# A CLOUD BASED DATA WAREHOUSE MODEL USING MODEL DRIVEN ARCHITECTURE APPROACH TO GENERATE MATERIALIZE VIEWS FOR BUSINESS INTELLIGENCE

ANIL SHARMA AND DR. MANU SOOD
*Department of Computer Science, Himachal Pradesh University, Shimla, India*
*Email: anilmanu2001@hotmail.com, soodm_67@yahoo.com*

**ABSTRACT**: Cloud computing based on the recent Information and Computing Technologies (ICT) offers software programs, platforms, infrastructure etc. as services. On the other hand, Business Intelligence (BI) on the Cloud is an assistance model that helps organizations further in their strategic business interests and it is for this reason, every organization is looking forward for its generation and effective usage. Business Intelligence systems and technologies are essentially designed in context to Data Warehouse because of the complexity in the decision making process which needs to be analyzed a huge historical data. And, authors suggest the utilization of materialized views to generate fast and relevant business reports by using data from data warehouse to formulate the Business Intelligence. Cloud SaaS refers to a cloud computing service model in which the software applications are offered as services. With the continuous advancements in ICT, the organizations are under constant pressure of replacing the older ones. Model Driven Architecture (MDA) is a software development model that aims at minimizing the impact of these technological advancements on software applications development and it is also the suggested model in the design the multi dimensional schema for the data warehouse. In this paper, authors are suggesting a cloud based data warehouse model and business intelligence through materialized views as a service in SaaS model of SOA using MDA approach.

**KEYWORDS:** Business Intelligence (BI), Cloud Computing, Data Warehouse, Materialized Views, Model Driven Architecture (MDA), Service Oriented Architecture (SOA), Software as a Service (SaaS).

## INTRODUCTION

A Data Warehouse (DW) is a repository of massive data that is used for query and analysis rather than for routine operations. The source of the data for DW is on-line transaction processing (OLTP) systems. DW receives data from diverse sources, performs aggregation and summarization on the data to renovate it into information for business users and allows them to achieve greater productivity and to take more conversant decisions. These benefits are encouraging more and more organizations to look forward to implement data warehouse technologies for gaining viable advantages [1].

Model Driven Architecture (MDA) is a software design approach for the development of software systems. It provides a set of guidelines for the structuring of specifications, which are expressed as models. It is a kind of domain engineering, and supports model-driven engineering of software systems. It was launched by the Object Management Group (OMG) in 2001. The MDA approach defines system functionality through a platform-independent model (PIM) using an appropriate domain-specific language. And then, the PIM is translated to one or more platform-specific models (PSMs). This requires mappings and transformations and should be modeled too [2][3].

Cloud Computing provides resources like hardware and software as services over the network. Cloud computing has evolved from a range of relevant legacy technologies and concepts such as grid computing, virtualization, Web Service Definition Language (WSDL), Simple Object Access Protocol (SOAP) and Universal Description Discovery and Integration (UDDI), Service Oriented Architecture (SOA), Software as a Service (SaaS) etc. The Software as a Service (SaaS) which is a part of classification of Cloud Computing is an emerging business model in the software industry. It is a software application delivery model by which an enterprise vendor develops a web-based software application, and then hosts and operates that application over the Internet for use by its customers [4].

Business Intelligence (BI) is a combination of systems and technologies which helps in strategic planning to have a better decision making for the organization. BI systems help the company to gather, store, access and analyze corporate data to assist in decision-making process. BI as a discipline is made up of several related activities, including data mining, online analytical processing, querying and reporting. BI as a concept provides a means to obtain the crucial information to improve strategic decisions and therefore plays an important role in current decision support systems [5]. Because of the profound competition in different areas of the business like customer profiling, customer support, market research, market segmentation, product profitability, statistical analysis, inventory, distribution analysis etc., there is an immense need of BI [6].

A Materialized View (MV) is a database object that stores the results of a query. MV is a snapshot of data collected from remote tables. MV has an approach in which the result of the query is stored as a concrete table that may be updated from the master tables on regular basis. MV helps in attaining more efficiency in the data access, at the cost of some out of date data. This characteristic of MV is the most useful in data warehouse scenarios, where recurrent queries on the master tables can be expensive. MVs obviously pre-compute joins and aggregations and, reduce the query execution time significantly, when required. MVs are widely used for the formation of business reports which supports the decision making process. To ensure accurate outcome, MV must be up to date at the time of access [7].

Rest of the paper has been organized in various sections. In section 2, the authors suggest aligning the whole process of the multi dimensional database development after the conversion from relational database by following the MDA approach for the generation of star schema. Section 3 and 4 highlights the usage and creation of CIM and PIM for the Result Evaluation System; an example to illustrate the concept and section 5 explains the creation of star schema for the example taken as an illustration. Section 6 describes the business intelligence as a service on cloud. In section 7, the authors suggest the steps to create the SaaS Architecture for the cloud BI whereas section 8 highlights the procedure to develop the web services business process model followed by section 9 that highlight the significance of application of MDA and SOA approaches respectively. Section 10 explains the usage and generation of materialize views. And finally section 11 highlights the conclusion and the scope of the future work.

**MULTI DIMENSIONAL DATABASE USING MDA**

The authors suggest aligning the whole process of the multi dimensional database development after the conversion from relational database by following the MDA approach for the generation of Star Schema. As an illustration to support the proposition, the authors are taking an example of a Result Evaluation System (RES) of an educational institution as an instance to describe the design

of a multidimensional database. In a process of taking timely and strategic decisions, easy and fast access to students' performance data is the key. Along with data access, institute also needs to analyze students' performance from a range of perspectives. Because of this detailed view of each individual student's performance, teachers will in a better position to identify probable problems and to take the suitable and preventive measures to ensure that each student can perform to the best of his abilities.

