# Development Of ITSM Attachment Viewer: An Web UI Application

**Yuvaraj N[1] Dr Cauvery N K[2]**
[1]M.tech Student [2]Head of the Department
[1,2] Department of Information Science
[1,2]R.V. College of Engineering, Bangalore, India

*Abstract*--- In recent years organizations are switching into more and more automated systems. The main challenge faced by the service providing organization is time constraint. All the request has to be processed and results has to be provided within the specified short span of time.

The purpose of the application tool is to implement a platform for extracting the information technology service management (ITSM) attachments from BMC application server and provide those attachments for viewing and downloading over a web user interface (UI). The application tool utilizes operational data store (ODS) model. The ODS data model was designed to integrate the heterogeneous ITSM transaction data and support the direct copy from different data sources to ODS database by extraction, transforming and loading (ETL). There are thousands of ITSM transactions on daily basis, for which the attachments has to be extracted, transformed and loaded.

**Keywords:**-extraction-transforming-loading, operational data store, ITSM, BMC.

## I. INTRODUCTION

ODS acts as a centralized repository for the ITSM and holds all the transaction related data in a structured format which is then exposed to Business Object (BO) Universe for reporting purpose.

ITSM uses BMC's Remedy AR (Action Request) System to provide a consolidated Service Process Management platform for automating and managing Service Management business processes. ITSM manages services for different type of transactions, namely Incident, Task, Service Request, Change and Work Order transactions. For each type of transaction the client provides information about the incident in the Forms provided by the BMC's Remedy AR System. The information includes text data, attachment data about the incident which are stored in ITSM database. Because of the security reasons the data obtained by the client is encrypted and stored. This information provided by the client about an incident is used to resolve the incident. As part of Data collection, the ODS system gets the attachment data for all kinds of transaction and stores the same in ODS.

For business purpose clients request for the attachment data that they provide during change, incident, task, work order and service request transactions. Attachment data of each transaction loaded on to ODS cannot be read as it is, because of the encryption mechanism enforced while saving the attachment to ITSM database. To overcome the issue, ODS uses the Action Request (AR) System's Application Programming Interface (API) which queries the raw ITSM table and extracts the attachment in its original format.

For different types of transactions ITSM stores the attachment data individually in different tables. This attachment table names are defined in a specific manner in order to distinguish between the attachments associated with different fields of the transaction. In a transaction, the ITSM database maintains a separate table for every attachment field in a form which stores the attachment data.

In this paper transaction details are collected in eXtensible Markup Language (XML) format and have included an enhanced way of extracting attachments from AR server, where we can extract number of attachments sequentially irrespective of the transaction type. Here we use BMC's AR API's to connect to the AR server.

## II. NEED FOR AN XML BASED DATA COLLECTION

XML design goals are mostly emphasized on simplicity and generality. It is classified as an extensible language, because it allows the user to define the mark-up elements. XML's purpose is to aid information systems in sharing structured data to serialize data and to encode documents [1]. XML is a structured document format that includes not only the data but also metadata that describes that data's content and context in a way that a person or machine can understand it. Other text based formats contain data in predefined locations or contain delimiters. Metadata naming can be done with relative names that increases the understandability of the document [2]. Each metadata can have attributes specified to it, which helps when one metadata has many values. Creating XML document of transaction details also reduces the database server cost.

XML is designed to separate syntax from semantics to provide a common framework for structuring information. XML data is stored in text format. This makes it easier to expand or upgrade to new application, or new browsers, without losing data. Exchanging data as XML greatly reduce the complexity, since the data can be read by different incompatible applications. It supports internationalization and platform independence [3].

## III. STATEMENT OF THE PROJECT PROBLEM

The clients request for the attachment data of the transaction. One of the underlying problems with the current system is the attachment extractor which is a console application needs to be run for all kind of transaction one by one and for each run a separate kind of configuration needs to be maintained, so that the solution queries the proper table for extracting the attachment data. Once the attachments are extracted, attachments needs to be sent to the client which requires human intervention. This entire process is a time consuming one.

