

Design and Application of the Hibernate Persistence Layer Data Report System using JasperReports

Bhushan S. Sapre, Rohan V. Thakare, Santosh V. Kakade, Dr. B. B. Mesh ram

Abstract - The contents of this paper mainly focus on study of the JasperReports, an open-source reporting tool, for realization and implementation of dynamic and reusable report generation system. iReport is a GUI based report designer assists the realization of the reports. The separation of the report template from report data provided by JasperReports has provided ease of development and maintenance of reports in overall system. This data report system can be used to generate reports dynamically and improve overall system's reusability and maintainability.

Index Terms – Hibernate, iReport, JasperReports, SpringMVC.

I. INTRODUCTION

Report is a self-explanatory statement of facts relating to a specific subject and serves the purpose of providing information for decision making and follow up actions. Report is an important tool in project management, and its purpose is to help managers quickly grasp the raw data of the basic elements and relationships, this is useful for making effective management decision [10]. The traditional generation method of reports is relatively curable, once the system development was completed; all reports' formats are fixed. Therefore it is not possible to change format of report according to changes in circumstances and cannot customize them of personal fashion. When content or format of the report changes, we need to modify the code constantly to adapt changes which restricts progress of the project. Therefore we generally use XML documents to give the definition, using the idea of the separation of report format and report data. Format module of reports extract XML format files from the template library of report format, data module of reports extract relevant data from the database, finally, the report was compiled and displayed by reports builder[1]. However, this kind of reports needs to develop oriented database, resulting in reliance on the data source and reducing reuse and portability of the reports. This paper focuses on Jasper Reports, an open source reporting tool along with a GUI based report designer, iReport. It is combined with Hibernate to create dynamic and reusable report generation system. In project which based on open-source framework of SpringMVC, we used this report system to generate reports and invoices and successfully constructed a stable and efficient system.

II. THE CHOICE, ANALYSIS AND APPLICATION OF THE REPORT BUILDER

A. Working of Jasper Reports

With the development of open source software, especially Linux OS, open source software being gradually accepted and respected by the people, it played an increasingly important role in IT. Jasper Reports is an open source report tool which written in pure Java, it can run on any Java-application environment, including J2EE and other web applications [10]. Jasper Reports has a good cross platform and compatibility. It can dynamically display reports of various styles and formats. It also can display rich content of the page and convert these pages into XLS, RTF, PDF, HTML and XML format, showing its applicability. Moreover, iReport as a production tool of front-end interface, it has made a considerable improvement by providing facility to create charts, sub-tables, meet schedules, and other more complex reports [3].

Figure 1 describes the process of report generation by Jasper Reports. Report generation starts with JRXML file which is also called as Report template. JRXML file contains the report structure information. JRXML file parsed by JRXmlLoader and converted into Jasper Design Object. Jasper Design Object contains only design information of the report. After compiling Jasper Design Object using JRCompiler it gets converted into Jasper Report without Data. Now Data filled into empty Jasper Report Object using RS (Result Set) and it will be converted into Jasper Print Object. From Jasper Print Object you have two options: JasperPrintManager from which you can print the report or view the report on the screen else Jasper Export Manager from which you can export report in HTML, PDF, XLS format.

B. iReport: Visual Report Designer Studio

Process of report generation starts from JRXML file. This JRXML can be written manually but it is complex and time consuming. Moreover we need to learn how to structure JRXML file as per syntax and semantics of Jasper Reports. Therefore we choose iReport, a front page showing and producing tool that matches with Jasper Reports [11].

It is mainly used in the visual design and generation of JRXML file in statements form used by Jasper Reports, as Jasper Reports itself did not provide a good visual statements design tools, the appearance of iReport has just make up for the deficiencies. At the same time, iReport can call the Jasper Reports library and apply it to any kind of Java application process. iReport is distributed, and described by pure Java which

correspondence with GNU General Public License. iReport provides a relatively perfect platform for visual editing of report template, using it we just need a simple operation to design the demand interface. It can graphically edit JRXML documents and support for PDF, RTF, XML, XLS, CSV, HTML and many other formats, and it also can support for the realization of XML data sources. The relationship between the content and performance of report is configured by JRXML file; this can bind different domains and different databases, so that reports can be applied under different data source environment.

spent with manual data handling in SQL and JDBC[21]. Convert the result set from the table format into the value of an object, and achieve the operation of cross-database platform [5]. Hibernate relieves the developer from of common data persistence related programming tasks. Hibernate makes use of persistent objects commonly called as POJO (Plain Old Java Object) along with XML mapping documents for persisting objects to the database layer. The term POJO refers to a normal Java objects that does not serve any other special role or implement any special interfaces of any of the Java frameworks (EJB, JDBC, DAO, JDO, etc...).

