

**2012**

**University of North Carolina Wilmington**  
**Master of Science in**  
**Computer Science and Information Systems**  
**Proceedings**

**<https://csbapp.uncw.edu/mscsis>**

**AN ANALYSIS, DESIGN, AND IMPLEMENTATION OF A BUSINESS  
INTELLIGENCE SOLUTION FOR POWERPLAY MOBILE**

Patrick Green

A Capstone Project Submitted to the  
University of North Carolina Wilmington in Partial Fulfillment  
of the Requirements for the Degree of  
Master of Science

Department of Computer Science  
Department of Information Systems and Operations Management

University of North Carolina Wilmington

2012

Approved by

Advisory Committee

---

Dr. Jeff Brown

---

Dr. Bryan Reinicke

---

Dr. Ron Vetter, Chair

## **Abstract**

An Analysis, Design, and Implementation of a Business Intelligence Solution for PowerPlay Mobile, Green, Patrick, 2012. Capstone Paper, University of North Carolina Wilmington.

This paper focuses on an evaluation of several different business intelligence solutions, both open source and proprietary, in order to select the best candidate for implementation of a reporting services application. The evaluation criteria are based off the needs of a mobile marketing services company, PowerPlay Mobile, to be used by their customers. PowerPlay Mobile's customers are merchants who wish to connect with their customers via texting and other mobile services. This reporting services application allows these merchants to better understand the behaviors of their subscribers, in order to optimize PowerPlay Mobile's services.

The paper follows the evaluation process step-by-step, concluding with a beta implementation of a reporting services application. Upon migration of data to the data warehouse, the application will be made available to customers through the Internet.

## Table of Contents

	Page
Section 1: Introduction.....	6
Section 2: Literature Review and Analysis.....	9
2.1: Literature Review .....	9
2.2: BI Solutions Review .....	11
Section 3: Methodology.....	14
Section 4: Results and Lessons Learned.....	22
Section 5: Conclusions and Future Work.....	30
References.....	31
Appendices:	
Appendix A: Screen Shots of BIRT Reports.....	32
Appendix B: Microsoft SQL Server Stored Procedures .....	37
Appendix C: BIRT Implementation Source Code .....	42
Appendix D: PowerPlay Mobile Transaction Database.....	49
Appendix E: Developer’s Guide .....	53

## List of Figures

Figure 1 .....	7
Figure 2 .....	10
Figure 3 .....	14
Figure 4 .....	16
Figure 5 .....	23
Figure 6 .....	24
Figure 7 .....	24
Figure 8 .....	25
Figure 9 .....	25
Figure 10 .....	26

## List of Tables

Table 1.....	13
Table 2.....	18
Table 3.....	18
Table 4.....	19
Table 5.....	19
Table 6.....	20
Table 7.....	20

## **Section 1: Introduction**

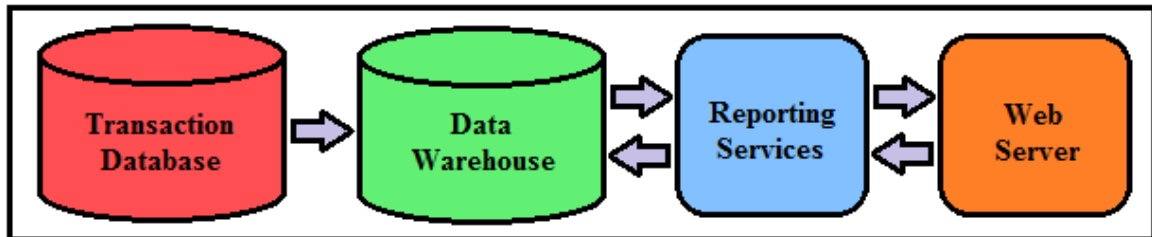
PowerPlay Mobile required a Business Intelligence (BI) solution to provide feedback to their customers. The company provides mobile services for merchants who wish to reach consumers via their cell phone. This includes sending coupons and other information in the form of text messages to subscribers. PowerPlay Mobile wanted to provide a layer of reporting services on top of their transactional data in order to give their customers a better understanding of the behavior of their subscribers.

Before PowerPlay Mobile can develop and recommend a BI solution for reporting to their customers, an analysis of existing BI solutions had to be conducted. This analysis was performed on several open source solutions and one proprietary solution. Several factors were evaluated during the analysis, including security, ease of use, web accessibility, built-in functionality, and others.

PowerPlay Mobile also needed a solution that allowed their customers to log into the reporting services interface via a web browser over the Internet. The interface had to be secure and allow customers to generate custom reports based on variable input. The solution should also allow for reports to be built ad hoc to PowerPlay Mobile's specifications. PowerPlay Mobile had a limited budget for maintaining and expanding its reporting services, so the solution needs to include a report designer that is intuitive and easy to use.

Transactional data for PowerPlay Mobile is stored in a MySQL database located on a Linux server. The BI solution implemented needed to be compatible with a MySQL database created from a MySQL dump of this transactional data, or a common data export format such as XML. This exported data, in whatever form, serves as PowerPlay

Mobile's data warehouse on which the reporting services are built. *Figure 1* below demonstrates a basic configuration of how the reporting services are setup.



*Figure 1 – A basic configuration for a web enabled reporting services application.*

Aside from the analysis of existing BI solutions, PowerPlay Mobile was also prepared to make changes to the structure of their relational database in order to optimize their reporting capabilities. A discussion of how this may be achieved is covered in more detail later in the paper.

The proprietary BI solution considered for this analysis was Microsoft Reporting Services (2008 R2) [10]. The open source solutions are Business Intelligence and Reporting Tools (BIRT 3.7.1) [4], JasperReports Server (4.5.0) [6], Pentaho Reporting (4.2.1) [9], JavaEye Reporting Tool (JERT 0.3) [7], and Art Reporting Tool (ART 2.1) [2]. This paper provides an analysis and evaluation of these solutions, and describes the results of beta implementations of two of these products. Finally, the paper recommends one solution for implementation of PowerPlay Mobile's reporting services application.

A few terms are used in this paper, so we define them here.

- *Business Intelligence* refers to the reporting and analysis of electronically stored data in order to provide benefit to business decision-making.
- *Data Warehouse* refers to a structure of data used by business intelligence systems

for analyzing and report generating.

