The main roles of the CMS are creating digital multimedia contents and saving and retrieving the contents. The SV-Solution is our broadcasting system. If a subscriber requests a video service from a TV, a smart phone, or a PC, then the SV-Solution streams out the video to the subscriber. The SV-Solution consists of many server computers, such as streaming servers, IPTV middleware servers, mobile distributors, and so on. The ‘DB and NAS’ represents the storage system of the system. Multimedia files are saved in the NAS and the metadata for the files are saved in the DB. For this system, this paper implements a web system that allows operators to manage the contents from remote locations.

2. Related Works

This paper develops a web CMS for the ‘Open IPTV System’. Therefore, the IPTV system, contents management, and the web system are included in the related topics. The services provided by IPTV systems can be classified into live television, time-shifted programming, and video on demand (VOD). Among them VOD is most popular [1-3]. An IPTV system stores a lot of videos in its archive system and efficiently retrieves the video the user wants to watch. CMS is the system that manages the video files.

It would be nice if the VOD service knows in what video the user is interested and actively plays it for the user. If a smart phone user takes a picture of a building on a campus then we can conclude that the user is interested in the building. The author of [1] introduced a very efficient building recognition process. The process makes use of sensors equipped on the smart phone and an electronic campus map. It collects the GPS and the orientation values in order to find out the current location of the user and the line of sight of the camera. Then, it finds the elements of the electronic map intersecting with the line of sight. With the intersected elements, it finds the building taken by the camera.

By the definition of IPTV, the quality of IPTV service is very important. The quality of service required by the customers is specified in the Service Level Agreements. The authors of [2] introduced a protocol that supports Service Level Agreements of an IPTV service provider via access control.

One of the most distinguished and beneficial features of IPTV services is the diversity of channels. Even though IPTV provides a lot of channels, the subscribers demand more channels. Therefore, we have to expand the channel capacity. In order to address this problem, the authors of [3] introduced an efficient channel allocation method using the concept of virtual channels.

In [4], the authors illustrated how IPTV would be adopted to wireless and mobile networks. Technical obstacles, such as capability limitations, bandwidth limitations, vulnerable wireless links, coverage implications, and so on were discussed. Business issues regarding mobile IPTV and the mobile IPTV standard status are also discussed.

The existing IPTV architecture dedicates a centralized regional station to serve the subscribers in their respective region. This causes a significant imbalance in resource utilization. In [5], the authors proposed a novel request dispatching protocol that allows IPTV stations of different regions to collaboratively serve user requests.

Quality of Service is one of the key factors to the success of the IPTV businesses. In [6], the authors depicted an end-to-end IPTV network management architecture and developed a unified network management system that monitors and controls the multi-vendor multi-technology broadband network elements. The in-house network management system supports a loosely-coupled systems integration and automatic collection of IPTV-related data from set-top boxes.

IPTV service is widely believed to be the next killer application on the Internet. It is obvious that the success of an IPTV business will depend on the ability of the service providers to provide the right IPTV content and services to the right subscribers at the right time. The authors of [7] propose a business model in which service providers collect content and services and package them along with advertisements to generate the correct IPTV content and services to the right subscribers.

If User Created Content (UCC) is allowed to be a part of IPTV, then the IPTV format is said to be open. One of the challenges in realizing open IPTV is rights protection. To address this challenge, the authors of [8] designed a packaging content authoring and consumption system for IPTV application services based on open IPTV.

A service delivery platform (SDP) is middleware that provides necessary service enablers, such as a web enabler, broadcast enabler, and communication enabler for service creation and deployment. A web enabler interprets instructions written in the standard application program interface (API) format and translates them into legacy APIs. The authors of [9] define the functional requirements of a web enabler.

One of the major features of web 2.0 is in its support of more interactive capabilities than web 1.0. The authors in [10] address the relationship between web 2.0 and IPTV service; an IPTV electronic program Guide (EPG) should use Really Simple Syndication (RSS) to support personalized service. Web 2.0 tagging should be utilized to efficiently categorize IPTV content, an IPTV system should use Web 2.0 technology to easily monitor user-generated content (UGC) to avoid offensive content and copyright infringement and only post UGC that is relevant to the IPTV service. The IPTV EPG should use AJAX to make it more responsive; mash-up and open APIs should be used in IPTV systems.

There are many different types of IPTV set-top boxes. Consequently, while setting up a digital channel for a certain set-top box, the design engineer has had to implement a program. The authors of [11] introduced an apparatus that easily sets up a digital channel without programming.

