Developing Independent Data Mart and Analysis Tools as Standalone Application

Rully Agus Hendrawan
Department of Information System, Faculty of Information Technology, Institut Teknologi Sepuluh Nopember

Keywords:
Data warehouse
Independent data mart
Standalone deployment
Mobile business intelligence

ABSTRACT

Since organizations are forced to make fast decision, the urgency of data becomes significant. Data must be maintained in a way that analyst could use it easily. On the other side, even when the data is already easy to use, analysts still have to understand the data and translates database structures into business terms before they can start working.

This paper proposes architecture of data mart application considering two perspectives: data and software. There are three situations to use this system: urgent data and important analysis, short-term strategy to complement long term, and data mart as pragmatic solution. A case study explained as example implementation of the proposed architecture.

The result of implementation is standalone software to deliver independent data mart and common analysis tools as one package. The package distributed to end-user (analyst). The software should not tightly couple with data mart. When advanced analysis is required, data mart inside the package must be able to be used immediately with other analysis tools.

Corresponding Author:

Rully Agus Hendrawan,
Department of Information System, Faculty of Information Technology,
Institut Teknologi Sepuluh Nopember,
Jalan Raya Kampus ITS, Gedung Sistem Informasi, Sukolilo, Surabaya, Indonesia.
Email: eraha@is.its.ac.id

1. INTRODUCTION

Every organization needs a good data management. Data are very important. They are raw materials that can be processed into valuable information to help making a wise decision. Since organizations are forced to make fast decision, the urgency of data becomes significant. Data must be delivered to analyst quickly. Data must also be processed and prepared properly so that the analyst can utilize it easily. The most endorsed method to solve this situation is by building data warehouse and/or data mart. Data mart has special advantage. It is easy to build especially with limited IS resources.

Data mart is being read by different kind of applications. It is being processed for analysis or generating reports. Some general/common analysis and report could be performed to the data mart, for example: multi-dimensional analysis (OLAP) and reporting. If those basic needs could be fulfilled without third party application, it would bring several benefits. Analyst does not need to understand the data structure to start working. Analyst does not need to connect or do additional processing to use the data mart in his analysis / reporting software.

This research provides the reason, architecture, and case study for the development of standalone software to deliver independent data mart and common analysis tools as one package. The package distributed to end-user (analyst), so that they can start their work immediately. The package should not tightly couple with data mart. When advanced analysis is required, data mart inside the package must be able to be used immediately with other analysis tools. The method presented in this paper has been validated in a real case study concerning Fishery Resources in East Java Indonesia.

2. ADVANTAGE OF INDEPENDENT DATA MART

Data mart started from the intention to separate data from transactional processing with analytical processing. Independent data mart is most likely focused exclusively on one area. One or more operational systems were being used as the data source. Analysis or reporting tools query it directly, delivering information to analyst [1]. Data mart caters specific needs. It only performs very well in specific conditions. Here are some considerations with philosophical approach to determine when to use independent data mart.

2.1. Urgent data and important analysis

The main consideration is the needs and capability of the organization itself. Data must be given to analyst quickly and make immediate impact. Mostly, data mart is considered as inexpensive alternative to a data warehouse. It takes significantly less time and lower cost to build [2][3][4]. Data mart usually use minimal IS resources and often leads to very fast implementation [5].

Compared to data warehouse that are better-architected decision support system, independent data marts are less expensive. Furthermore, constructing independent data marts do not require the organization to understand their data beyond individual departments. Data warehouse creates a complex problem [6], that was expensive, time-consuming, and expert-intensive [7].

The main point of creating data mart is that analyst can do their own work so that they do not have to integrate with other data marts [8]. The problem only rise in integrated environment, when there are multiple independent data marts that are dependent to each other [9]. Data marts are intended for limited area implementation and have no enterprise context [1]. Data mart also solves the situation where we need to work in isolated environment without any network connection. We could also use data mart to isolate and protect production data [10].