In the educational institution taken as an example, there are different colleges like Management College, Computer Science College, Pharmacy College, and Hotel Management College. And these different colleges are maintaining their different databases related to various entities likes Admission, Course, Subject, Faculty, Student Personal Information, Internal Marks, External Marks, and Accounts etc. and there are different attributes associated to these entities as well. While designing RES database manually, it is difficult to handle technical complexity determined in database system due to the problems of maintainability and reusability. Hence, the authors here propose a design for the prototype database for this system using MDA approach. In database design process using this approach, MDA enables the segregation of RES database into various models at different level of abstraction**.**

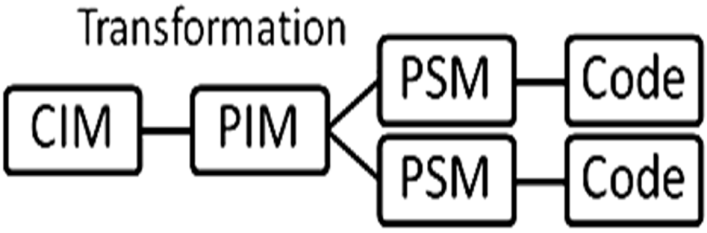


Figure 1.[8] Model Driven Architecture Framework

As shown in Fig. 1[8], the first level of abstraction as a CIM describes requirements of the system which specifies the business model. At the second level of abstraction, a PIM describes the software specifications defining the domain model of the system [3]. The third level of abstraction as a PSM describes the software realization model and defines detailed design of the system. Then, the resulting database design can be implemented for a particular required platform. These transformations are formally specified by using a standard called QVT (Query/View/Transformation) [9]. Finally, the necessary code to implement the multi dimensional database model can be obtained from PSMs.

**COMPUTATIONAL INDEPENDENT MODEL (CIM) of RES**

In MDA, system requirements are modeled using a Computation Independent Model (CIM). This model is called business model and it uses a vocabulary that is familiar to the domain experts. A CIM does not show details of the systems structure, but the environment in which the system will operate, being useful to understand the problem [7]. After understanding business needs of RES, an MDA software developer specifies the Fig. 2 [10] that may occur in the appropriate working of RES. It presents a CIM that has been developed for Students Result Evaluation System (RES) and consists of business entities and the relationships among them.

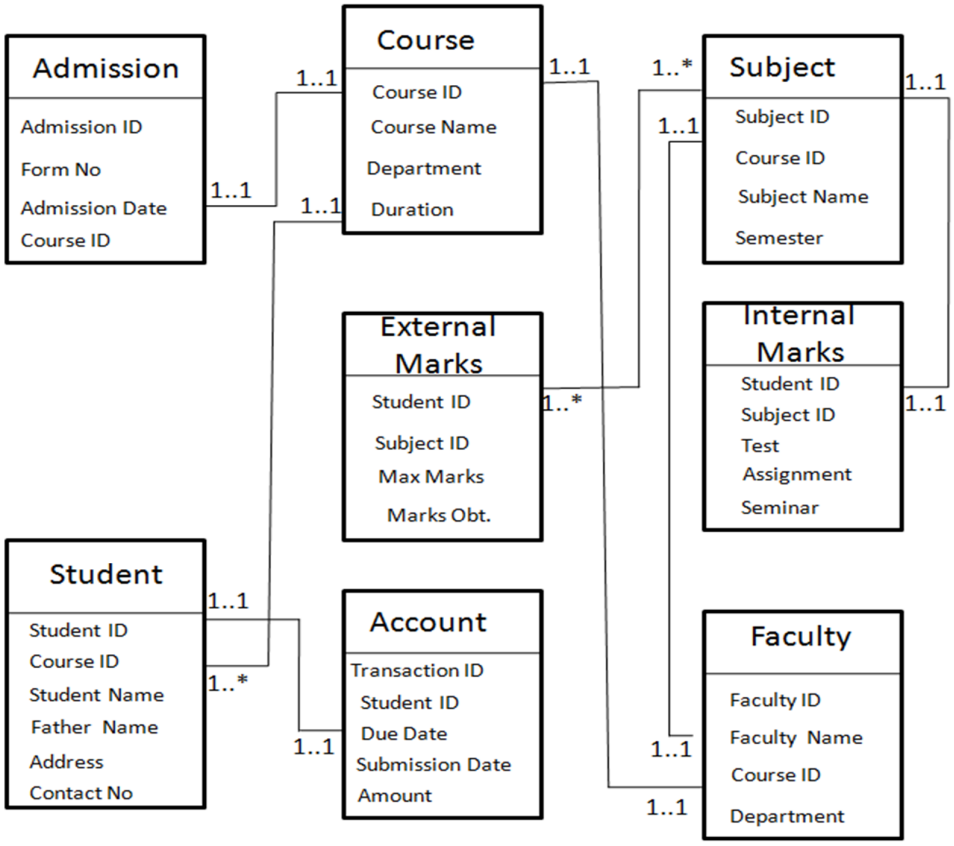


Figure 2.[10] CIM of RES

**PLATFORM INDEPENDENT MODEL (PIM) of RES**

At the second level of abstraction is defined the Platform Independent Model (PIM). This model is at a relatively high abstraction level and by definition, is independent of any implementation technology. This PIM is a platform independent analysis model can be derived by analyzing the requirements model and will form the basis for the design of the multidimensional data model. The system functionalities are described through this PIM maintaining traceability to the requirements model. Developers may use appropriate model elements stored in a model repository to produce some parts of this PIM. This model is not the final PIM, but forms the foundation for producing the final version. Conventional Object Oriented (OO) analysis techniques can be used for this activity, which is typically executed in an iterative and incremental manner [11]. As shown in Fig. 3 [10], PIM reflects all the information needed to describe the platform independent system behavior. PIM aims to capture implementation-independent information about the system and business process modeled. As an OMG standard, MDA uses the UML models as its core representation. IBM's Rational Rose 2003 and OMG's XML Metadata Interchange (XMI) are thus popular tools that support the process [12].