- *Transactional Data/Database* refers to data undergoing asynchronous changes over time as significant events take place.
- *Data Integration* refers to changing combined data into a useful format for analysis.
- *Web Application Archive (WAR)* is a JAR file used to deploy a web application.
- *Strengths, Weaknesses, Opportunities, and Threats (SWOT) analysis* is a two by two matrix which organizes features and limitations of a product.

The rest of the paper is organized as follows. Section two gives a review of the literature concerning Business Intelligence software, as well as covers how it is implemented and utilized. The third section discusses the methodologies used to approach the issues and challenges mentioned in the introduction. Section four provides results and lessons learned while implementing two beta solutions, followed by conclusions and future work in section five.

## Section 2: Literature Review and Analysis

### 2.1 Literature Review

Business Intelligence (BI) is an emergent field in the IT industry, and has become a priority for many companies in recent years. BI broadly refers to the computer-based techniques for reporting and analyzing data, in order to improve business decision-making [12]. Its goal is to allow for quick and convenient access to meaningful information, presented in a useful manner. This gives the business a greater understanding of its operational and strategic management; insight it would otherwise not have. It can also discover previously unforeseen patterns which could be unknowingly impacting the business [5].

The usefulness of BI greatly depends on the quality and quantity of the available data. This insight into business data can be useful, not only for the business itself, but sometimes their customers as well. When the customers of a business are businesses themselves, BI can offer ways of optimizing the services being offered. This gives customers much more incentive to continue these services.

There are two forms of data that can be analyzed by BI systems, structured and unstructured. Structured data refers to data stored in relational databases, whereas unstructured data refers to text stored in articles, documents, web pages, etc. Generally, about 20% of available data is structured, and 80% is unstructured [8]. Ad hoc queries are commonly used to make use of the structured data in BI systems. These queries are then embedded in reports, so that they can be dynamically generated based on user input. An example of what these generated reports can look like is demonstrated in *Figure 2* below, using the open source BI solution: Jaspersoft.



Figure 2 – An example of a generated report from Jaspersoft.

It is important that these ad hoc reports be flexible and intuitive enough to allow for information self-service by their users. For this, developers must work closely with the business users to best understand their needs from the reporting system. Many of these systems fail to serve their businesses in useful ways because of this lack of collaboration [3].

Reporting is not commonly generated against live, transactional data. Instead, a data warehouse is produced from the transactional data so that the reports can be generated quicker and without interruption to the transaction database. Often, production of a data warehouse requires data integration, due to the 80% of unstructured data previously mentioned. It also may be required on the structured data as well, if the format being using by the reporting service is different than that of the relational database of the transactional data.

One major factor that must be considered when implementing a BI system is scalability. As the data warehouse grows, so will the workload for reporting.

Representation of the data, and optimization of queries, is key in maintaining an efficient reporting services application. Evaluating hardware specifications to allow the workload to scale up is also vital during the designing and development phase of any BI reporting system [11]. Before deciding on hardware, developers need to take into consideration that, unlike transactional data which tends to focus on individual records, BI systems need to be able to frequently process mass quantities of data in a reasonable amount of time.

## ***2.2 BI Solutions Review***

A preliminary analysis of the solutions chosen to be evaluated demonstrates that some of the features desired by PowerPlay Mobile are common and readily available. Others, such as prepackaged security, are not. The following subsection discusses the preliminary analysis of these BI solutions.

Business Intelligence and Reporting Tools (BIRT) is an Eclipse open source plug-in for web-based reporting applications. It includes an Eclipse view for report design, and a run-time component for web application servers. BIRT supports many different types of data representation, including graphs, diagrams, and charts. It can query data from any database with a JDBC connector, an XML file, and even a Flat File. BIRT supports dynamic parameters given by the user to customize a report. Its latest release was February 2012. BIRT has a large support community, which includes forums, FAQs, and tutorials.

JasperReports Server is a rich open source web-based reporting server that can query data from any relational database with a JDBC connector, an XML file, and a Flat File. It supports authorization/authentication security for its web server, and dynamic parameter inputs from the user. JasperReports offers a wide variety of data representation, including graphs, diagrams, and charts. Its latest release was January 2012 and offers online support in the form of community forums, FAQs, tutorials, and examples.

Pentaho Reporting offers an open source reporting service which can query data from any relational database using the JDBC connector, or from an XML file. It offers a report design wizard for creating new reports, it can easily be embedded in any web applications server, and it supports dynamic parameters given by the user. Its latest release was in September 2011, and offers a wide variety of support, such as community forums and FAQs.

JavaEye Reporting Tool (JERT) is a lightweight open source reporting tool which can easily be deployed on any web server using its web application archive (WAR) package. By default, it uses a JDBC driver to connect to any SQL database where a JDBC connector is available. It stores reports in an XML format, and supports dynamic parameters for report queries. However, JERT does not support graphical data representation, such as graphs and diagrams. There is no report designer available for this solution. Its last release was in March 2005.

Art Reporting Tool (ART) is another lightweight open source reporting tool which can be deployed to a web server using its WAR package. It can connect to any SQL database where a JDBC connector is available. It offers dynamic parameter reporting, as well as graphical data representation. This solution does not offer a report designer to

create new reports. Its latest release was in November 2011.

Microsoft Reporting Services is a rich proprietary reporting server which can connect to any known data sources. It includes a user friendly report designer, built-in web server with authentication/authorization security, and support for graphical data representation, such as charts, diagrams, and graphs. Microsoft Reporting allows for dynamic parameters based on user input. The edition being evaluated here is Microsoft Reporting Services 2008 R2. Microsoft offers a large amount of support, including forums, tutorials, real-time assistance, FAQs, etc.