2.2. Short-term strategy to complement long term

Independent data marts are a poor architectural solution based on information quality, system quality, individual impacts, and organizational impacts [11][8]. They may achieve rapid and inexpensive results in the short term; but they can give rise to long-term costs and inefficiencies [1]. The data mart vendors are selling a very short-term perspective at the expense of long-term architectural success [9].

However, data mart is not a substitute of data warehouse. When we want to go to certain places, we can go by bicycle or car depends on which one we could afford. Data warehouse is not an upgrade of data mart either, we cannot upgrade bicycle to car. We can own both bicycle and car. Developing data mart for short-term solution, while working to build warehouse in long term, is also an alternative. Data mart should be designed accommodating the long-term data warehouse strategy [12].

2.3. Data mart as pragmatic solution

Another opinion said that data warehouse could be built from data mart [13][14]. We can find balance between the ideal way to do things, and a pragmatic approach. Pragmatic might be ideal when it comes to value individual initiative and results is important [15]. Reach the target first, and refine later.

3. PIVOT TABLE AS BASIC ANALYSIS TECHNOLOGY

It was said that Pito Salas was the "father" of pivot tables [16]. Pivot table has solved many analytical needs, it can summarize and analyze data faster and transform gigantic data sets into crystal-clear summary reports [17]. Many vendors provide PivotTable to enhance the functionality of their data processing software [18][19]. Software library for pivot table are also available in many programming languages [20][21][22][23].

There are three kinds of data warehouse applications: information processing, analytical processing, data mining [24]. Table X shows the types of application, usage and existing solution for the application of data warehouse / data mart. Most of the information and analytical processing can be solved using Pivot Table and Chart. Some advanced analytical processing such as spatial and geographical analysis, data mining and data visualization require either static / dynamic-link library, custom tailored implementation, or feeding the exported data to more sophisticated analysis tools.

Table 1. Utilizing Data Warehouse / Mart with Analysis Technology

Type of Application	Usage	Technology
Information processing	Reporting (crosstabs, tables, charts, graph)	Pivot Table, Pivot Chart
Analytical processing	Multidimensional Analysis (OLAP)	Pivot Table, Pivot Chart
	Geographical Analysis	GIS, Data visualization library / custom tailored
Data mining	Various Methods	Embedded data mining library
Other / advanced needs	Various Methods	Export to sophisticated analysis tools

4. THE PROPOSED ARCHITECTURE

The architecture is constructed and derived from both data and software perspectives. Within data management and engineering perspective, there are two considerations. The first is independent data mart architecture [8][1][5]. The second one is kinds of data warehouse application: informational processing, analytical processing, and data mining. From the software engineering perspective, the architecture is also being constructed by considering rich client application archetype and stand-alone deployment [25]. The full architecture is shown in figure 1.