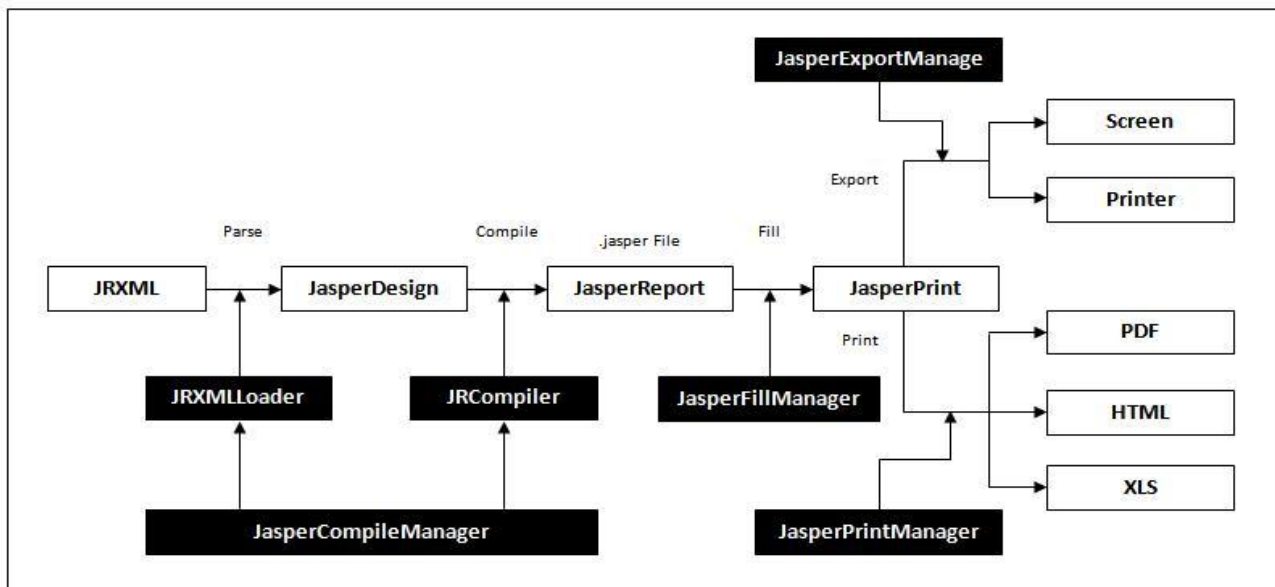


Figure 1: Process of generating reports by JasperReports

iReport also compiles JRXML file to form ‘JASPER’ file which is used to fill data from data source. This method provides separation between template of the report and data of the report i.e. report template is independent of the data displayed by the report. It avoids complexity and is perfect for rapid production of the reports.

III. USE OF HIBERNATE TO GENERATE REPORTS

A. Introduction to Hibernate

Hibernate is the most popular open source ORM (object/relational mapping) framework at present. Hibernate lets you develop persistent classes following common Java idiom - including association, inheritance, polymorphism, composition and the Java collections framework. Its goal is to become complete solution to solve the management of persistent data in Java. The primary role of Hibernate is to be an integral plan that it can manage the persistent object in java. Hibernate not only takes care of the mapping from Java classes to database tables (and from Java data types to SQL data types), but also provides data query and retrieval facilities and can significantly reduce development time otherwise

Rather than utilize byte code processing or code generation, Hibernate uses runtime reflection to determine the persistent properties of a class. The objects to be persisted are defined in a mapping document, which serves to describe the persistent fields and associations, as well as any subclasses or proxies of the persistent object. The mapping documents are compiled at application startup time and provide the framework with necessary information for a class. Additionally, they are used in support operations, such as generating the database schema or creating stub Java source files[21].