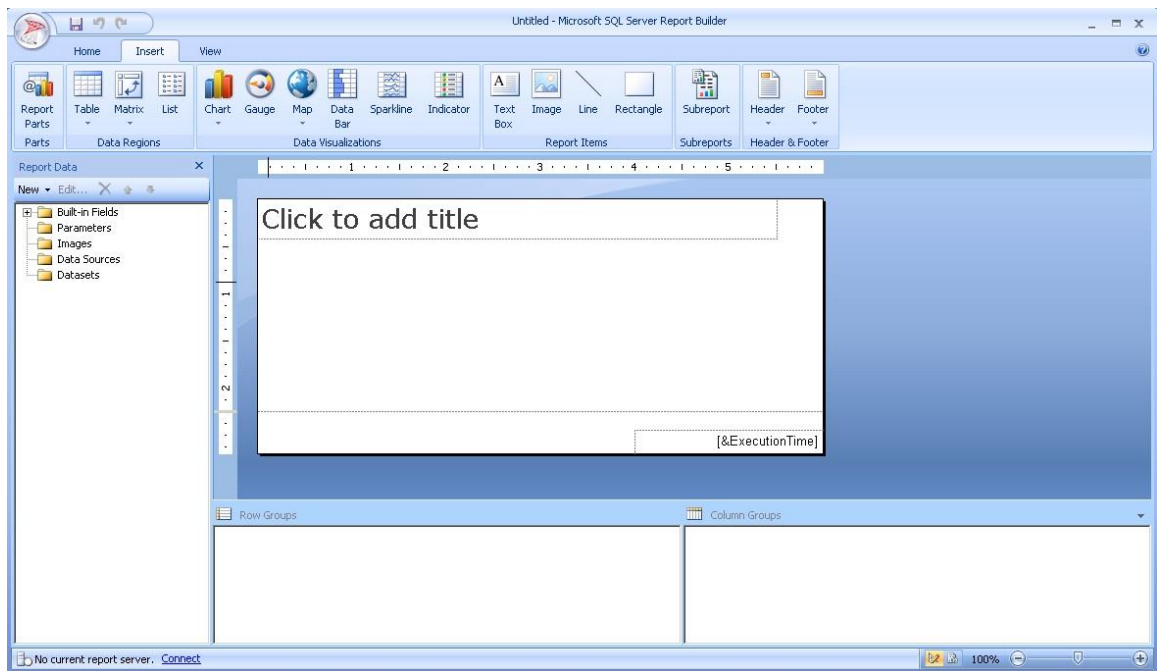
The descriptions above are just preliminary evaluations of the functionality desired by PowerPlay mobile. Below is a summary of features offered by these solutions for comparison.

*Table 1 – A comparison matrix of features for each BI solution being evaluated.*

Features→ Products↓	Dynamic Parameters	Ongoing Development	Web Server	Graphical Representation	Document- ation	Report Builder	Security
Microsoft Reporting Services	Yes	Yes	Included	Yes	Yes	Yes	Included
BIRT	Yes	Yes	Not Included	Yes	Yes	Yes	Not Included
Jasper-Reports Server	Yes	Yes	Included	Yes	Yes	Yes	Included
Pentaho Reporting	Yes	Yes	Not Included	Yes	Yes	Yes	Not Included
JERT	Yes	No	Not Included	No	No	No	Not Included
ART	Yes	Yes	Not Included	Yes	Yes	No	Not Included

### Section 3: Methodology

The method of analysis of these BI solutions will be a comparison of their strengths and weaknesses, as they are relevant to the needs of PowerPlay Mobile. As previously mentioned, PowerPlay Mobile needs a product that is easy to learn, maintain, and expand. This means finding a product that has an intuitive design interface for building new ad hoc reports without requiring intimate knowledge of the product. An applicable example of this is shown in *Figure 3* below of Microsoft SQL Server Report Builder (3.0).



*Figure 3 – A screenshot of Microsoft SQL Server Report Builder.*

The design interface also needs to be powerful enough to represent data in various ways. PowerPlay Mobile does not want to be restricted in their data representation by the limitations of the BI solution. Ideally, the report design tool should offer a variety of

different charts, graphs, tables, and diagrams with several styles to make the reports more visually appealing.

Another major factor that was evaluated is the presence and functionality of a provided web server. PowerPlay Mobile wishes to have a web server with built-in security and account management. Customers need to log-in in order to view the reporting services interface. It will be convenient if this functionality already exists within the provided web interface; otherwise, it will have to be built and integrated in-house.

Data integration was also considered and evaluated in this analysis. The BI solution needs to be compatible with an available format for the data warehouse. The most useful compatibilities are with the MySQL Open Database Connectivity (ODBC) driver, the MySQL Java Database Connectivity (JDBC) driver, or XML. Performance will most likely be best with one of the MySQL connectivity drivers, since the data will not have to be parsed before use.

As mentioned previously, compatibility with Linux OS is desired. Because the transactional data is processed on a Linux machine, data integration and reporting works much more seamlessly if done in the same environment. This also gives PowerPlay Mobile the option of placing everything on one server, which can increase performance and ease of use.

Another desired attribute is support provided by the BI solution. The solution should have useful documentation, FAQ's, tutorials, reporting examples, community forums, etc. The more support provided by a solution, the lower the learning curve, and thus the costs of maintenance and expansion. Also to be taken into account is if the

product itself is still being updated and maintained.

Pricing of the BI solution will not effect its evaluation. PowerPlay Mobile is associated with the University of North Carolina Wilmington. Because of the Microsoft Developer Network Academic Alliance (MSDNAA), PowerPlay Mobile is not required to pay additional licensing fees to use Microsoft Reporting Services. Since this is the only proprietary solution, cost of licensing will not be an issue for implementation of this project. However, should this solution be chosen, PowerPlay Mobile will need to acquire an appropriate license for commercial use.

A Strengths, Weaknesses, Opportunities, and Threats (SWOT) analysis can be used to determine which products should be considered for further evaluation. A SWOT analysis is used to identify internal and external aspects of a product which can help demonstrate which products are best suited for implementation of this project. The analysis includes a two by two matrix of information for each product, in order to organize important information about each for comparison. *Figure 4* demonstrates this matrix.

	Positive	Negative
Internal	Strengths	Weaknesses
External	Opportunities	Threats

*Figure 4 – A SWOT analysis matrix.*

The strengths section contains features giving the product an advantage that is useful to the project. The weaknesses section lists limitations or shortcomings of the product in reference to the project. The opportunities section describes the external features of a product that can lead to improvements. Finally, the threats section of the matrix lists any external features of the product that can lead to disruption of the project, or cause trouble.