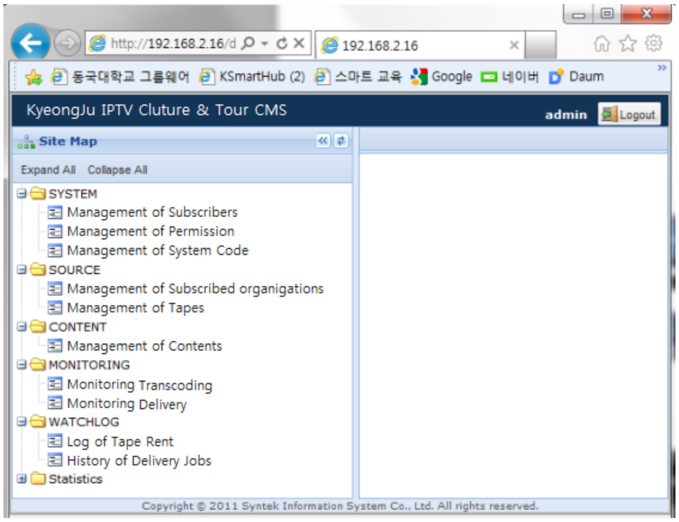
The IP Multimedia Subsystem (IMS) is the first implementation towards reaching converged communications which allows users to communicate with video, audio, and multimedia content. The authors of [12] analyzed the mechanisms of resource allocation and admission control by employing logical interfaces that carry Session Initiation Protocol (SIP) messages.

Wireless mesh networks can receive a TV program and forward it over multiple hops to the user devices that are equipped with 820.11 cards. The authors of [13] described the set up of TV streaming over the Magnets mesh network in Berlin.

3. The Functional User Requirements for the CMS

The CMS functions are categorized into SYSTEM, SOURCE, CONTENT, MONITORING, WATCHLOG, and STATISTICS, as shown in Figure 2. Each of these categories is described further in this section.

‘Management of users’, ‘management of authority’, and ‘management of code’ are submenus of SYSTEM. The ‘Management of users’ provides the functions of displaying user (subscribers) information, searching information with the user ID, creating and registering a new user, updating user information, and deleting a user. User information includes the user ID, name, description, e-mail address, office phone number, mobile phone number, and Detail. Detail is a pop-up button and a web page containing detailed information about the user is linked to it. Changing password and authority level can be done on this page. There are many sorts of authorities: basic authority, administrator's authority, authority to register and delete assets, authority to edit metadata, and delivery authority. The 'Management of authority' lists many different authorities so that an administrator can make check marks on the list to grant the authority to the designated user group (not to an individual user). The last subcategory of SYSTEM is the 'management of code'. This subcategory provides the function of specifying whether or not each of the content items is available.



Copyright © 2011 Syntek Information System Co., Ltd. All rights reserved.

Figure 2. The CMS main menu

SOURCE has 'management of suppliers' and 'management of tapes' as its submenus. 'Management of suppliers/tapes' provides functions used to manipulate the information of the subscribed organizations/tapes. A subscribed organization usually opens its own IPTV broadcasting station using the Open IPTV System and uploads its own content to be made available to the end users. The functions provided by 'management of suppliers' and 'management of tapes' are both similar to those provided by 'management of users'. They are used to display, search, retrieve, update, and delete information.

CONTENT has only one submenu, 'management of contents'. The functions provided by this submenu are also used for displaying, searching, retrieving, updating, and deleting content information. Content information includes title, length (running time), status (available or not), details, preview, and delivery. The details item is similar to the details item used in the user information menu, however this one regards the content format and thumbnail items. Preview plays a low capacity version of the content when it is clicked. Delivery transcodes the content into the H.264 format (for IPTV delivery) and saves it in the BMS (Broadcast Management System) content storage.

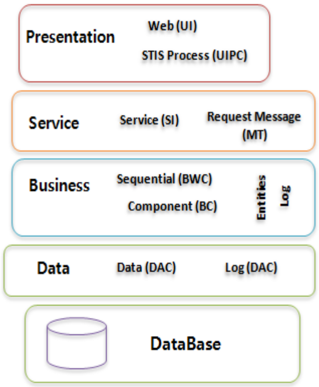
'Monitoring transcoding' and 'monitoring delivery' are submenus of MONITORING. They manipulate the information in the transcoding and delivery functions, respectively. Delivery information includes the registration status (done or not), error type, and status of delivery. We can request redelivery here. As the others do, these submenus also display, search, retrieve, update, and delete information.

WATCHLOG has 'Tape rent' and 'history of delivery jobs' submenus. The information of delivery jobs include the title of the content, the name of the file stored in the BMS storage, the name of the operator, and the date.

STATISTICS includes the numbers of subscribers, subscribed organizations and content items, graphs representing the number of content items for each of the categories, information of content items open to end users, and information of content items closed to end users. The DATA option is available on this menu. It shows the statistical information regarding the designated day, week and month.

4. The Design

As was mentioned in Section 1, the 'Open IPTV System' described in Figure 1 shows that the CMS is already installed and running. The purpose of this paper is to design and implement a web CMS so that operators can access it through the Internet. That means that we should use the database system and library functions in the CMS in our implementation. Therefore, we designed the structure of our system to have four layers, as shown in Figure 3: the Data Access Layer, Business Layer, Service Layer, and Presentation Layer, so that we can use the CMS database as it is.