Hibernate is built on top of many Java APIs and other Java open source frameworks/APIs. The architecture is layered to keep you isolated from having to know the underlying APIs. In fact, while the available Hibernate API is large, the API used by most application developers is relatively small. One only needs to know fewer than a dozen Hibernate classes to use Hibernate.

The Configuration object is the first Hibernate object you use. It represents a configuration or properties file for Hibernate. The Configuration object is usually created once during application initialization. The Configuration object reads and establishes the properties Hibernate uses to get connected to a database and configure itself for

work. A Configuration object is used to create a Session object and then typically is discarded.

The Session Factory is created from a Configuration object, and as its name implies it is a factory for Session objects. The Session Factory is an expensive object to create. It, like the Configuration object, is usually created during application start up. However, unlike the Configuration object, It should be created once and kept for later use. The Session Factory object is used by all the threads of an application. It is a thread safe object. One Session Factory object is created per database. Multiple Session Factory objects (each requiring a separate Configuration) are created when connecting to multiple databases. The Session Factory can also provide caching of persistent objects.

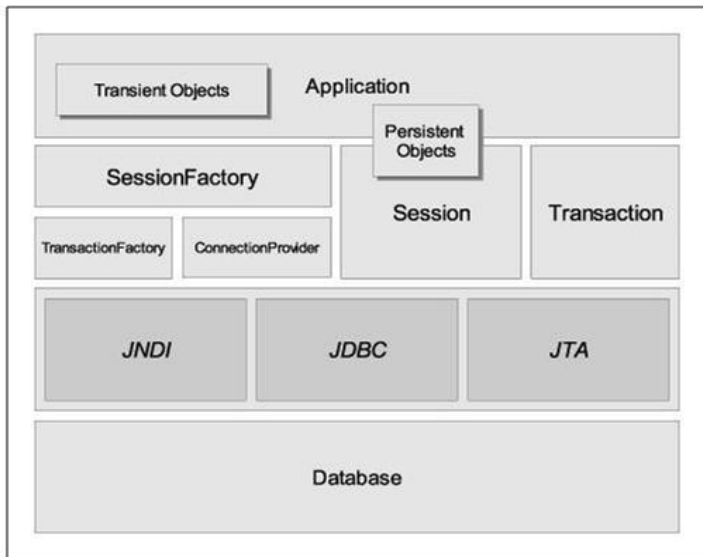


Figure 2: Hibernate Architecture

Session objects provide the main interface to accomplish work with the database. Persistent objects are saved and retrieved through a Session object. A Session object is lightweight and inexpensive to create. A Session object does the work of getting a physical connection to the database. Session objects maintain a cache for a single application thread (request). Session objects are not thread safe. Therefore, session objects should not be kept open for a long time. Applications create and destroy these as needed. Typically, they are created to complete a single unit of work, but may span many units. When modifications are made to the database, Session objects are used to create a Transaction object. A Transaction represents a unit of work with the database (and potentially other systems). Transactions in Hibernate are handled by an underlying transaction manager and transaction (from JDBC, JTA or CORBA). Hibernate with use whatever transaction implementation is available (JDBC, JTA, CORBA, etc.). The Hibernate Transaction object abstracts the developer from having to deal with the underlying transaction manager/transaction. This allows developers to use a Transaction Manager of their choice without having to code to a specific Transaction

Manager. Transaction objects should be kept open for as short of time as possible. Query and Criteria objects are used to retrieve (and recreate) persistent objects. Query objects use SQL or Hibernate Query Language (HQL) string to retrieve data from the database and create objects. Criteria uses an object/method based means of constructing and executing a request to retrieve objects.

B. Using Hibernate with Jasper Reports

Hibernate is one of the most popular ORM tools in use at the moment. Using Hibernate as a data source for Jasper Reports can be very simple when a collection of objects is returned from a Hibernate query, but when a tuple of objects is returned then a custom JRDataSource implementation is required. When a Hibernate query returns a collection of objects, a `net.sf.jasperreports.engine.data.JRBeanCollectionDataSource` can be used to map the Hibernate POJO instance fields to the report fields. All that's required for this simple solution is to use the `JRBeanCollectionDataSource` (`java.util.Collection beanCollection`) constructor, passing it the Hibernate Query result set. `JRBeanCollectionDataSource` implements `JRDataSource` like `StaticDataSource` but its `getFieldValue(JRField jrField)` method implementation maps the report template field names to the query result bean properties.