One of the major advantages to this analysis is the organization of data. The format makes it simple to review and understand a product's features and limitations. However, this format can be misleading, since each feature or limitation at first glance appears to hold the same weight. This is obviously not true, since a product can have many strengths, but includes a single essential weakness that makes it unworthy of implementation. Ultimately, these matrixes are used to help PowerPlay Mobile make a decision in narrowing down its search for a product.

*Table 2* shows the SWOT analysis for Microsoft Reporting Services. *Table 3* shows the SWOT analysis for BIRT. *Table 4* shows the SWOT analysis for JasperReports Server. *Table 5* shows the SWOT analysis for Pentaho Reporting. *Table 6* shows the SWOT analysis for JERT. Finally, *Table 7* shows the SWOT analysis for ART.

Table 2 – A SWOT analysis of Microsoft Reporting Services.

	Positive	Negative
Internal	<p><b>Strengths:</b></p> <ul style="list-style-type: none"> <li>• Includes a report builder</li> <li>• Includes a pre-configured web server</li> <li>• Supports dynamic parameters for reports</li> <li>• Includes graphical data representation</li> <li>• Can connect to MySQL database using an ODBC driver</li> <li>• Reports are easy to deploy</li> </ul>	<p><b>Weaknesses:</b></p> <ul style="list-style-type: none"> <li>• Requires linking a MySQL database to a Microsoft SQL Server</li> <li>• Requires using stored procedures to utilize report parameters</li> <li>• Not compatible with Linux OS</li> </ul>
External	<p><b>Opportunities:</b></p> <ul style="list-style-type: none"> <li>• Various outlets of support available</li> <li>• Well maintained and developed</li> </ul>	<p><b>Threats:</b></p> <ul style="list-style-type: none"> <li>• Requires the purchase of a license for commercial use</li> </ul>

Table 3 – A SWOT analysis of BIRT.

	Positive	Negative
Internal	<p><b>Strengths:</b></p> <ul style="list-style-type: none"> <li>• Includes a report builder</li> <li>• Supports dynamic parameters for reports</li> <li>• Includes graphical data representation</li> <li>• Compatible with Linux OS</li> <li>• Can connect to MySQL database using an JDBC driver</li> <li>• Simple to configure on various web servers</li> <li>• Reports are easy to deploy</li> </ul>	<p><b>Weaknesses:</b></p> <ul style="list-style-type: none"> <li>• Does not include a web server</li> </ul>
External	<p><b>Opportunities:</b></p> <ul style="list-style-type: none"> <li>• Open-source</li> <li>• Online support available</li> <li>• Currently maintained and developed</li> </ul>	<p><b>Threats:</b></p> <ul style="list-style-type: none"> <li>• Better solutions may come along</li> </ul>

Table 4 – A SWOT analysis of JasperReports Server.

	Positive	Negative
Internal	<p><b>Strengths:</b></p> <ul style="list-style-type: none"> <li>• Includes a report builder</li> <li>• Supports dynamic parameters for reports</li> <li>• Includes graphical data representation</li> <li>• Compatible with Linux OS</li> <li>• Can connect to MySQL database using an JDBC driver</li> <li>• Reports are easy to deploy</li> <li>• Includes a pre-configured web server</li> </ul>	<p><b>Weaknesses:</b></p> <ul style="list-style-type: none"> <li>• Loaded solution; many features would never be used</li> </ul>
External	<p><b>Opportunities:</b></p> <ul style="list-style-type: none"> <li>• Open-source</li> <li>• Online support available</li> <li>• Currently being maintained and developed</li> </ul>	<p><b>Threats:</b></p> <ul style="list-style-type: none"> <li>• Better solutions may come along</li> </ul>

Table 5 – A SWOT analysis of Pentaho Reporting.

	Positive	Negative
Internal	<p><b>Strengths:</b></p> <ul style="list-style-type: none"> <li>• Includes a report builder</li> <li>• Supports dynamic parameters for reports</li> <li>• Includes graphical data representation</li> <li>• Compatible with Linux OS</li> <li>• Can connect to MySQL database using an JDBC driver</li> <li>• Offers a web server plugin</li> <li>• Reports are easy to deploy</li> </ul>	<p><b>Weaknesses:</b></p> <ul style="list-style-type: none"> <li>• Does not include web server</li> </ul>
External	<p><b>Opportunities:</b></p> <ul style="list-style-type: none"> <li>• Open-source</li> <li>• Online support available</li> <li>• Currently being maintained and developed</li> </ul>	<p><b>Threats:</b></p> <ul style="list-style-type: none"> <li>• Better solutions may come along</li> </ul>

Table 6 – A SWOT analysis of JERT.

	Positive	Negative
Internal	<p><b>Strengths:</b></p> <ul style="list-style-type: none"> <li>• Lightweight solution</li> <li>• Simple to configure on various web servers</li> <li>• Reports are easy to deploy</li> <li>• Supports dynamic parameters for reports</li> <li>• Compatible with Linux OS</li> <li>• Can connect to MySQL database using a JDBC driver</li> </ul>	<p><b>Weaknesses:</b></p> <ul style="list-style-type: none"> <li>• Does not include a report builder</li> <li>• Does not include a web server</li> <li>• No graphical data representation is offered</li> <li>• Limited online support</li> </ul>
External	<p><b>Opportunities:</b></p> <ul style="list-style-type: none"> <li>• Open-source</li> </ul>	<p><b>Threats:</b></p> <ul style="list-style-type: none"> <li>• No longer being developed and maintained</li> <li>• Last release was in 2005</li> <li>• Better solutions may come along</li> </ul>

Table 7 – A SWOT analysis of ART.

	Positive	Negative
Internal	<p><b>Strengths:</b></p> <ul style="list-style-type: none"> <li>• Lightweight solution</li> <li>• Simple to configure on various web servers</li> <li>• Reports are easy to deploy</li> <li>• Supports dynamic parameters for reports</li> <li>• Includes graphical data representation</li> <li>• Compatible with Linux OS</li> <li>• Can connect to MySQL database using a JDBC driver</li> <li>• New versions are compatible with JasperReports' iReport report builder</li> </ul>	<p><b>Weaknesses:</b></p> <ul style="list-style-type: none"> <li>• Does not include a report builder</li> <li>• Does not include a web server</li> </ul>
External	<p><b>Opportunities:</b></p> <ul style="list-style-type: none"> <li>• Open-source</li> <li>• Online support available</li> <li>• Currently being maintained and developed</li> </ul>	<p><b>Threats:</b></p> <ul style="list-style-type: none"> <li>• Better solutions may come along</li> </ul>

After each product was evaluated on the chosen criteria, two products were selected for further evaluation. The two chosen products were both fully implemented (in beta) to generate a few of the reports needed by PowerPlay Mobile. The effectiveness of the two products was then compared so that one could be chosen for final implementation by PowerPlay Mobile.