Technology Selection :
Web UI uses ASP.NET / ExtJs
Service Interface uses WCF
Data Access uses ADO.NET
Database uses SQL Server

Legend
UI = User Interface
UIPC = User Interface Process Component
SI = Service Interface
MT = Message Type
BWC = Business Workflow component
BC = Business Component
DAC = Data Access Component

Figure 3. Our system design strategy

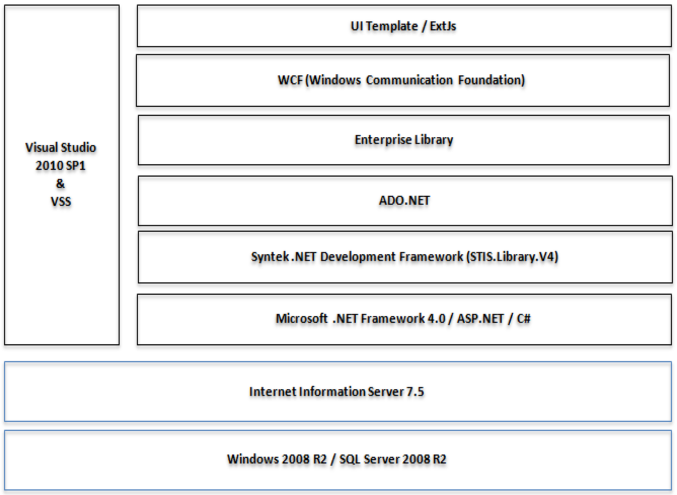

Visual Studio 2010 SP1 & VSS
UI Template / ExtJs
WCF (Windows Communication Foundation)
Enterprise Library
ADO.NET
Syntek .NET Development Framework (STIS.Library.V4)
Microsoft .NET Framework 4.0 / ASP.NET / C#
Internet Information Server 7.5
Windows 2008 R2 / SQL Server 2008 R2

Figure 4. The system components

We used Visual Studio 2010 as the integrated development environment (IDE) for the implementation. Windows 2008 was installed on the server and Internet Information Server

(IIS) 7.5 was run on it. A SQL Server 2008 was used as the database management system (DBMS), Microsoft .NET Framework 4.0 was used for the windows programming, ASP .NET was used for the web programming, and C# was used for the general programming.

In the CMS shown in Figure 1, Syntek .NET, which is a library of database access functions, is already included, and so we used it in our implementation, as shown in Figure 4. For the other database accesses that are not supported by Syntek .NET, we used ADO .NET.

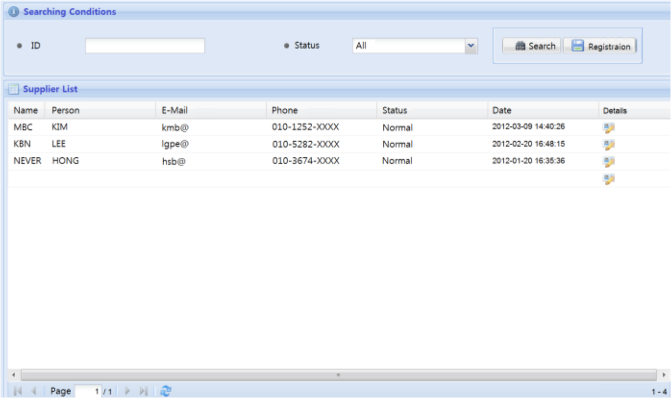
Microsoft provides the Windows Communication Foundation (WCF) as a service oriented application software development framework. We use WCF for the communication between the server and the client.

Recalling the user requirements, we notice that displaying a list of items is done in many places. Listing subscribers, listing contents, listing sorts of authority, listing providers are just a portion of the jobs requiring item lists. Therefore, we made a template user interface for listing items.

These are the components of our system; they are listed in Figure 4. Again recalling the user requirements, we conclude that the functions provided by the menus and submenus can be summarized as a CRUD (create, read, update, and delete) system.

5. The Implementation

We summarized the user requirements into eleven items in Section 3. We also noticed that the functions provided by all eleven items can be summarized as CRUD. This implies that one implementation method can be used to implement each of the eleven items. Therefore, we only discuss our way of implementing ‘reading subscribed organizations’ in this section; the other functions are implemented in a similar manner. As shown in Figure 5, ‘reading subscribed organizations’ displays a list of the subscribed organizations information and buttons and textboxes representing CRUD operations. We implemented a four tier program, as shown in Figure 3.