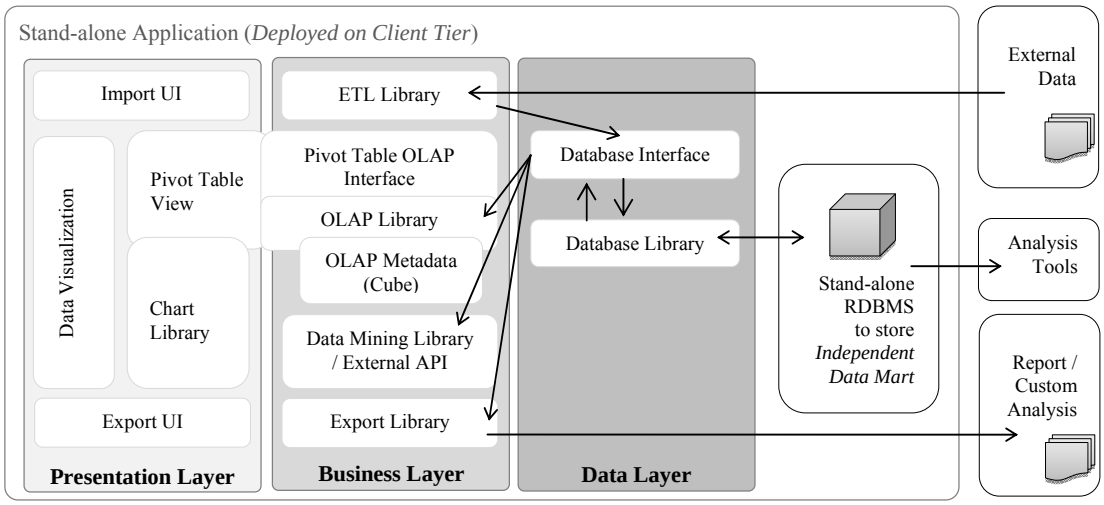

Figure 1. Architecture of Stand Alone Data Mart Analysis Tools

First, external data sources are being read by ETL library and populated to stand-alone RDBMS using database interface and library. Other analysis tools using direct database connection can access data mart. Data mart can also be exported and used for report archive, or as input for other analysis tools.

OLAP Library is accessing data mart via database interface and library. OLAP Library calculates the aggregation and summarization. It use the OLAP Metadata to perform the basic OLAP operation (slide, dice, roll up, and drill down). OLAP metadata or cube definition can be hard-coded in business layer or stored inside the stand-alone RDBMS. OLAP Library provides the aggregate data to Pivot Table OLAP Interface, to be shown on Pivot Table View. Utilizing Chart Library, Pivot Table can also become Pivot Chart.

Data mining library access the data mart and perform a specific knowledge discovery process. Data visualization can get data from data mining output, Pivot Table, OLAP, or directly from data mart to represent data in a more meaningful way.

5. CASE STUDY

Many software development methodologies can be used to implement the architecture [26]. In this case study, we will use 4D methodology, popular software development manifesto [27], to explain the process. 4D stands for Define, Design, Develop and Deploy. Inside this methodology, we also incorporate the development of independent data mart.

The major steps in building a data mart are designing, constructing, populating, accessing, and managing [28]. Data mart designing were done on "design" phase. Data mart constructing, populating and accessing were done in "develop" phase. Data mart managing, such as backup and restore can be done on "deploy" phase. We will not fully explain how we develop the system. We highlight some of the process that directly related with implementation of the architecture.

5.1. Define

In this phase, we define what and how the application should work. We determine the business objectives and goals. We acquire and interpret business requirements. Then, we write them as software requirement specification. We evaluate the requirement to ensure that the purpose can exploit the advantage of independent data mart.

The summary of the requirements covers three parts: data gathering, data maintenance and data analysis. Data are being gathered from various sources especially from survey, because data collection is expensive. The data collection and cleaning will be done manually. The data must be collected in one place, as data mart. Only one department will maintain the data. Other stakeholders only require using the data in read-only mode. They use the data in the same platform, for reporting or analysis. The system must provide some basic reporting (table, chart) and analysis tools (OLAP). Special data visualization is required to aid spatial analysis of fishery resources. There is no requirement yet for advanced analytics, reporting or data mining. Advanced analysis might be done by third party analysis tools, thus the data should be able to be accessed directly or indirectly by using export mechanism.

5.2. Design

This phase consist of two types of work: data design and software design. In designing data model, it's very important requirement to make the data structured in a way that analyst could easily understand and use. This is the main reason why dimensional modeling technique was being chosen. It ensure simple data structure and user query [13]. The multidimensional model is great for flexibility of analysis but is not optimal for large amounts of data [8]. In software design, we are not going to explain the logic / process but focused on the software architectural design. The main works are discovering and deciding which technologies that are going to be used on each of the architecture component, based on the requirement. Table 2 shows the summary of chosen implementation method.