After evaluation and selection of the final BI solution, a beta implementation was built for PowerPlay Mobile to begin their reporting services application. This gives them a baseline start for which they can begin to expand as needed.

PowerPlay Mobile also wanted to evaluate the current structure of their transaction database as it relates to reporting. Until now, the database primarily served as a log of events. There was not a good use for the database to exhibit a great deal of relationships between data. This makes report generation a little more difficult, and less efficient. PowerPlay Mobile would like an analysis of their current structure, and how it can be transitioned to optimize report generation.

## Section 4: Results and Lessons Learned

Based on the detailed SWOT analysis (see previous section) and after discussions with PowerPlay Mobile stakeholders, it was decided to take a closer look at BIRT and Microsoft Reporting Services. For further evaluation of these two products, seven reports were chosen to be implemented: current subscribers, message count by date range, new subscribers by date range, opt-outs by date range, a message log by date range, text offers sent by date range, and subscriber growth by date range. These reports can be seen in *Appendix A*.

The Microsoft Reporting Services reports were deployed on the web server included in the solution. The BIRT reports were deployed on an Apache Tomcat web server integrated with the BIRT Viewer [1]. Both report builders are very similar in design and functionality. Both solutions can produce very similar reports. There are some disadvantages to Microsoft Reporting Services when using it to report from a MySQL database. The MySQL database must be linked to a Microsoft SQL Server. In order to utilize dynamic report parameters, Microsoft Reporting Services requires using stored procedures if the report is using a MySQL database as its data source. BIRT does not have the restrictions of Microsoft Reporting Services, it is free to use, and it can run on Linux OS. Because of these factors, PowerPlay mobile decided to use BIRT for implementation of the reporting application for their clients.

The BIRT report design is easy to learn and use. Shown in *Figure 5*, the BIRT report builder looks similar to Microsoft Report Builder seen previously in the paper. The design is fairly intuitive and reasonably organized.

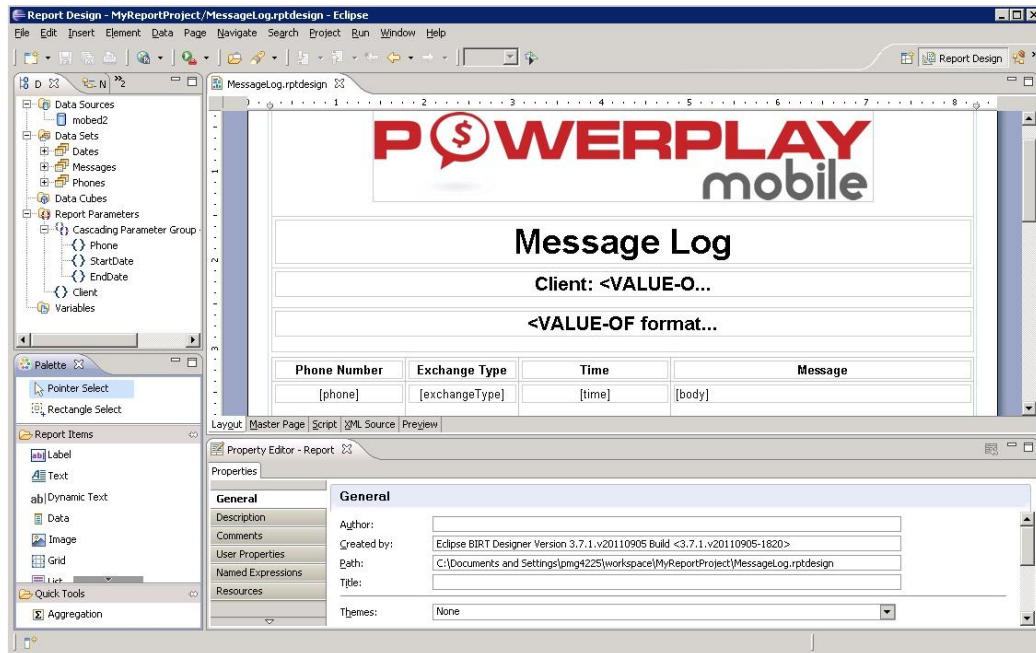


Figure 5 – A screenshot of the BIRT report builder.

The Apache Tomcat web server with the final implementation of the BIRT reporting application was placed on a Linux server running at UNCW in the Academic Research Computing (ARC) services. This required installing the BIRT Viewer on the Tomcat server and adding the reports to a directory in the viewer. The sequence diagram in Figure 6 shows the basic flow of the reporting application. Clients can view reports by going to the Available Reports report and typing in their client ID and password (shown in Figure 7). Once they have entered these values, the report presents a list of available reports the client can view (shown in Figure 8). If the client clicks on a link for one of the reports, a new page is loaded to generate that report. Dynamic parameters are generated based on the client ID, for the user to set (shown in Figure 9). Once the parameters are set, the report will generate (shown in Figure 10).