When a Hibernate query returns a tuple of objects it's necessary to write a custom implementation of the `JRDataSource` similar to `HibernateDataSource`. The implementation of the required `next()` method in this class returns true if there is another list item in the Hibernate query result set, while putting the current list item in a current Value holder for use in the `getFieldValue(JRField)` method. The `getFieldValue()` method implementation gets the field index in the current Value object via a call to the `getFieldIndex(String field)` method. This method iterates through the mapped field names passed to the `HibernateDataSource` constructor until it finds the field name it was passed and then returns the index of this field in the current Value information. The `getFieldValue()` method then returns the value at this index in the current Value result object.

IV. SYSTEM ARCHITECTURE AND ITS ADVANTAGES

A. Process of Report Generation

The system uses architecture of SpringMVC which is an open-source framework. Function modules of system can be divided into: user management module, system management module, project management module, schedule management module, query and statistics report module. SpringMVC framework is one of the best MVC frameworks by far, SpringMVC has a flexible view of expression and can support for report output of Jasper Reports, so that it can greatly simplify and speed up report development [8]. System accesses data through SpringMVC and Hibernate, besides SpringMVC will

monitor the request of spring affairs. Business layer uses construction function of spring, we use it to manage the data source and the establishment and recovery of database connection pool. Data persistent layer integrates Hibernate in the SpringMVC framework. This has realized a changement from a relational query to an object-oriented query, and Hibernate largely optimize a query process. Chart of system framework and process of Jasper Reports generating report shown in Figure 3.

Figure 3 is divided into module 1, 2 and 3. Module 1 consists of Web application, SpringMVC framework and Jasper Reports Engine. Module 2 contains Data Access Object (DAO), JDBC Connection Pool and Data server. Module 3 contains iReport visual report designer. Module 1 has responsibility to accept request from client and pass it to module 2. Module 1 basically forms Controller part of the MVC framework. Web application accepts HTTP requests from client passes it to the Spring framework. Spring framework has a controller implementation for each type of request. By analyzing the request and parameters packed with the request, an appropriate call is made to the Jasper Reports Engine. Jasper Reports Engine is provided with DAO and path of .JRXML file. Compilation of the .JRXML file is responsibility of Jasper Reports Engine. Jasper Reports Engine makes call to database through DAO. Module 2 achieves data access and model package: integrate the Hibernate which is the open source O/R relational mapping tools and bases on Spring framework, package the relationship data that

instance of JavaBeans type which is object-oriented, then return to the Jasper Reports to generate report view engine. Module 3 provides report templates in the form of .JRXML file generated through iReport to Jasper Reports Engine.

B. Advantages of Data Report System

From Figure 3 we can see that Jasper Reports in the SpringMVC framework has two modules to generate a report: module 2 and module 3. If we want to develop different reports, only have to package the data in the module 2 into different JavaBean data model, such a development process well reflects parallel design pattern of the MVC, it can greatly improve the development efficiency of report. Module 3 realizes style design of reports by iReport, that's easy to design reports according to business needs, then enabling the division and parallel development of the data and view [10].

Data query module is extremely important modules in the project management system; in this module we use the approach that the object-oriented data model and Jasper Reports report driver and iReport designer. In the Jasper Reports we use iReport to dynamically design report according to actual needs, resulting in high speed report generation. In the system database layer, we use Hibernate framework to realize lightweight package on JDBC. In internal of data processing layer, system achieves the separation between the logic of business and the logic of data [9]. With needs of business, we only have to replace the Hibernate mapping files, greatly