## IV. PROPOSED SYSTEM

The basic idea is to design and implement ITSM attachment viewer tool that automates the attachment extraction process for all kinds of transactions with the use of XML based data representation as allows users to provide processing control information to programs [4]. This technique will eliminate

 

the human intervention in attachment extraction process, making the process simple and fast. Data used for attachment extraction is represented XML format. The BMC application system can make use of this XMLised reliable data for extraction of attachments of all the transaction sequentially. The XML representation promises interoperability and portability as it can exchange the data as required by the application and the ITSM database.

The proposed system working architecture is as shown in figure 1. Large system is decomposed into sub-systems that provide some related set of services, namely collector, extractor and user interface. The data flow of the attachment viewer is as shown in the figure 2, where collector collects the required transaction details of each and every transaction in XML format and produces it to the extractor. Extractor utilises BMC application programming interface (API) for extracting the attachments from BMC application server.

Extractor utilizes transaction details in XML format for extraction of the attachment in byte array format. BMC application server takes transaction details and hits the ITSM database and extracts the attachments in readable format. The attachment that is extracted is updated to the ODS database. All the attachments are extracted and updated sequentially. Processing power of ODS database server is utilized only during the collection of the transaction details in XML format.

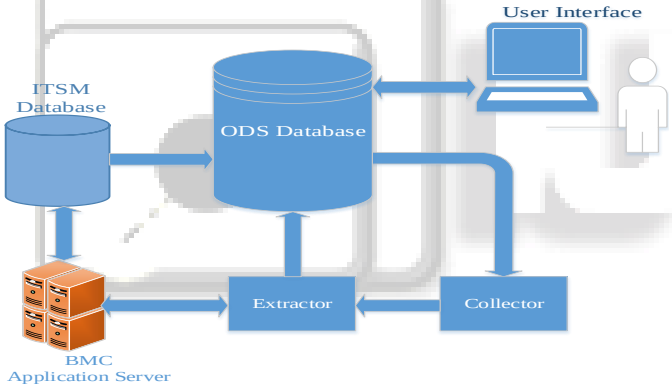


Fig. 1: Working architecture diagram of ITSM attachment viewer.

User request the system through user interface (UI). User request provides the transaction id in particular field, the user interface engine parses the requests and sends the parsed request to the ODS database. UI provides attachment data in readable format that has to be downloaded and viewed.

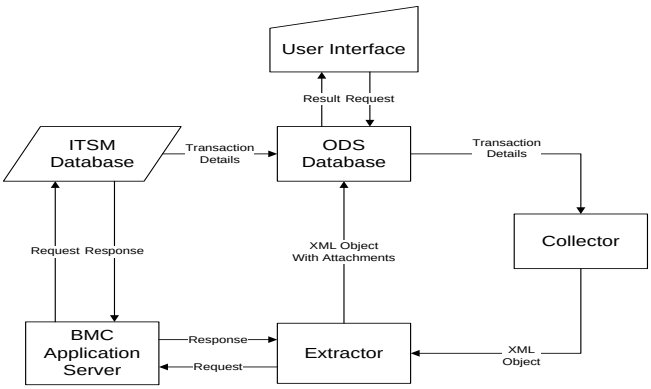


Fig. 2: Flow diagram for ITSM attachment viewer.

## V. RESULT

The login page of the tool is shown in the figure 3, where the user provides his credential and authenticates himself to the tool. The tool verifies and validates the credential provided and based on verification, tool logs into main page that is the input page as shown in figure 4.

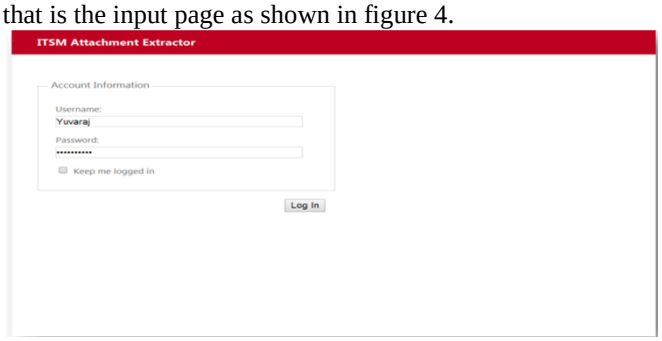


Fig. 3: Login screen

In the input page we select the type of the transaction in the drop down menu. The input is processed with the above given values and the transaction details along with the download button for attachments is provided as show in the figure 5. On clicking the download button, attachment gets downloaded on to the local machine.