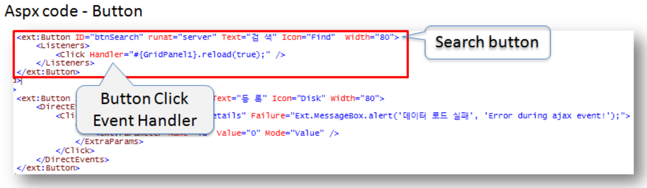


Status: All

Name	Person	E-Mail	Phone	Status	Date	Details
MBC	KIM	kmb@	010-1252-XXXX	Normal	2012-03-09 14:40:26	
KBN	LEE	lgpe@	010-5282-XXXX	Normal	2012-02-20 16:48:15	
NEVER	HONG	hsb@	010-3674-XXXX	Normal	2012-01-20 16:35:36	

Figure 5. The ‘management of subscribed organizations’ user interface

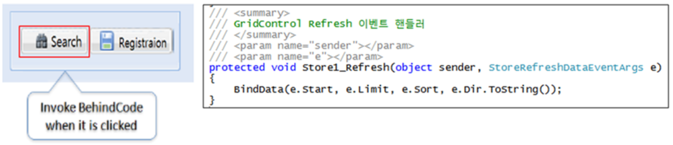
1) Using GridPanel, this program specifies the user interface shown in Figure 5, reads the information about the subscribed organizations from the database, and writes it to the user interface. We can find a button labeled ‘search’ on the user interface. This button is associated with a button click event handler as shown in Figure 6 so that the BehindCode function is invoked when the button is clicked.


Aspx code - Button

```
<ext:Button ID="btnSearch" runat="server" Text="검 색" Icon="Find"  Width="80">
    <Listeners>
        <Click Handler="#{GridPanel1}.reload(true);" />
    </Listeners>
</ext:Button>
]>
>
<ext:Button                      Text="등 록" Icon="Disk" Width="80">
    <DirectE
        <Cli                     etails" Failure="Ext.MessageBox.alert('데이터 로드 실패', 'Error during ajax event!');">
                                 Value="0" Mode="Value" />
            </ExtraParams>
        </Click>
    </DirectEvents>
</ext:Button>
```

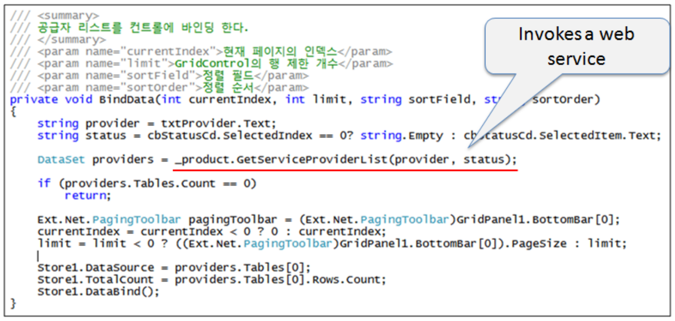
Figure 6. The button click event handler

2) The BehindCode function is defined as shown in Figure 7. It invokes BindData which is a C# function. BindData invokes a web service as shown in Figure 8. The web service is running at the server, accesses the database, and returns the search results.



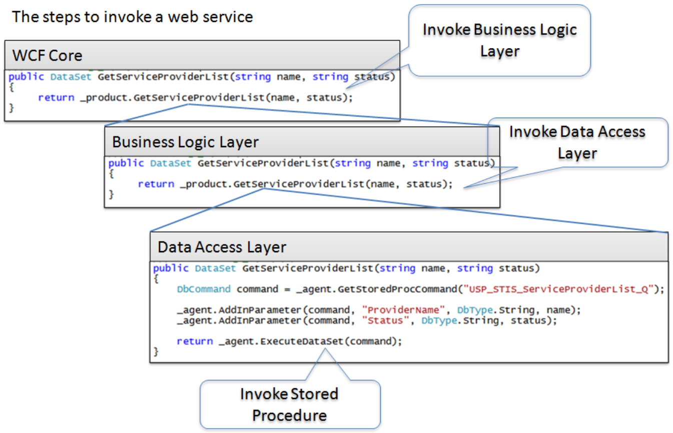
```
/// <summary>
/// GridControl Refresh 이벤트 핸들러
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
protected void Store1_Refresh(object sender, StoreRefreshDataEventArgs e)
{
    BindData(e.Start, e.Limit, e.Sort, e.Dir.ToString());
}
```

Figure 7. The BehindCode definition



```
/// <summary>
/// 공급자 리스트를 컨트롤에 바인딩 한다.
/// </summary>
/// <param name="currentIndex">현재 페이지의 인덱스</param>
/// <param name="limit">GridControl의 행 제한 개수</param>
/// <param name="sortField">정렬 필드</param>
/// <param name="sortOrder">정렬 순서</param>
private void BindData(int currentIndex, int limit, string sortField, st      sortOrder)
{
    string provider = txtProvider.Text;
    string status = cbStatusCd.SelectedIndex == 0? string.Empty : cb tatusCd.SelectedItem.Text;

    DataSet providers = _product.GetServiceProviderList(provider, status);

    if (providers.Tables.Count == 0)
        return;

    Ext.Net.PagingToolbar pagingToolbar = (Ext.Net.PagingToolbar)GridPanel1.BottomBar[0];
    currentIndex = currentIndex < 0 ? 0 : currentIndex;
    limit = limit < 0 ? ((Ext.Net.PagingToolbar)GridPanel1.BottomBar[0]).PageSize : limit;
    |
    Store1.DataSource = providers.Tables[0];
    Store1.TotalCount = providers.Tables[0].Rows.Count;
    Store1.DataBind();
}
```