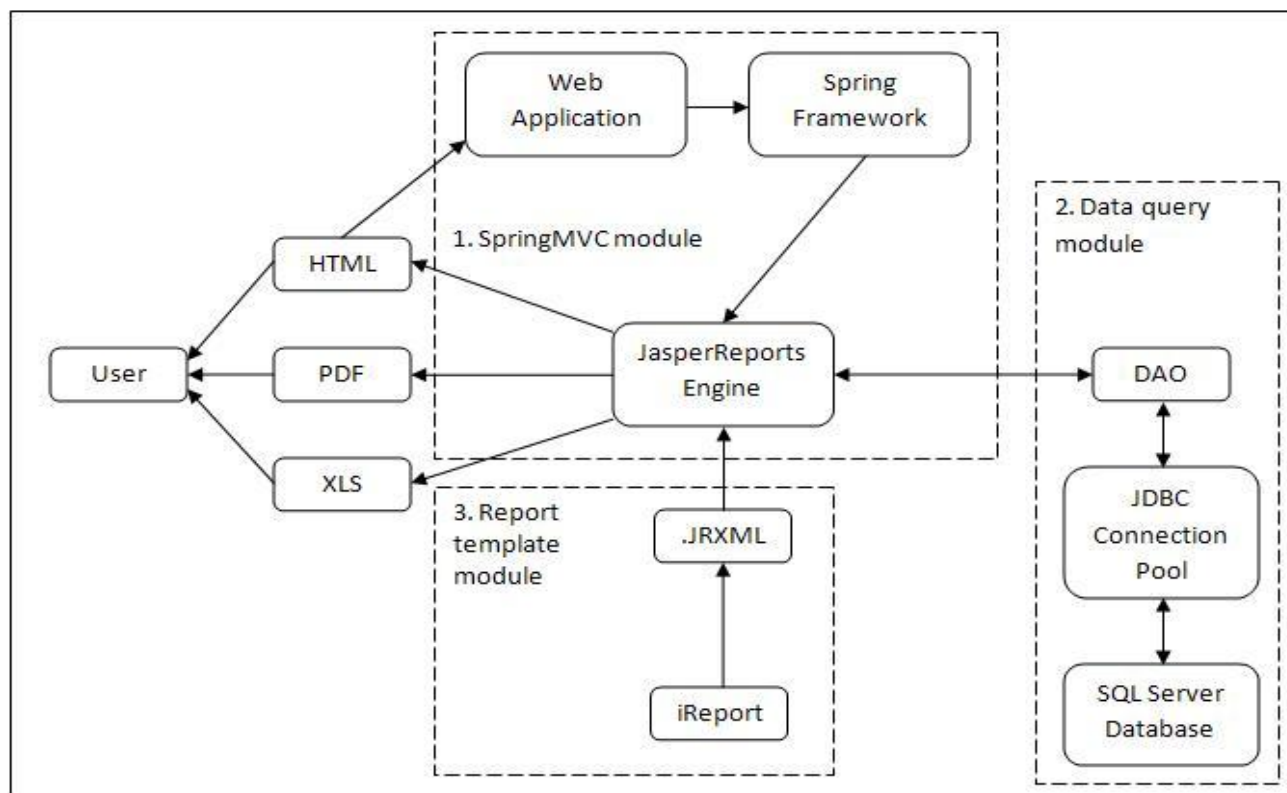


Figure 3: Flow of report generation based on Hibernate using JasperReports

obtained by JDBC connection pool into the object

increasing reusability of code. With query optimization of

Hibernate itself, greatly improving efficiency of whole system and saving more development time. These reports not only fully meet the needs of the project management system, but also a good time to save programming time [10].

C. Future Enhancements

As Jasper Reports is written in pure Java, it is preferred with SpringMVC combination. As compatibility of other reporting tools enhances, reporting tools like Pentaho, BIRT can also be used with spring and Hibernate in future.

V. CONCLUSION

This paper mainly focuses on an open-source reporting tool Jasper Reports, visual report designer iReport and Hibernate. This data report system separates report template from report data. This helps to achieve flexibility in reports. The report template can be changed irrespective of the report data displayed by report. Currently this idea has been applied to the open-source SpringMVC framework. This data report system has helped to meet business needs of the project and has provided a fast and scalable data report system.