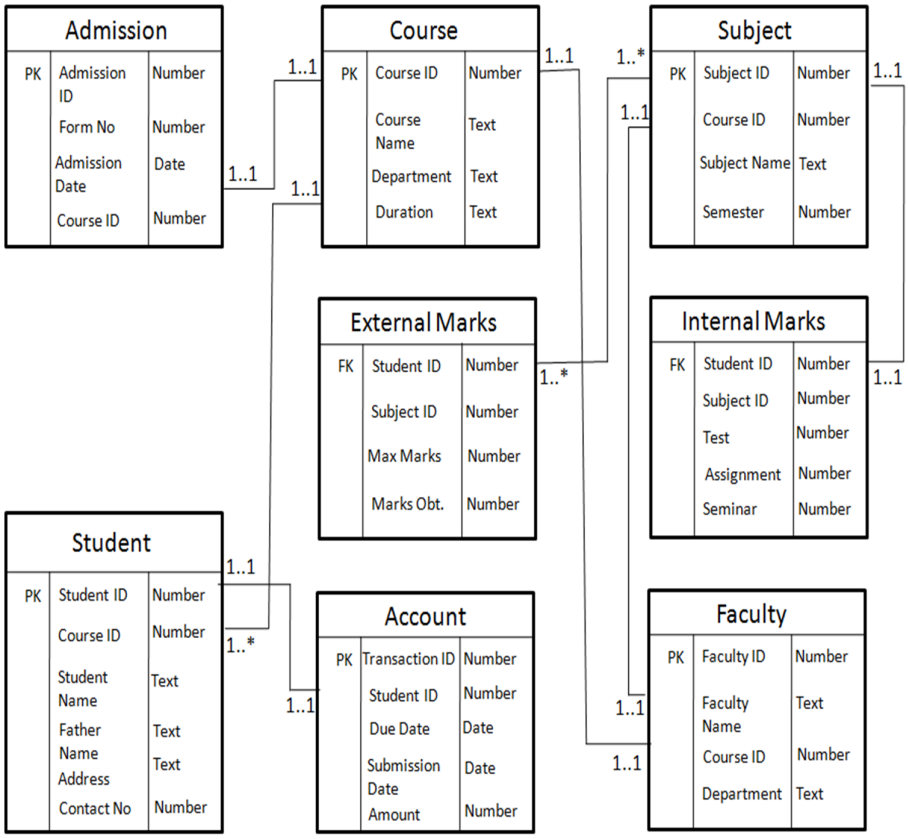


Figure 3. [10] PIM of RES

**STAR SCHEMA DESIGN OF DATA WAREHOUSE OF RES**

Star Schema is the most common modeling paradigm for multi-dimensional databases, in which the data warehouse contains: (a) fact table, containing the bulk of the data, with no redundancy, and (b) a set of dimension tables, one for each dimension. Fig. 4 [10] depicts the star schema of data warehouse for RES using one fact table along with multiple dimension tables. The schema graph resembles a starburst, with the dimension tables displayed in a radial pattern around the central fact table.

**BUSINESS INTELLIGENCE AS A SERVICE ON CLOUD**

Data Warehouse is the base to formulate Business Intelligence. BI systems and technologies are designed with the primary goal of extracting meaningful data from an organization's raw data to disclose insights to help a business to make faster and more accurate decisions. The systems and technologies typically integrate data from across the enterprise and provide end-users with self-