Figure 8. The BindData definition that invokes a web service


The steps to invoke a web service

WCF Core

```
public DataSet GetServiceProviderList(string name, string status)
{
    return _product.GetServiceProviderList(name, status);
}
```

Business Logic Layer

```
public DataSet GetServiceProviderList(string name, string status)
{
    return _product.GetServiceProviderList(name, status);
}
```

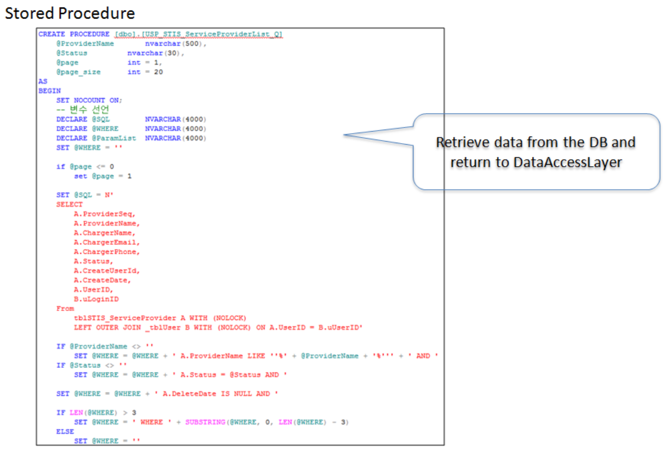
Data Access Layer

```
public DataSet GetServiceProviderList(string name, string status)
{
    DbCommand command = _agent.GetStoredProcCommand("USP_STIS_ServiceProviderList_Q");

    _agent.AddInParameter(command, "ProviderName", DbType.String, name);
    _agent.AddInParameter(command, "Status", DbType.String, status);

    return _agent.ExecuteDataSet(command);
}
```

Figure 9. The steps for invoking the layered methods

3) The web service invokes the method of requesting search. The method is defined at the business layer.

4) The method at the business layer invokes the method defined at the data access layer. These steps of invoking methods are depicted in Figure 9.



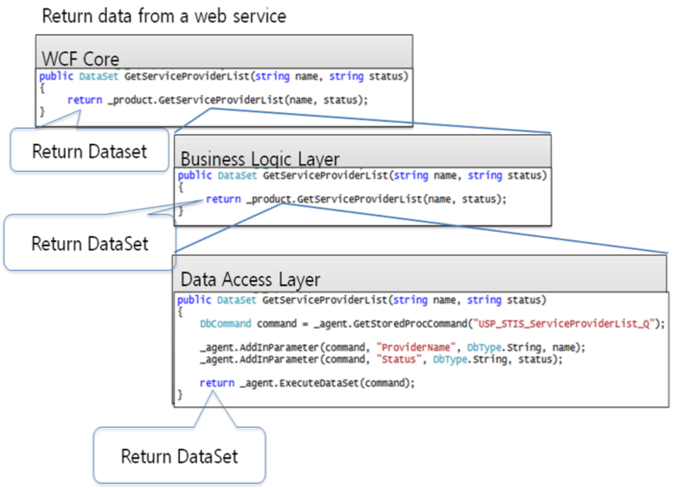
Stored Procedure

```
CREATE PROCEDURE [dbo].[USP_STIS_ServiceProviderList_Q]
    @ProviderName       nvarchar(500),
    @Status         nvarchar(30),
    @page           int = 1,
    @page_size      int = 20
AS
BEGIN
    SET NOCOUNT ON;
    -- 변수 선언
    DECLARE @SQL        NVARCHAR(4000)
    DECLARE @WHERE      NVARCHAR(4000)
    DECLARE @ParamList  NVARCHAR(4000)
    SET @WHERE = ''

    if @page <= 0
        set @page = 1

    SET @SQL = N'
    SELECT
        A.ProviderSeq,
        A.ProviderName,
        A.ChargerName,
        A.ChargerEmail,
        A.ChargerPhone,
        A.Status,
        A.CreateUserId,
        A.CreateDate,
        A.UserID,
        B.uLoginID
    From
        tblSTIS_ServiceProvider A WITH (NOLOCK)
        LEFT OUTER JOIN _tblUser B WITH (NOLOCK) ON A.UserID = B.uUserID'

    IF @ProviderName <> ''
        SET @WHERE = @WHERE + ' A.ProviderName LIKE ''%' + @ProviderName + '%''' + ' AND '
    IF @Status <> ''
        SET @WHERE = @WHERE + ' A.Status = @Status AND '

    SET @WHERE = @WHERE + ' A.DeleteDate IS NULL AND '

    IF LEN(@WHERE) > 3
        SET @WHERE = ' WHERE ' + SUBSTRING(@WHERE, 0, LEN(@WHERE) - 3)
    ELSE
        SET @WHERE = ''
```

Figure 10. The database accessing procedure

5) The Data Access Layer invokes the procedure defined as shown in Figure 10 and stored in the database.