The case study used Embarcadero Delphi as an integrated development environment (IDE) due to its flexibility for rapid application development (RAD) [29]. SQLite was picked as the RDBMS because it can be implemented as embedded database [30]. SQLite is lightweight and powerful enough to store data mart that contains aggregated or summarized data [9]. User interface, chart and database interface was being implemented with native library [31][32]. Proprietary library was being used to implement pivot table [23], and the OLAP metadata was hardcoded inside business layer.

ETL mechanism was implemented using native library. Aside from scripted ETL, manual data modification should be added on the Import UI. This could also cause aggregation issue [33]. There is a special need in data visualization to analyze the geographical trends of fishery resources. This requirement could be solved using proprietary GIS library, but the cost is not feasible. Custom-coded implementation from scratch using native API is more favorable. Export mechanism to CSV and popular office would be used for special ad-hoc needs. CSV text is the easiest format for data visualization [34][35]. Office documents such as spreadsheets are valuable to those who need to make sense of quantitative business data [36].

Table 2. Implementation Plan for Each Architecture Component

Layer	Components	Implementation
Presentation	Import UI	Native [31]
	Data Visualization	*Custom Tailored*
	Chart Library	Native [31]
	Pivot Table View	Proprietary Library [23]
	Export UI	Native [31]
Business	ETL Library	*Custom Tailored*
	Pivot Table OLAP Interface	Proprietary Library [23]
	OLAP Metadata	*Custom Tailored*
	Data Mining Library	N/A
	Export Library	Proprietary [23] and Open Source Library [35]
Data	Database Interface	Native [32]
	Stand-alone RDBMS	Open Source [30]

5.3. Development

Data mart construction and population were being done in this phase. OLAP schema is also being constructed. This is heavyweight-lifting phase for programming. Several functional and non-functional requirements are being implemented. The adapted architecture based on requirements and design is shown in figure 2.

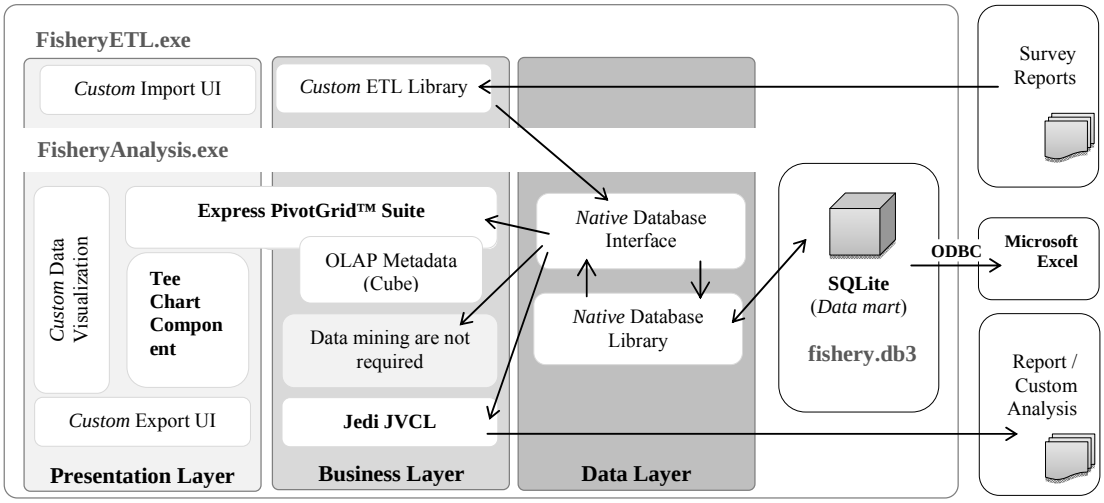

Figure 2. Architecture of Stand Alone Data Mart Analysis Tools

5.4. Deployment

As shown in Figure 3, the deployment consists of three files. The first file is fishery.db3, which is the independent data mart, stored as SQLite 3 database. FisheryAnalysis.exe is the main analysis tool. The ETL module was separated as FisheryETL.exe, because there is no access control being implemented. SQLite library is embedded in both applications. The result is highly portable, stand-alone analysis tool and independent data mart.