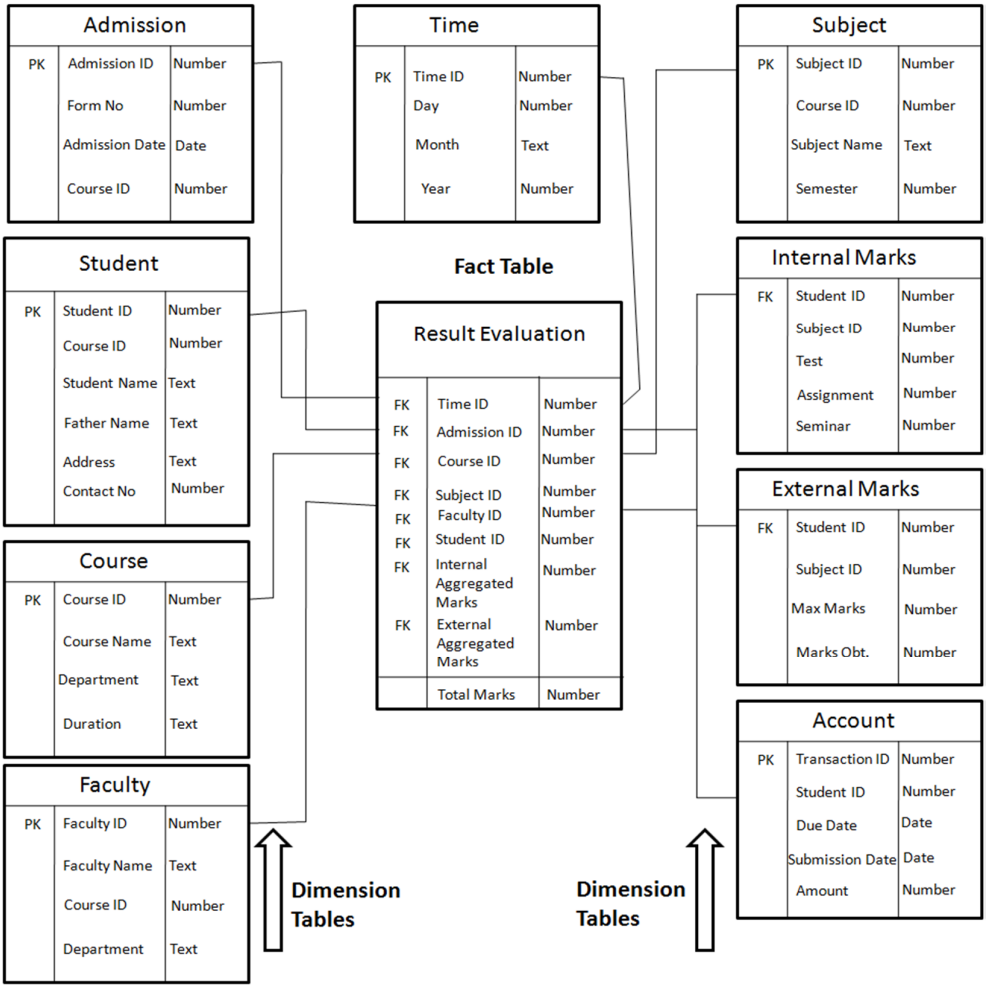


Figure 4.[10] Star Schema of RES

service reporting and analysis. The past two decades have seen explosive growth, both in the number of products and services offered and in the adoption of these technologies by industry. This growth has been amplified by the declining cost of acquiring and storing very large amounts of data arising from different sources. Businesses are leveraging their data asset aggressively by deploying and experimenting with more sophisticated data analysis techniques to drive business decisions and deliver new functionality such as personalized offers and services to customers [13]. Cloud BI is an assistance model that is growing gradually and many of the organizations are looking forward for this. Cloud SaaS refers to a cloud computing service model in which the software applications are offered as services. The advancements in the field of computing and information require development of software solutions in a manner that is independent of technological advancement. The MDA approach may be leveraged to develop the software

applications that would be deployed in the cloud, the cloud Software-as-a-Service (SaaS). A platform-independent model (PIM) of the cloud application would reflect its structure, behavior and functionality irrespective of the technology used for its implementation. The platform-specific model (PSM) of the application, on the other hand, would be more implementation-oriented and bound to a given execution platform. The transformations from the PIM to PSM would be carried out using transformation tools developed for the purpose.

**SAAS ARCHITECTURE FOR CLOUD BI**

Cloud SaaS is a software application delivery model by which an enterprise vendor develops a web-based software application, and then hosts and operates that application over the Internet for use by its customers. Sharma et al. in [4] suggest that many of the efforts are done in the development of these services in-house but with the advancements in the communication technology these services become obsolete and need to be replaced on regular time interval. So the need of the hour is to develop these services in such a way that these should absorb the technological changes and should be scaled up accordingly. By keeping this into the context, the authors of this paper proposed the usage of Model Driven Architecture (MDA) to develop the services to formulate Cloud based Business Intelligence.

The proposed problem needs two fold solutions. First the design of data warehouse for the RES and then design of business intelligence services on cloud computing. The authors have already suggested the MDA approach to design the relational schema for RES and further designed the Star Schema of RES using MDA. After designing Star Schema, there are many of the Service Providers those who provide the DWH as a service on cloud. We can choose any one for the reference. In this paper, authors are focusing on the design of various services which will be dealing with this cloud based DWH to generate business intelligence. The architecture of the Cloud BI SaaS is given in the Fig. 5 [14]. Data collection is done from various operational systems and data belongs to various types like marketing data, sales data, finance data etc.

Business Organizations also collect data from External sources to know about what their counter parts are doing to enhance their business capabilities. Data collection from heterogeneous data sources is a complex task and the Authors propose a service to fetch data from these various operational systems. After fetching the data, next service is to perform the data cleaning process. Data cleaning includes the finding of missing values, spurious values, misspelled data etc. Data Integration Service will take place after the completion of data cleaning process and finally Data Upload Service will take charge to upload this fine tuned data to the data warehouse repository. At this point of time the whole data is centralized and to improve the performance of query response time, this data warehouse can be distributed further into Data Marts. Data Marts are created to keep the data at the nearest site i.e. data related to subject of interests to the respective users. Now Business users need to interact with the data warehouse to formulate business intelligence. Inputs given by the users will be in the form of a query. So, Query Input Service will serve this purpose. After receiving the query input successfully, Query Buildup service will take charge to build the query in the required format. And finally to send the output of the query back to the Business Managers, Query Output Service will take the charge [14].

The structure for various services is shown in the following Fig. 6 [14] which will serve the purpose of RES data warehouse. All these services and their interfaces will be developed by using the MDA approach. Interoperability of the services will also be taken care with the help of MDA.