6) The procedure accesses the database and returns the search results. The search results are delivered to the user interface in the reverse order of invoking the methods, as shown in Figure 11.



Return data from a web service

WCF Core

```
public DataSet GetServiceProviderList(string name, string status)
{
    return _product.GetServiceProviderList(name, status);
}
```

Business Logic Layer

```
public DataSet GetServiceProviderList(string name, string status)
{
    return _product.GetServiceProviderList(name, status);
}
```

Data Access Layer

```
public DataSet GetServiceProviderList(string name, string status)
{
    DbCommand command = _agent.GetStoredProcCommand("USP_STIS_ServiceProviderList_Q");

    _agent.AddInParameter(command, "ProviderName", DbType.String, name);
    _agent.AddInParameter(command, "Status", DbType.String, status);

    return _agent.ExecuteDataSet(command);
}
```

Figure 11. The return data steps

We have discussed our implementation of retrieving (read) data from the database. The remaining parts of CRUD, namely create, update and delete, are implemented in a similar manner. For example, we implemented the following functions for ‘management of authority’: getPermissionSetList for reading, addPermissionSet for creating, updatePermissionSet for updating, and deletePermissionSet for deleting. For another example, we implemented the following functions for ‘management of subscribed organizations': GetServiceProviderList for reading, GetServiceProviderInfo for reading detailed information, AddServiceProvider for creating, UpdataServiceProvider for updating, and DeleteServiceProvider for deleting.

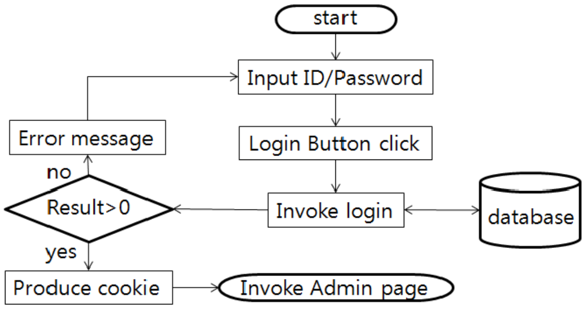


Figure 12. The flowchart for the login process

Our login process described in Figure 12 invokes _user.login with username, password, and IP address as follows:

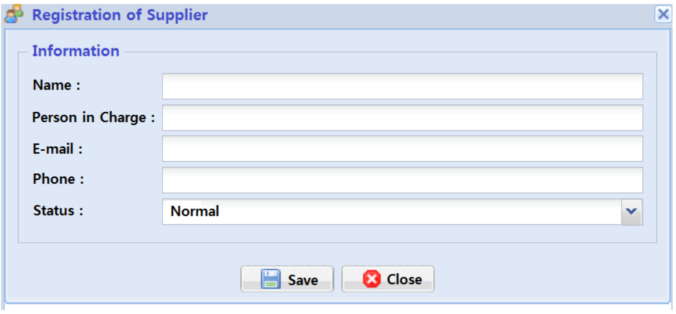
```
int result = _user.login(txtUsername.Text, txtPassworg.Text, GetIpAddress());
```

If the result of login is success then it produces a cookie and identifies the permissions the user has:

```
user loginUser = _user.getUser(result);
if (result > 0)
{
...
  SetCookie(loginUser);
  foreach(permission pm in loginUser.permissions)
  { ...
```

6. The Experiments

A screenshot of the 'management of subscribed organizations' menu is shown in Figure 5. It prints out all the subscribed organizations. It currently prints out five organizations per page. We can easily change the number of organizations per page by assigning a new number to the variable in the program. The button on the top at the right most position is labeled 'Registration'. When we click the button a window similar to Figure 13 pops up. After filling up the form, we then click the 'save' button in order to register a new organization.



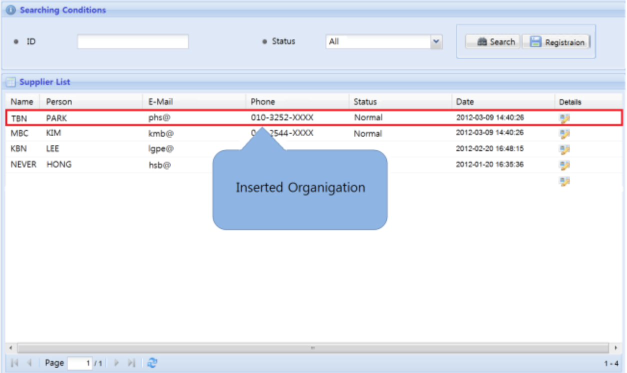
Registration of Supplier

Information

Name :	
Person in Charge :	
E-mail :	
Phone :	
Status :	Normal

Figure 13. The registration pop-up window

The registration result is shown in Figure 14. By investigating the figure, we notice that a new line is inserted into the table shown in Figure 5.