Managing data mart is very simple. Backup can be done by copying the .db3 file to other location. Restoring data mart can be done by replacing the .db3 file with backup file.

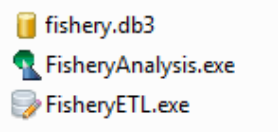

Figure 3. Deployment of Standalone Analysis Tool and Data Mart

6. CONCLUSION

With resource constraint, independent data mart that typically low cost and fast to implement is a good option. Data mart is good to fill the short-term requirement, although it cannot replace data warehouse. Independent data mart should also be considered as pragmatic solution to help reaching immediate result. It is also a preferred solution for isolated environment. Pivot table solves many common reporting and analysis need. It is the swiss army knife for analytical needs and must be included in the architecture.

Architecture is constructed from data and software perspective. The system is deployed on client tier as stand-alone application. The data mart was stored in stand-alone RDBMS. Case study has shown a successful application of the architecture. Using 4D methodology, the architecture was implemented using proprietary, open source and custom-tailored library. The result is ultra-portable standalone software to deliver independent data mart and common analysis tools as one package.

However, there are still some issues in this architecture. I still does not cover access control list. The security is bare minimum. It currently only work best on non-sensitive data. In the future, it has potential to be developed further as portable stand-alone Business Intelligence.

REFERENCES

[1] Christopher Adamson. Star Schema: The Complete Reference™. McGraw-Hill; 2010.
[2] Alex Berson, and Stephen J. Smith. Data Warehousing, Data Mining, and OLAP. New York: McGraw-Hill; 1997.
[3] Srinivasan Kanchi. Selecting the Right Architectures for Successful Data Warehouses. Tata Consultancy Services. 2005.
[4] Hugh J. Watson, Thilini Ariyachandra. Data Warehouse Architectures: Factors in the Selection Decision and the Success of the Architectures. [Publisher unknown]. 2005 Jul.
[5] Chuck Ballard. Data Modeling Techniques for Data Warehousing. IBM Redbooks; 1998.
[6] Inmon,W.H. Building the Data Warehouse. 3rd Ed. New York: John Wiley & Sons; 2002.
[7] Joseph Guerra, David Andrews. Why You Need a Data Warehouse. Cheshire: Andrews Consulting Group; 2011.