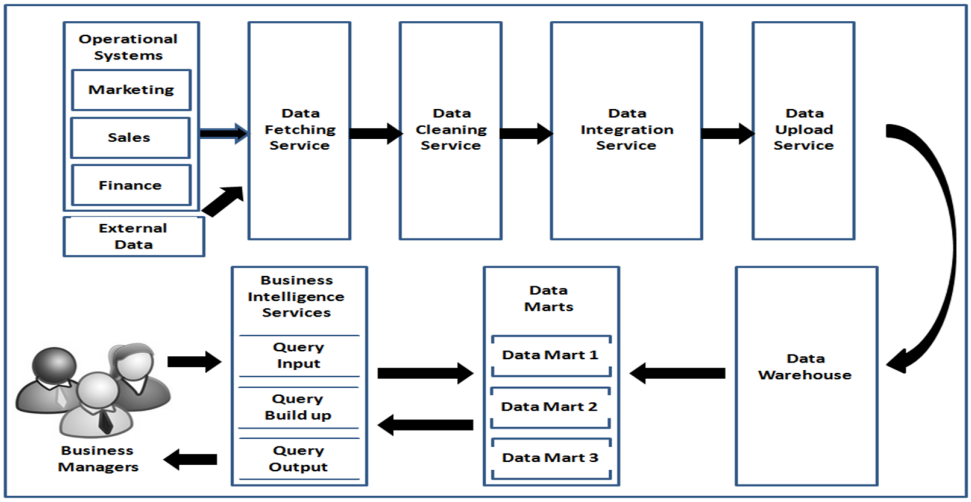


Figure 5.[14] Cloud BI SAAS Architecture

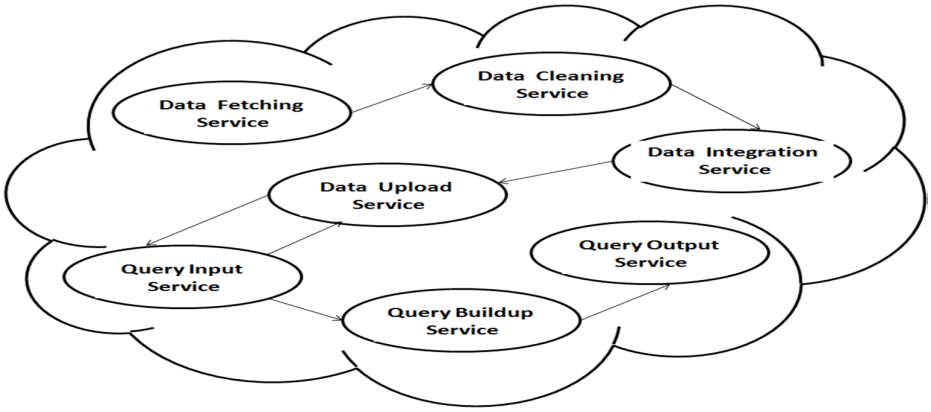


Figure 6.[14] RES Services Model

**WEB SERVICE – BUSINESS PROCESS MODEL USING MDA**

MDA based development of cloud SaaS (application) will enable defining web services in a technology-independent manner and will play a significant role in improving the quality of cloud software services, making them more robust, flexible and agile. Encapsulating business logic in a manner that is independent of the technical mechanisms will formally capture the essence of the applications; and will also make it possible to reuse them in a variety of contexts [15].

To illustrate the concept, we take an illustrative example of the data warehouse of RES as a service available on the cloud. As such, there are many vendors offering data warehouse as a service on the cloud [16], so any one of these could also have been taken as a potential candidate for the

example. But, since the authors have already been working on the RES, they thought it is appropriate to continue with the same. Though, the concept presented in this paper can easily be applied to these example too with a little bit of extrapolation. The RES may be accessed by anyone connected to the Internet, through a web browser interface. Authors of this paper assume a simplified approach to RES where following steps are considered for the business process. A use case diagram capturing the functionality of the system is depicted in the Fig. 7 [14]. To illustrate the example, authors are using Unified Modeling Language (UML), but MDA is not restricted to it only.

The characteristics of the actors in the system are:

- Student is the person acting as hub of the system on which the whole RES is running.
- Accountant is the person taking care of all the financial transactions related to the fee deposited by the student.
- Faculty again is the key person of the RES who is teaching and awarding the marks to the student for his/ her grading purpose.
- Clerk is the person who is actually recording all the transactions related to the marks in a centralized database.

Management is a person or a group of persons using this resultant database to perform various types of analysis like student wise analysis, faculty wise analysis, subject wise analysis etc. Management performs these types of analysis to take better decisions to improve the overall performance of the institute which include students, subjects and faculty as a whole.