Searching Conditions

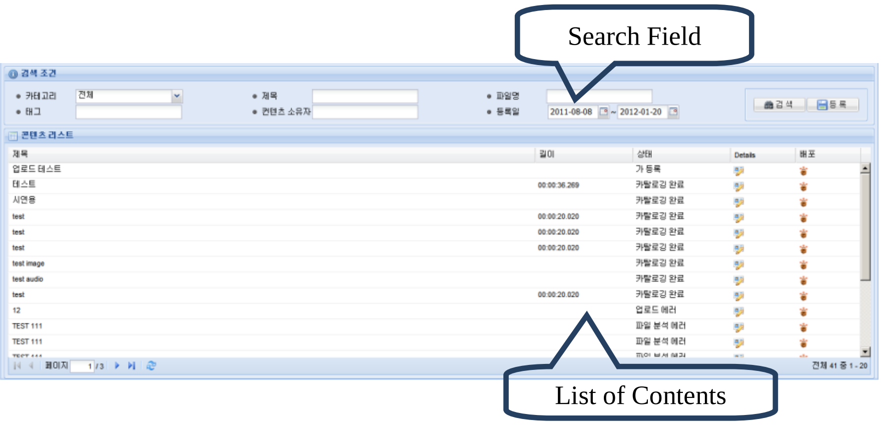
● ID
● Status: All

Supplier List

Name	Person	E-Mail	Phone	Status	Date	Details
TBN	PARK	phs@	010-3252-XXXX	Normal	2012-03-09 14:40:26	
MBC	KIM	kmb@	0[illegible]2544-XXXX	Normal	2012-03-09 14:40:26	
KBN	LEE	lgpe@			2012-02-20 16:48:15	
NEVER	HONG	hsb@			2012-01-20 16:35:36	

Figure 14. A new organization is inserted

As another example of testing the CMS, a screenshot of the content management is shown in Figure 15. We can specify search conditions such as category, title, file name, tag, owner of the contents, registration date. The button located at the top-right is labeled Registration. In the figure, only registration date is specified as search condition, though. All contents that meet the search conditions are listed in the figure. If we want to register a new content, then we click the registration button. If the details button associated with a content is clicked, then a new window as shown in Figure 16 is popped up.



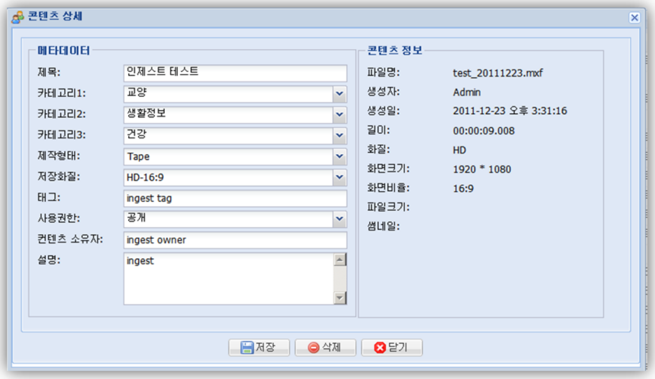
검색 조건

● 카테고리: 전체
● 태그
● 제목
● 컨텐츠 소유자
● 파일명
● 등록일: 2011-08-08 ~ 2012-01-20

콘텐츠 리스트

제목	길이	상태	Details	배포
업로드 테스트		가 등록		
테스트	00:00:36.269	카탈로깅 완료		
시연용		카탈로깅 완료		
test	00:00:20.020	카탈로깅 완료		
test	00:00:20.020	카탈로깅 완료		
test	00:00:20.020	카탈로깅 완료		
test image		카탈로깅 완료		
test audio		카탈로깅 완료		
test	00:00:20.020	카탈로깅 완료		
12		업로드 에러		
TEST 111		파일 분석 에러		
TEST 111		파일 분석 에러		

Figure 15. A screenshot of content management



콘텐츠 상세

메타데이터

제목: 인제스트 테스트
카테고리1: 교양
카테고리2: 생활정보
카테고리3: 건강
제작형태: Tape
저장화질: HD-16:9
태그: ingest tag
사용권한: 공개
컨텐츠 소유자: ingest owner
설명: ingest

콘텐츠 정보

파일명: test_20111223.mxf
생성자: Admin
생성일: 2011-12-23 오후 3:31:16
길이: 00:00:09.008
화질: HD
화면크기: 1920 * 1080
화면비율: 16:9
파일크기:
썸네일:

Figure 16 A screenshot of “content detail”

In the “content detail” window, we can specify detailed information of the content, save the information, or delete the content. The items we can specify include the title, category, media type, quality, tag, permission, owner, and description. A content can belong to up to three different categories. For example, a content can belong to “general education”, “Health” and “Information about daily life”. Our experiments showed that our web CMS meets all the functional user requirements.