[8] T. Ariyachandra, H. J. Watson. Which Data Warehouse Architecture is Most Successful. Business Intelligence Journal. 2006; 11(1):4-6.
[9] William Inmon. Data Mart Does Not Equal Data Warehouse. DM Review Magazine. 1998 May.
[10] Tom Mochal. Get IT Done: Create stand-alone data marts to get end users off your live production data [Internet]. [Place unknown]: TechRepublic; 2003 [cited 2013 Jun 01]. Available from: http://goo.gl/QRgdT7
[11] T. Ariyachandra, H. J. Watson. Key Factor in Selecting a Data Warehouse Architecture. Business Intelligence Journal. 2005; 10(2):19-26.
[12] Angela Bonifati, Fabiano Cattaneo, Stefano Ceri□, □Alfonso Fuggetta Stefano Paraboschi. Designing Data Marts for Data Warehouses. ACM Transactions on Software Engineering and Methodology. 2001 Oct; Vol. 10(4):452–483.
[13] Ralph Kimball, Margy Ross. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling. Wiley; 2013.
[14] Ralph Kimball, Margy Ross, Warren Thornthwaite, Joy Mundy, Bob Becker. The Data Warehouse Lifecycle Toolkit. Wiley; 2008.
[15] James Standen. Pragmatic Business Intelligence. Datamartist; 2009 May [cited 2013]. Available from: http://www.datamartist.com/pragmatic-business-intelligence
[16] Bill Jelen, Michael Alexander. Pivot Table Data Crunching. Indianapolis: Que Publishing; 2006.
[17] Bill Jelen, Michael Alexander. Excel 2013 Pivot Table Data Crunching (MrExcel Library). Que Publishing; 2013.
[18] Google. How to Create a Pivot Table Report. Google. 2013 [cited 2013 Jun 01]. Available from: https://support.google.com/drive/answer/1272900?hl=en
[19] Microsoft. Overview of PivotTable and PivotChart reports. Microsoft. 2011 [cited 2013 Jun 01]. Available from: http://goo.gl/QNO2MX
[20] ZK Framework. ZK Pivottable [Internet]. ZK Framework; 2013 [cited 2013 Jun 01]. Available from: http://www.zkoss.org/product/zkpivottable
[21] Jonathan Jackson, Robert Jackson. Pivot.js [Internet]. [Publisher unknown]; 2013 Mar [cited 2013 Jun 01]. Available from: https://github.com/rjackson/pivot.js
[22] Flexmonster. Pivot Table & Charts Component v1.9 Documentation [Internet]. 2013 [cited 2013 Jun 01]. Available from: http://www.flexmonster.com/pivot-table-component-docs
[23] Developer Express. ExpressPivotGrid™ Suite. Developer Express. 2013 [cited 2013 Jun 01]. Available from: https://www.devexpress.com/Products/VCL/ExPivotGrid/
[24] Jiawei Han, Micheline Kamber, Jian Pei. Data Mining: Concepts and Techniques: Concepts and Techniques. 3rd ed. Waltham: Elsevier; 2011.
[25] Microsoft Patterns & Practices Team. Microsoft® Application Architecture Guide. O'Reilly Media; 2009.
[26] Dubberly H. How do you design. A Compendium of Models [Internet]. San Francisco: Dubberly Design Office; 2008 [cited 2013 Jun]. Available from: http://goo.gl/IIT2xy
[27] MyProgrammer. The 4D Methodology: Our Time-Tested Process [Internet]. MyProgrammer; [Date unknown] [cited 2013 Jun]. Available from: http://www.myprogrammer.com/how-we-work/
[28] Alejandro Butinof. The Oracle Data Mart Suite Cookbook. Release 2.6. Oracle Corporation; 1999.
[29] Embarcadero Technologies. RAD Studio XE4 Feature Matrix. Embarcadero Technologies. 2013 [cited: 2013 Jun]. Available from: http://www.embarcadero.com/products/delphi/delphi-feature-matrix
[30] Kreibich, Jay. Using SQLite. O'Reilly; 2010.
[31] Chris Rolliston. Delphi XE2 Foundations Paperback. CreateSpace Independent Publishing Platform; 2012.
[32] Bob Swart. Delphi XE3 Starter Essentials. Bob Swart Training & Consultancy; 2012.
[33] Li, J. Han, Z. Yin, J.-G. Lee, Y. Sun. Sampling Cube: A Framework for Statistical OLAP over Sampling Data. SIGMOD. 2008
[34] Ben Fry. Visualizing Data: Exploring and Explaining Data with the Processing Environment. O'Reilly Media; 2008.
[35] Project JEDI. JVCL Help: TJvCsvDataSet Class. Project JEDI; 2012 [cited on 2013]. Available from: http://wiki.delphi-jedi.org/wiki/JVCL_Help:TJvCsvDataSet
[36] Stephen Few. Now You See It: Simple Visualization Techniques for Quantitative Analysis. Analytics Press; 2009.

BIBLIOGRAPHY OF AUTHORS

Rully Agus Hendrawan has been involved with Data Warehousing and building Business Intelligence since 2006. He also has developed various applications using different software architecture archetype. He received his master degree in Information Science and Engineering from Ritsumeikan University. He teaches Database Management, Data Warehouse and Business Intelligence in Institut Teknologi Sepuluh Nopember, department of Information System.