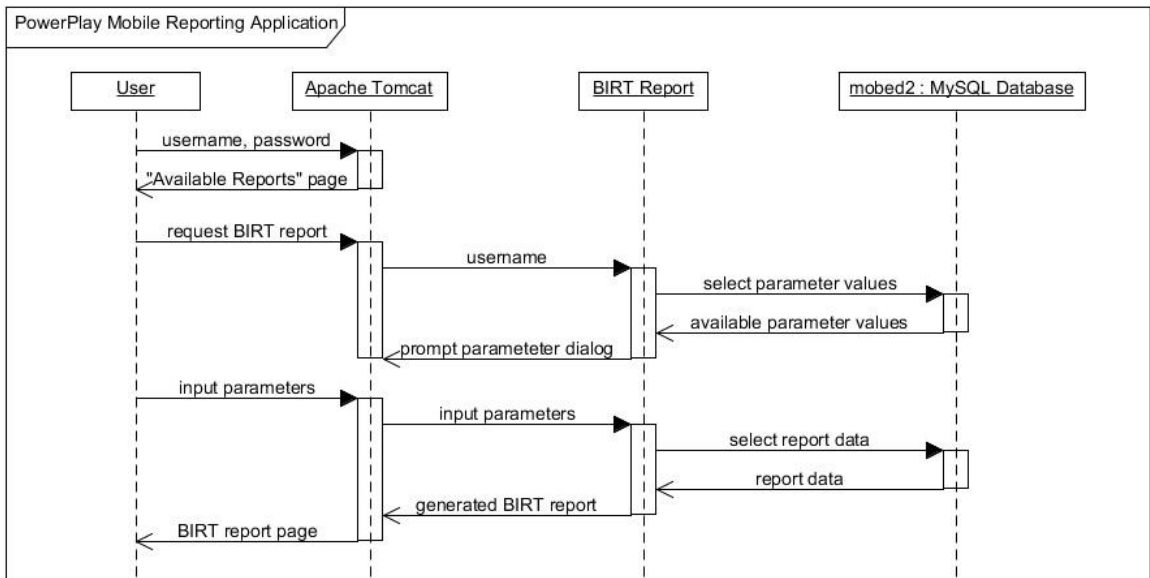


Figure 6 – A sequence diagram of PowerPlay Mobile’s reporting application.

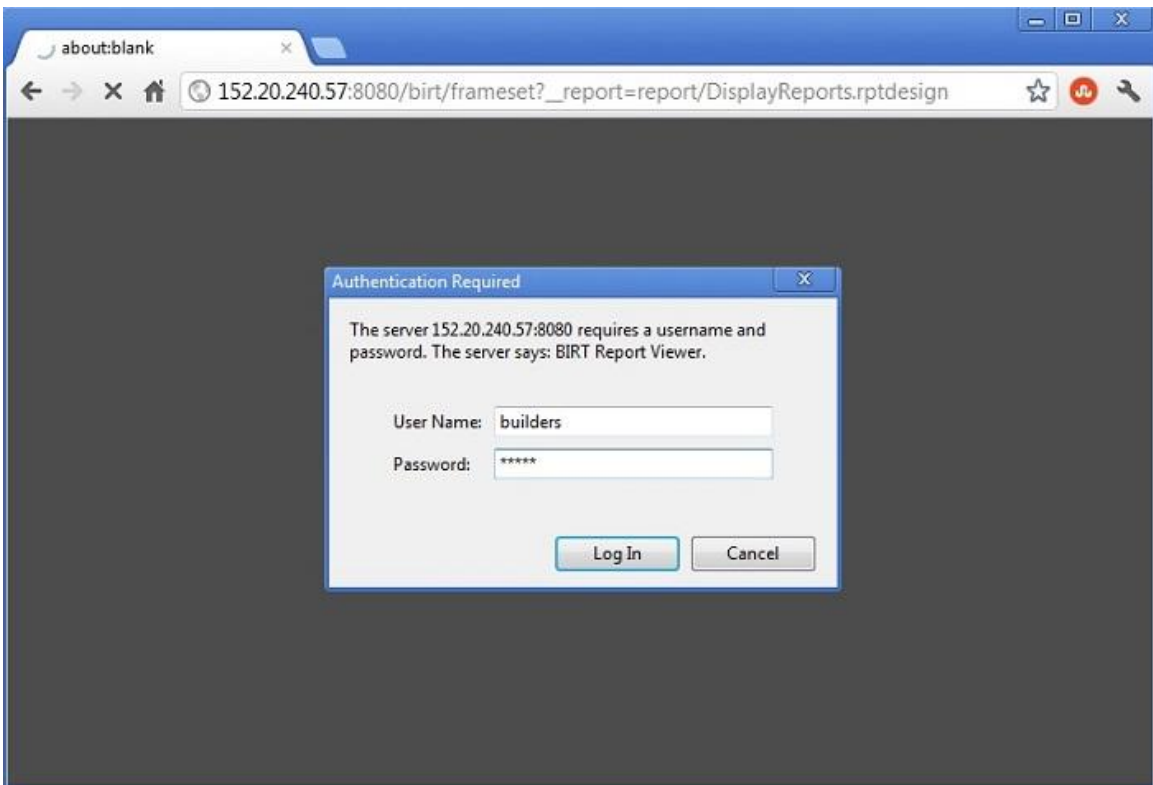


Figure 7 – The Tomcat HTTP basic authentication dialog.



Figure8 – The Available Reports web page for PowerPlay Mobile’s clients.