REFERENCES

- [1] Rong Li, Yuewei Ding, Xiaojun, Xu, "Dynamic Data Report Generation Technology based on XML", Micro-computer information, vol.12-3, pp.87-89, 2005.
- [2] Lei Liu, Hong Liu, Huayu Wang, "The research and implementation of reuse report generation system based on Component", Min-microcomputer system, vol.24-4, pp.739-744, 2003.
- [3] Official website of JasperReports. <http://jasperreports.sourceforge.Net/>, 2002.
- [4] Wenhao Cui, "The research and application of reports generation using Jasper Reports", Journal of Shenyang chemical Institute, vol.22-2, pp.176-179, 2008.
- [5] Feng Zhang, Hong Wang, Jiao Zhu, "Know-how of Java Advanced", Beijing, Science Press, 2008.
- [6] Reporting Made Easy with Jasper Reports and Hibernate <http://xml.ultizer.com/node/171467>, 2007.
- [7] Using Jasper Reports with Hibernate <https://www.hibernate.org/79.html>, 2007.
- [8] Weiqing Sun, "Java Web design and development based on MVC", Beijing, Electronic Industry Press, 2004.
- [9] Li Feng, Wensheng Li, "The design and implementation of J2EE-based object-oriented data report system", Computer system application, vol.2, pp.57-58, 2006.
- [10] Fangxia Zhang, Renjin Zhang, "The Research and Application of JasperReports in Project Management System", OSSC-2009, pp 56-59.
- [11] Ming Xue, Changjun Zhu, "Design and Implementation of the Hibernate Persistence Layer Data ReportSystem Based on J2EE", pp 232-235.
- [12] Official website of JasperForge, <http://jasperforge.org/>, 2009.
- [13] Importance of Reporting, <https://www.msu.edu>, 2010.
- [14] Reporting Software – Comparisons & Reviews of Reporting Tools, <http://www.reportingsoftware.info>, 2010.
- [15] Understanding Meaning of Report <http://kalyan-city.blogspot.in/>, 2010.
- [16] <http://www.thoughtsdecoded.com/>, 2011.
- [17] Open-source reporting tool comparison www.slideshare.net/OpenLogic/open-source-reporting-tool-comparison, 2011.
- [18] Open Source BI vs Closed Source BI, <http://blog.jinfonet.com/58/>, 2008.
- [19] Reporting Made Easy with JasperReports and Hibernate, <http://java.sys-con.com/node/171467>, 2006.
- [20] Hibernate Tutorial, <http://www.allapllabs.com/>, 2007.
- [21] Hibernate Architecture, <http://www.hibernate-training-guide.com/>, 2008.

AUTHOR BIOGRAPHY



Bhushan S. Sapre was born in Mumbai on 17th February 1988. He completed his Bachelor's degree in Computer Engineering from Lokmanya Tilak College of Engg., Navi Mumbai, and Maharashtra, India. Currently he is pursuing his Master's degree in Computer Engineering from V.J.T.I (Veermata Jijabai Technological Institute), Mumbai, India. He is member of Computer Society of India. He has published paper on 'Fault Tolerant Environment Using Hardware Failure Detection, Roll Forward Recovery Approach and Micro rebooting For Distributed Systems' in 'International Journal of Engineering Research and Applications' in 2011.



Rohan V. Thakare was born in Nagpur on 19-Dec-1986. He received BE degree (Computer Science & Engineering) from Vishwakarma Institute of Technology, Pune, and Maharashtra, India in 2008. He is pursuing M.Tech degree in Computer Technology [Specialization in Network] from V.J.T.I (Veermata Jijabai Technological Institute), Mumbai, India. Mr. Rohan is a member of Computer Society of India and his paper "Enhancing Mobile Phone Security" in International Journal of Internet Computing is to be published.



Santosh V. Kakade was born in Chandrapur on 20-May-1986. he received BE degree (Computer Science & engineering) from K.K. WAGH Institute of Engineering, Nashik, Maharashtra, India in 2009. He is persuing M.Tech degree in Computer Science & Engineering from V.J.T.I (Veermata Jijabai Technological Institute), Mumbai, India. Mr. Santosh is a member of Computer Society of India and his paper on "Security and Intrusion Detection in Mobile Ad Hoc Network" in IJJC (IFRSA's International journal of computing) has published.



Dr. B. B. Meshram is currently professor and Head of Computer Technology Department of Veermata Jijabai Technological Institute (VJTI), Matunga, Mumbai (INDIA). His areas of interest include Object oriented database management systems, Computer network security and multimedia systems. He has published more than 110 papers in National & International Conferences & refereed Journals.