7. Conclusion

This paper describes the implementation of a web content management system (CMS) for the ‘Open IPTV System’. Using this system, operators can access the CMS anytime and anywhere as long as they are connected to the Internet.

The design of this system is unique in that it is structured in four layers. One of the advantages of four tier structure is that the implementation of the user interface is independent from the implementation of the database. That is, even when the structure of the database is changed, the presentation layer of the system can remain intact.

It is well known that create, read, update and delete (CRUD) are the four basic functions of a database. Our system consistently provides CRUD to all data items, such as subscribers, content, administrators, subscribed organizations, assets, etc. Therefore, we could apply one design pattern repeatedly on all of the program modules.

The similar design reuse principle was also applied to the implementation of the user interfaces. That is, we repeatedly used one user interface design (refer to Fig. 5 for example) in the implementation of all the program modules. Therefore, we built one user interface frame and it was used in all program modules

Acknowledgements

Lee’s research was supported by the ‘Development of Global Culture and Tourism IPTV Broadcasting Station’ Project through the Industrial Infrastructure Program for Fundamental Technologies funded by the Ministry of Knowledge Economy (10037393). Yim’s research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012-0002611) and (Grants No. C0033172) was supported by Business for Cooperative R&D between Industry, Academy and Research Institute funded Korea Small and Medium Business Administration in 2012.

References

[1] J. Yim, "Implementation of Building Recognition Android App", International Journal of Multimedia and Ubiquitous Engineering, vol. 7, no. 2, **(2012)**, pp. 37-52.

[2] M. Elizabeth, A. Díaz and E. Huh, "Secure Collaboration Mechanism for SLA delivery among IPTV providers", International Journal of Control and Automation, vol. 4, no. 1, **(2010)**, pp. 71-85.

[3] D. Seo, B. Kim, H. Lee and S. Seo, "Virtual Channel Management for IPTV using Channel Domain Systems", International Journal of Control and Automation, vol. 5, no. 2, **(2012)**, pp. 65-72.

[4] S. Park, S. Jeong and C. Hwang, “Mobile IPTV Expanding the Value of IPTV”, Seventh International Conference on Networking, **(2008)**, pp. 296-301.

[5] S. Meng, L. Liu and J. Yin, “Mobile Scalable and Reliable IPTV Service Through Collaborative Request Dispatching”, IEEE International Conference on Web Services, **(2010)**, pp. 179-186.

[6] C. Hu, Y. Hsu, C. Hong, S. Hsu, Y. Lin, C. Hsu and T. Fang, “Home network management for IPTV service operations - A service provider perspective”, Proceedings Network Operations and Management Symposium Workshops, **(2010)**, pp. 1-7.

[7] B. Obele, S. Han, J. Choi and M. Kang, “On building a successful IPTV business model based on personalized IPTV content & services”, 9th International Symposium on Communications and Information Technology, **(2009)**, pp. 809-813.

[8] K. Kim, Y. Lee, J. Yoo, W. Lyu and H. Jung, “The design of the packaging contents authoring and consuming system for IPTV application service”, The 12th International Conference on Advanced Communication Technology, **(2010)**, pp. 586-590.

[9] K. Kim, H. Lee, S. Jo and W. Ryu, “Implementation of Open Web Portal Service Enabler Based on Service Delivery Platform”, Fourth International Conference on Computer Sciences and Convergence Information Technology, **(2009)**, pp. 293-295.

[10] S. Kim and S. Lee, “Web Technology and Standardization for Web 2.0 based IPTV Service”, 10th International Conference on Advanced Communication Technology, vol. 3, **(2008)**, pp. 1751-1754.

[11] M. Lee, “The Service Generation Apparatus for IPTV Interactive Digital Channel”, International Journal of Advanced Science and Technology, vol. 10, **(2009)**, pp.37-51.

[12] H. Yeganeh, M. Shakiba and A. Darvishan, “NGN Functional Architecture for Resource Allocation and Admission Control”, International Journal of Hybrid Information Technology, vol. 2, no. 3, **(2009)**, pp. 11-34.

[13] R. Karrer and T. Huehn, “Ubiquitous TV delivery to the masses”, International Journal of Multimedia and Ubiquitous Engineering, vol. 3, no. 2, **(2008)**, pp.21-34.

Authors

Jaegeol Yim received the M.S. and Ph.D. degrees in Computer Science from the University of Illinois at Chicago, in 1987 and 1990, respectively. He is a Professor in the Department of Computer Science at Dongguk University at Gyeongju Korea. His current research interests include Petri net theory and its applications, Location Based Service, AI systems, and multimedia systems. He has published more than 50 journal papers, 100 conference papers (mostly written in Korean Language), and several undergraduate textbooks.

Gyeyoung Lee received the M.S. degree in Computer Science from the Dongguk University in 1982 and the Ph.D. degree in Computer Engineering from the Dankook University in Korea in 1992, respectively. He is a Professor in the Department of Computer Science at Dongguk University at Gyeongju, Korea. His current research interests include Petri net theory, AI systems and Speech processing systems.