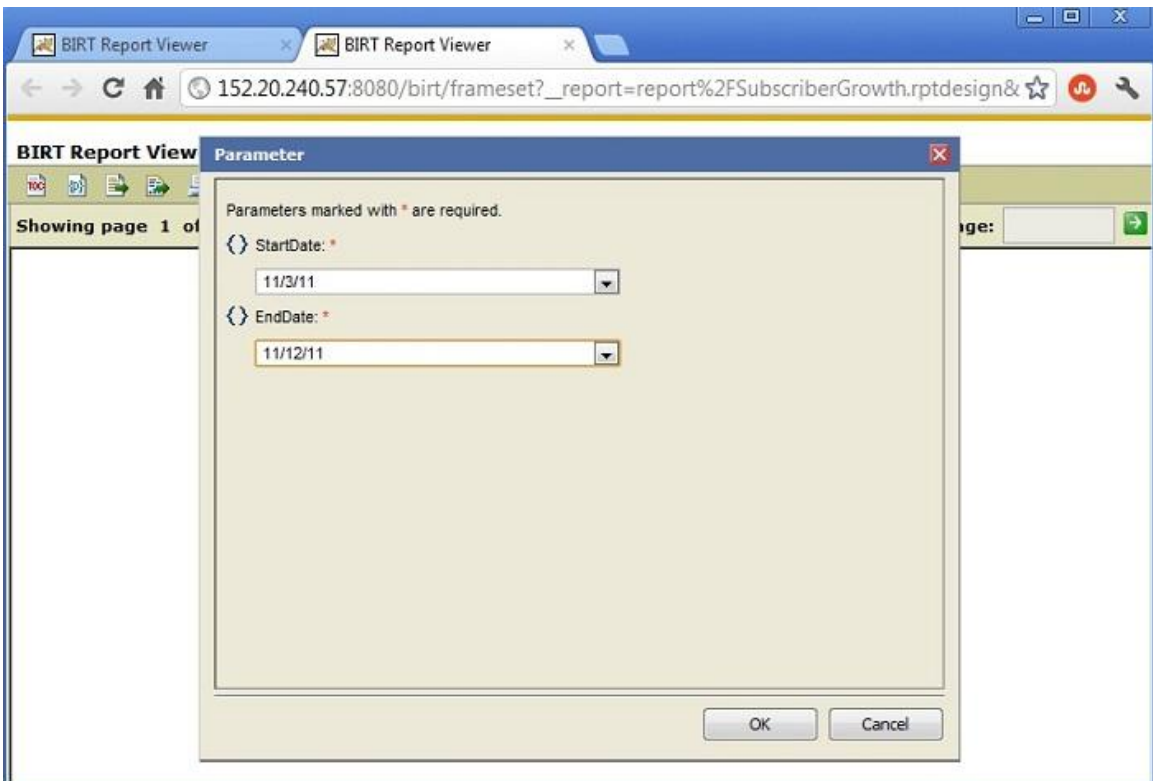


Figure 9 – The BIRT parameters dialog for the Subscriber Growth report.



*Figure 10 – The Subscriber Growth report for the “builders” client.*

Overall, BIRT’s functionality worked very well. Cascading dynamic report parameters are available if some parameters are dependent upon others. The dataset can be represented in many different ways, with an easy drag-and-drop functionality. Tables from datasets are pre-formatted to be embedded into reports. The report builder made it simple to format design quickly and easily.

Security for BIRT was implemented using Apache Tomcat’s HTTP basic authentication. This was done by adding a “client” role and each of PowerPlay Mobile’s

clients as users to Tomcat's *tomcat-users.xml*, shown in *Appendix C*. A security constraint was added to the BIRT viewer web application to force login upon a page request. The username is then passed into each BIRT report as a required client parameter using the method calls also shown in *Appendix C*.

It was decided to utilize a data warehouse for the reporting application. The advantage to using a data warehouse to generate reports, rather than the live transaction database, is to allow separate data processing for these functions. The transaction database will not be interrupted by report generation, and vice versa. This allows the reporting application to scale up more easily. The disadvantage is that the data represented in the reporting application will not be live. At most, the data will be a day old.

The transaction database will also remove old, unused data after a certain amount of time has passed. The data will need to remain viewable in the reporting application, however, giving another advantage to having a data warehouse. This does create a problem for data migration, since the state of the database cannot be simply copied over. PowerPlay Mobile's system does not support any method for logging data insertions, updates, deletions. One possible solution for this is to modify the system to log these changes in a SQL text file as INSERT, UPDATE, and DELETE queries, so that the log may be scheduled to execute each night on the MySQL data warehouse to update the altered tables.

Since some tables will lose data, and others will not, there could be another way of migrating data into the data warehouse. The tables that will not lose data can simply be copied over on a regular basis. The tables that will lose data in the transaction database

can be updated in the data warehouse using select statements, as long as deletions are logged somewhere in the transaction database and updates will not occur.

As the size of the data warehouse grows, scalability of the reporting application may become an issue. Most of the reports requested by PowerPlay Mobile use indexed values, allowing them to scale up well. However, a few of the reports query the message log based on non-indexed values. This log will grow very large over time, causing this report to take longer to generate. It is unclear if this will be a noticeable change as the log grows.

During the course of this project, I learned many new concepts and techniques within the domain of business intelligence. Before beginning this project, I was unaware of the impact business intelligence tools can have on a company. I had never worked with linking data to a reporting server using a connection driver, such as JDBC and ODBC. It was especially difficult with Microsoft Reporting Services, since it required linking the ODBC connection to an instance of Microsoft SQL Server. I ran into issues with the ODBC driver using Microsoft when attempting to use dynamic parameters in queries. This required me to reacquaint myself with stored procedures, since it seems Microsoft's OPENQUERY() function does not support string concatenation, making it impossible to use a parameter in a simple SQL query.

I had not had any previous experience in report building. I was surprised at how intuitive and easy to learn the report builders were when designing, especially with the open-source solutions. I was impressed with how much was offered with some of the open-source solutions when compared with Microsoft's proprietary solution. Most reports created with Microsoft Reporting Services can be closely replicated with many of the

open-source products.

Prior to this project, I had never heard of the SWOT analysis. It is a simple analysis to use, but it is not without its shortcomings. As mentioned before, the analysis can be misleading by appearing to give the same amount of weight to all features and limitations of a product. However, this fault can easily be overcome simply by knowing which features are most important to the stakeholders of the project.

Another challenge in this project that I have never come across before is data migration from one MySQL Server to another. This can become very difficult when it cannot be solved simply by copying the current state of the database to the data warehouse.

There are many concepts and techniques that I have brought into this project from classes taken at UNCW. My database classes of course taught me the architecture behind databases and how to query them using SQL statements. I have had classes in which I have used Apache Tomcat to deploy web services, which became necessary by the end of this project. I have had some experience with evaluation a list of products for a specific purpose, but not to the extent of this project. This project has been a significant learning experience in preparing me for future endeavors in similar fields.