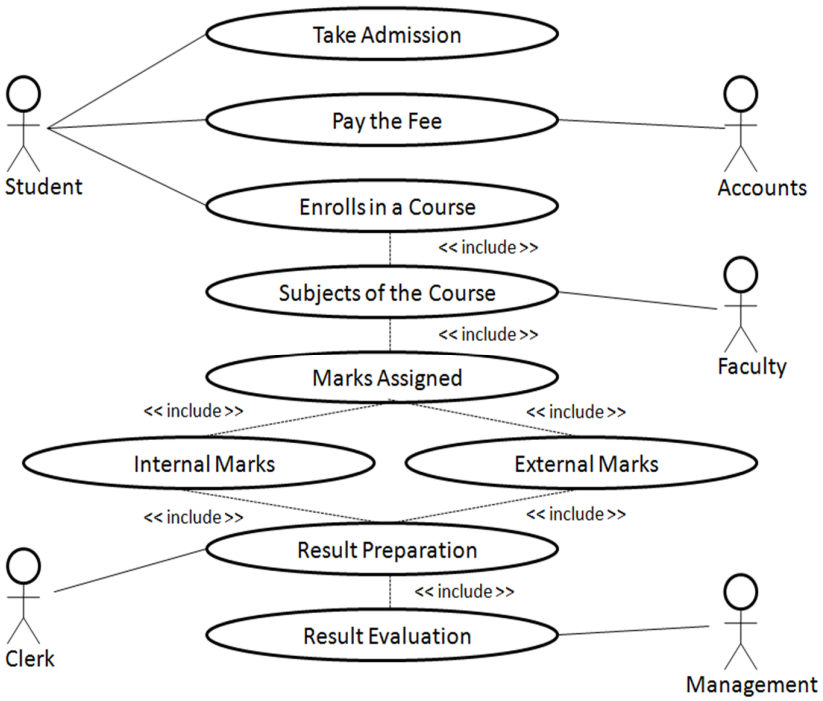


Figure 7.[14] Use Case Diagram for RES

**CLOUD SaaS AND MDA**

An MDA based development of cloud SaaS (application) will enable defining these services in a technology-independent manner and will play a significant role in improving the quality of cloud software services, making them more robust, flexible and agile. Encapsulating business logic in a manner that is independent of the technical mechanisms will formally capture the essence of the applications; and will also make it possible to reuse them in a variety of contexts [15]. Web service is a fundamental technology underlying the cloud computing paradigm; and is evolving too. Based on MDA approach, a formal, semantically rich platform independent model of the Web service capturing the information and functionality provided by it, may be defined which may then be used to generate the artifacts that support the service over some other set of technologies.

Hoyer et al. in [17] describe that the development process starts with the definition of the requirements. The next step is to model the data types and the workflow according to the defined requirements and the available legacy applications. Afterwards, model-driven transformation techniques are applied, generating formal interface descriptions by transforming the workflow modeled, by means of a UML activity diagram into a service model. Finally, a second transformation step is used to generate Web service interfaces in WSDL and corresponding data types in XML Schema. The requirements needed for designing the integration solution can be captured using manifold techniques and there is no consensus on any technique, since only the customer knows what he expects from the final software solution, but cannot express it in an unambiguously and well-formed form. So, author left it to the developer to choose any technique. Based on the defined requirements, the needed data objects are specified. Hence, the desired data objects are modeled as UML class diagrams by using Classes with typed Properties and Associations. Many UML modeling tools support the generation of SQL database schemas from UML class diagrams and vice versa. To generate standardized Web-based interface descriptions and data types, the next step is to transform the model described in [18], which, among other details, specifies the interfaces for each legacy application and the study progress workflow itself. The transformation rules are formalized in the transformation language “Queries, Views, Transformation” (QVT) [19]. The transformation rules are described by mapping the Meta elements of the source meta model to the target meta model [20].

To illustrate the process, we take the example of RES. Like, as we mentioned earlier that the development process starts with the requirement gathering; so in RES the major requirement is to generate the result of the students which is enrolled in a course and that course is being taught by different teachers according to different subjects. Every subject teacher evaluates the student on two different parameters namely internal marks and external marks. At the end, management wants to analyze the performance of students with respect to teacher, course and subject. At the same time management also want to evaluate the performance of every teacher as well. So, after getting the exact requirements, the next step is to design the work flow of the system. Work flow simply explain how the inputs will be received, in which order these inputs will be received and how these inputs will be interrelated with another input level. The complete flow of the sequence is shown in the Fig. 8 [14] which is the graphical representation of the above mentioned concept.

Taking the source model, we use a model-to-model transformation to generate service interfaces. The transformation generates a service interface for each invoked application. So, after defining the workflow of the system, the next step is to transform the work flow model to the service model of SOA as shown in Fig. 9 [14].