Fig. 4: Input screen to select transaction type and provide transaction ID

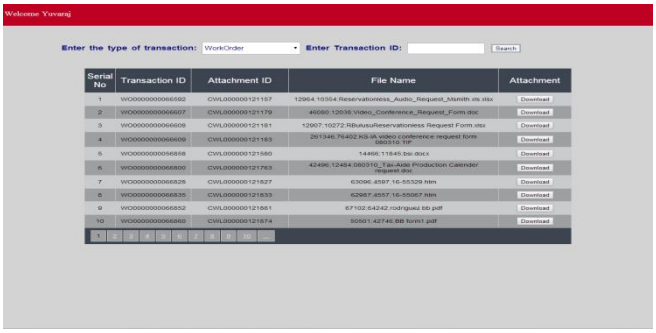


Fig. 5: Output screen-when a transaction type is selected

By selecting the type of transaction and providing the transaction id, we get the attachments of that particular transaction only as show in figure 6.

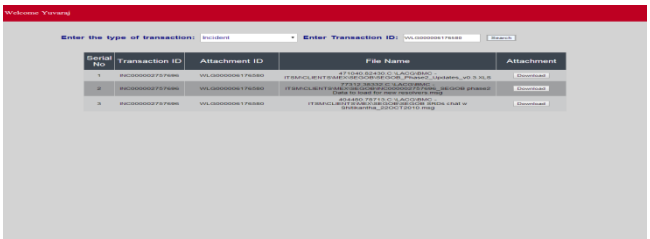


Fig. 6: Output screen-when a transaction type and a transaction ID is provided

## VI. CONCLUSION

This project will be useful in corporate environment where an organisation processes thousands of transactions and maintains the details of the same for different clients on daily basic. This tool is assert to ITSM provider that presents the transaction attachments to the clients over the internet. This approach eliminates the human intervention in decrypting attachments into readable format, downloading and providing attachments to clients.

## REFERENCES

[1] Wansa Paoin and Pranot Boonchai, "Development of Surgical Operation Data Interchange Model Using XML and Relational Database", Eighth International Symposium on Natural Language Processing 2009.

[2] Ferreira R, Jaão M, Ramiro M, and Marta P, "XML Based Metadata Repository for Information System", Proc IEEE Artificial Intelligence, Portuguese, 2005, pp. 205-213.

[3] A.deVos, S.E. Widergren, J.Zhu, " XML for CIM Model Exchange", International Conference on Power Industry Compuyer Applications 2001.

[4] Marta H. Jacinto, Giovani R. Librelotto, Jose C. Ramalho and Pedro R. Henriques, "Bidirectional Conversion between XML Documents and Relational Data Bases", Seventh International Conference on Computer Supported Cooperative Work in Design 2002.

[5] Oviliani Yenty Yuliana and Suphamit Chittayasothorn, "Conceptual Schema Based XML Schema with Integrity Constraints Checking", International Conference on Convergence and Hybrid Information Technology 2008.

[6] "Extensible Markup Language (XML) 1.0", W3C Recommendation, Feb.1998, http://www.w3.org/TR/REC-xml.

[7] T.Werner, C. Vetter, T.Kostic, V.Lohmann, "Data Exchange in Asset Management Applications for Electric Utilities using Xml", Proceedings of the 5th International Conference on Advances in Power System Control, Operation and Management, APSCOM 2000,Hong Kong, October 2000

[8] Zhongyu Chen, Ting Guo, Zhongsheng Qian, Chunshui Xiao, "Research on the Method of Data Extraction Based on Category", 12th International Conference on Computer and Information Science (ICIS), 2013.

[9] Xishui Pan, Xuezhong Zhou, Hongmei Song, Runshun Zhang and Tingting Zhang, "Enhanced Data Extraction, Transforming and Loading Processing for Traditional Chinese Medicine Clinical Data Warehouse", 14th International Conference on e-Health Networking, Applications and Services 2012.

[10] Wikipedia (2014). XML. [online encyclopedia]. Available: http://www.wikipedia.org/wiki/XML.

 