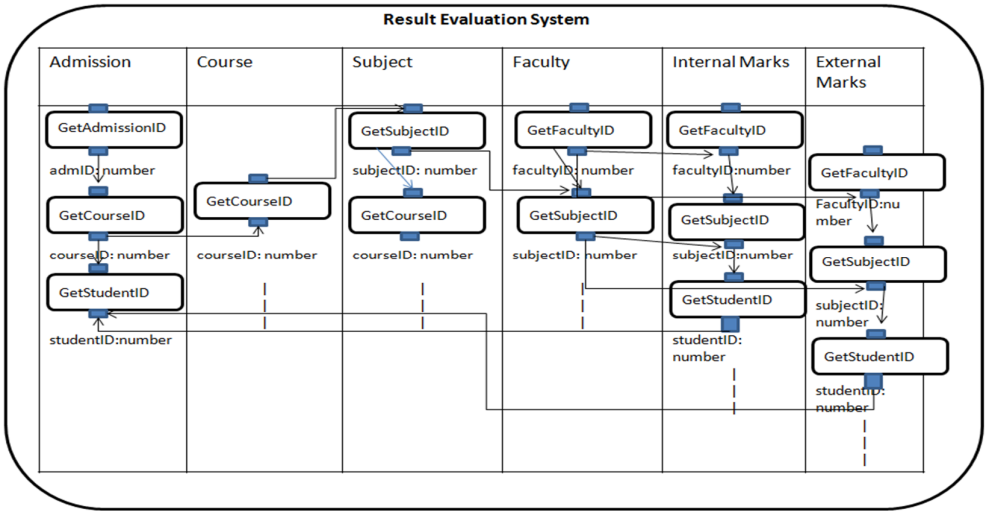


Figure 8.[14] Workflow Diagram of RES

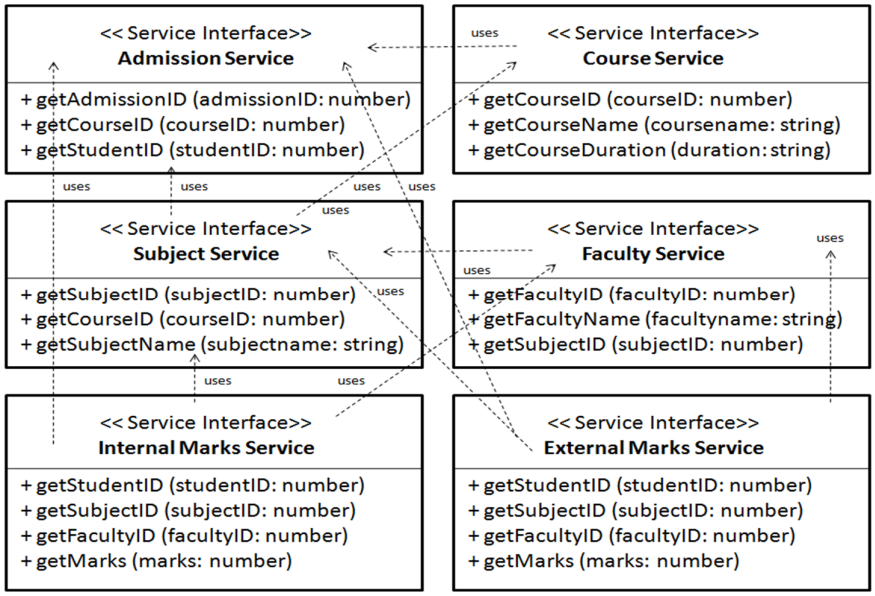


Figure 9.[14] Transformation to Service Model

On the basis of the service interfaces and the data type classes a model-to-text transformation creates WSDL documents (one for each service interface) and one XML Schema document ("ResultEvaluationSystem.xsd"). The available operations of the port types in the WSDL documents match the operations of the service interface. To facilitate the reusability of the XML

Schema definitions the ResultEvaluationSystem.xsd file is imported into the "types" section of each WSDL document. Finally, the generated WSDL documents are used to create base. We implement the adapter logic of the required Web services. The study progress process itself is implemented in the Business Process Execution Language (BPEL) [21] according to the UML Activity. We use an XSL transformation to generate XHTML from the tree map data structure defined before.

**MATERIALIZED VIEWS**

Materialized View is a tool to generate fast and accurate business reports to formulate business intelligence for better decision making process [22]. Materialized Views are most often used in data warehousing and business intelligence applications where querying large fact tables with thousands of millions of rows would result in query response times that resulted in an unusable application. Gupta et al. in [23], highlights the idea of studies the interaction of materialized views with indexes on materialized views which enhances the performance of the result. There are various database management software programs available which support the concept of Materialized Views. But in this paper authors focus on Oracle 10g as an illustration to show the use of MVs. During its creation, Authors specify the SQL used to populate the materialized view. As it is given in Fig. 4 that a fact table with the name Result and user can create a MV that will provide the sum of the Total_Marks data based on the Subject_ID. If a user applies a query the Result table for the sum of the Total_Marls data for a particular subject, DBMS software will direct that query to use materialized view and not towards the Result table. As a result, it will help in reducing the number of accesses against tables which will further help in improving the overall system performance. Following is the example code to create a MV to get the result of a student with respect to each subject:

*create materialized view student_result tablespace student_data*
*refresh complete start with sysdate next sysdate+1*
*enable query rewrite as select subject_ID, sum(Total_Marks) from result group by Subject_ID;*

The redirection of the queries of Result to student_result will be managed by enable query rewrite clause. Location to store data for the MV is explained by tablespace clause and the data will automatically get refreshed because of enabled background job processes. The start with and next clauses defines the schedule of refreshes the data. In the above example, a complete refresh is selected for every time this view is get refreshed and the data of the view will be completely deleted and created again. Fast refreshes of MVs use materialized view logs. A materialized view log is a table which is being stored along with the parent table for the MV. The moment data will be changed in the parent table, the changes will be done to the MV log file. At the time of fast refresh, the updated rows from the parent table, as recognized via the MV log, are sent to the MV. If the changes report for less than 25% of the total data in the parent table, a fast refresh method is usually quicker than a complete refresh method [24]. In general, it is recommended that one should create MVs in the same data base as the tables on which they are based; otherwise, there will be need to manage the permissions and grants required to create and maintain it. Another example for the same is as follows:

*create materialized view result_analysis refresh fast on commit*
*as select sum(Total_Marks), Subject_ID from result group bySsubject_ID;*

## CONCLUSION AND FUTURE SCOPE

The design of a multidimensional schema for a data warehouse is quite a complex task. The authors find out from the available literature that there is no consensus among researchers and practitioners on a common design method for this schema design. Therefore, the authors have suggested adopting an MDA approach for the design of multidimensional model using the star schema. CIM, PIM and PSM models have been designed for the RES (example taken as illustration) and further Star Schema has been generated using MDA approach. Authors also proposed the MDA approach to design services to generate Business Intelligence for the cloud SaaS Model of SOA using a cloud based data warehouse.

An illustrative example taken as a case study has outlined how a service-oriented integration of existing distributed legacy applications can be realized through model-driven development approach. Various RES services and interface models have also been introduced followed by a model for an XML based interaction between cloud SaaS. A workflow diagram of the system under consideration i.e. the RES along with transformation of the artifacts of this workflow diagram into the services of RES have been presented. The model-driven development approach targets a high level of formalization. And therefore supports automatic transformations of models into more concrete models. Finally, the authors have proposed Materialized Views as a tool to generate reports with the motive of fast query processing in order to find out real information which further is intended to be used as BI for better understanding of students' results.

In this research, the process of designing the multidimensional model of the data warehouse using the star schema based on MDA approach is manual. But, since this whole process is full of formalism, the authors suggest the development of a tool for the same as a future work. Using this tool, the multidimensional model of the data warehouse can be developed automatically. Moreover, the proposed research has been carried out using a bottom-up approach for the design. It will also be really interesting to explore the top-down approach for the generation of BI using the concepts of MDA, SOA and cloud SaaS.

## REFERENCES


R. Kimball, *The Data Warehouse ETL Toolkit*, Wiley Publishing, Inc., 2004.

J. D. Poole, "Model-Driven Architecture: Vision, Standards And Emerging Technologies", *Position paper submitted to ECOOP 2001*, http://www.omg.org/mda/mda_files/Model-Driven_Architecture.pdf

Y. Singh, M. Sood, "Models and Transformations in MDA, CICSyN09", *First International Conference on Computational Intelligence, Communication Systems and Networks*, IEEE Computer Society, pp.253-258, 2009.

R. Sharma, M. Sood and D. Sharma, " Modeling Cloud SaaS with SOA and MDA", in *Proc. of the International Conference on Advances in Computing and Communications (ACC 2011), Communications in Computer and Information Science (Springer-Verlag),* 2011, Vol. 190, Part 5, pp511-518, DOI: 10.1007/978-3-642-22709-7_50

W.H. Inmon, *Building the Data Warehouse*, 4th ed., Wiley & Sons, 2005.

Neev Technologies, *"Business Intelligence"*, http://www.neevtech.com/services/business-intelligence

J. Zhou, P. Larson, H.G. Elmongui, *Lazy Maintenance of Materialized Views*, VLDB '07, Sep. 23-28, Vienna, Austria, Copyright 2007 VLDB Endowment, ACM 978-1-59593-649-3/07/09.

A. Sharma and M. Sood, "Exploring Model Driven Architecture Approach to Design Star Schema for a Data Warehouse ", in Proc. of *Advances in Engineering and Technology Series*, Vol. 7, published by Elsevier, ISBN: 978-81-910691-6-3, pp. 466-471, 2013.

OMG 2nd Revised Submission: MOF 2.0 Query / Views / Transformations, http://www.omg.org/cgi-bin/doc?ad/05-03-02

A. Sharma and M. Sood, "Incorporating Model Driven Architecture (MDA) approach to design Multi-dimensional Database from Relational Schema", *International Journal of Computer Engineering and Applications*, Vol. VII, Issue I, July 14.

M. Asadi, M. Ravakhah and R. Ramsin, "An MDA-Based System Development Lifecycle", *Second Asia International Conference on Modeling & Simulation*, IEEE Computer Society, pp. 836-842, 2008.

F. Fleurey, J. Steel and B. Baudry, "Validation in Model Driven Engineering: Testing Model Transformations", in Proceeding of *MoDeVa, Rennes*, France, 2004.

S. Chaudhuri, U. Dayal and V. Narasayya, "*An Overview of Business Intelligence Technology*", Communications of the ACM, Vol 54, No. 8, Pages 88-98, http://cacm.acm.org/magazines/2011/8/114953-an-overview-of-business-intelligence-technology/Fulltext

A. Sharma and M. Sood, "Incorporating MDA to Design Business Intelligence Services by using SaaS Model of Cloud Computing", International Journal of Computer Applications (0975 – 8887) Volume 106 – No. 16, November 2014.

D. Frankel and J. Parodi, "Using MDA to develop Web Services", *IONA Technologies PLC*, 2nd Edition, April, 2002. http://www.iona.com/archwebservice/WSMDA.pdf

A. Ronthal, "*When is the Cloud Appropriate for Data Warehousing and Analytics*", http://bluforcloud.com/2014/cloud-appropriate-data-warehousing-analytics/

P. Hoyer, M. Gebhart, I. Pansa, A. Dikanski and S. Abeck, "Service Oriented Integration Using a Model Driven Approach", *International Journal on Advances in Software*, vol 3 no 1 & 2, year 2010, [Online]. http://www.iariajournals.org/software.

C. Emig, K. Krutz, S. Link, C. Momm and S. Abeck, "Model-Driven Development of SOA Services, Cooperation & Management", *Universität Karlsruhe (TH)*, Internal Research Report, 2008.

Object Management Group (OMG): Meta Object Facility (MOF) 2.0 Query/View/ Transformation Specification Version 1.0. [Online]. http://www.omg.org/spec/QVT/1.0

Object Management Group (OMG): Unified Modeling Language (UML), Superstructure Version 2.2. [Online]. http://www.omg.org/cgibin/doc?formal/09-02-02

Organization for the Advancement of Structured Information Standards (OASIS): Web Services Business Process Execution Language Version 2.0. [Online] http://docs.oasisopen.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html

A. Sharma and M. Sood, "Utilizing Materialized Views to formulate Business Intelligence", in *International Conference on Parallel, Distributed and Grid Computing (PDGC 2014)* at JUIT, Solan (HP), India, Dec.11-13, 2014.

H. Gupta, V. Harinarayan, A. Rajaramana and J. Ullman, *Index Selection for OLAP*, ICDE 1997.

Oracle Press, "*Database Data Warehousing Guide: Basic Materialize Views"*, http://docs.oracle.com/cd/E11882_01/server.112/e25554/basicmv.htm#DWHSG8181.