## **Section 5: Conclusions and Future Work**

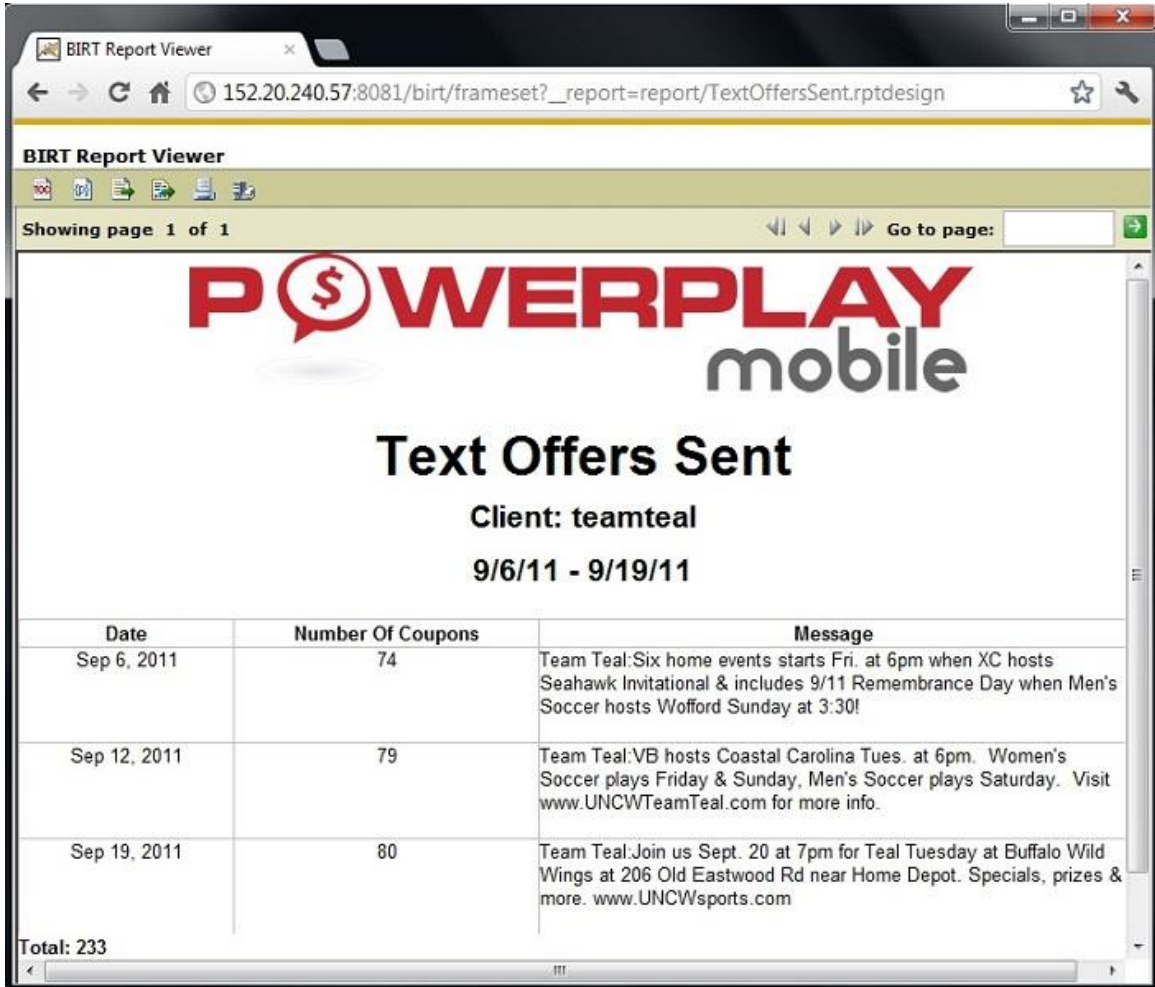
PowerPlay Mobile needed a solution for a reporting application which would be easy to maintain and grow. After a SWOT analysis of one proprietary solution and five open source solutions, two were chosen for further evaluation: Microsoft Reporting Services and BIRT. Once several requested reports were implemented in both solutions, a final product, BIRT, was chosen for implementation for PowerPlay Mobile. An Apache Tomcat web server with a BIRT Viewer was installed on a Linux server, and the reports were copied over into the viewer's directory. Security authentication and authorization for reports were implemented using Apache Tomcat's basic HTTP authentication. Finally, a data migration plan was created to schedule nightly updates to the data warehouse.

For future work on this project, new reports will be added in order to accommodate requests from clients. This should not be difficult with the current configuration. Changes will also be made in the future to the current state of PowerPlay Mobiles' database, which will allow for new reports to be generated which previously were not possible due to a lack of data. Data migration will need to be implemented in some way, possibly with one of the solutions mentioned earlier in the paper. Changes in the database may make data migration easier to implement.

## References

- [1] *Apache Tomcat*. 2012. The Apache Software Foundation. April 2012. <<http://tomcat.apache.org/>>.
- [2] *ART a lightweight reporting solution*. 2011. ART. February 2012. <<http://art.sourceforge.net/> >.
- [3] *Berthold, Henrike et al. (2010)*. An Architecture for Ad-hoc and Collaborative Business Intelligence. ACM Library.
- [4] *BIRT Project*. 2012. The Eclipse Foundation. February 2012. <<http://www.eclipse.org/birt/phoenix/> >.
- [5] *Gameiro, Carlos (2011)*. Implementation of Business Intelligence tools using Open Source Approach. ACM Library.
- [6] *JasperForge.org*. 2012. JasperForge. February 2012. <<http://www.jasperforge.org>>.
- [7] *Javaeye Reporting Tool*. 2004. Javaeye.com. February 2012. <<http://jert.sourceforge.net/> >.
- [8] *Park, Byung-Kwon and Il-Yeol Song (2011)*. Toward Total Business Intelligence Incorporating Structured and Unstructured Data. ACM Library.
- [9] *Pentaho Reporting Project*. 2012. Geeknet, Inc. February 2012. <<http://reporting.pentaho.com/>>.
- [10] *SQL Reporting Services*. 2011. Microsoft Corporation. February 2012. <<http://www.microsoft.com/sqlserver/en/us/solutions-technologies/business-intelligence/reporting-services.aspx>>.
- [11] *Wasserman, Ted J. et al. (2004)*. Developing a Characterization of Business Intelligence Workloads for Sizing New Database Systems. ACM Library.
- [12] *Viaene, Stijn (2008)*. Linking Business Intelligence into Your Business. IEEE Computer Society.

## Appendix A: Screen Shots of BIRT Reports



The screenshot displays a web browser window titled "BIRT Report Viewer" with the URL "152.20.240.57:8081/birt/frameset?\_report=report/TextOffersSent.rptdesign". The report content features the "POWERPLAY mobile" logo and the title "Text Offers Sent" for client "teamteal" covering the period "9/6/11 - 9/19/11". The report shows a table of text offers sent, with a total of 233 coupons.

Date	Number Of Coupons	Message
Sep 6, 2011	74	Team Teal:Six home events starts Fri. at 6pm when XC hosts Seahawk Invitational & includes 9/11 Remembrance Day when Men's Soccer hosts Wofford Sunday at 3:30!
Sep 12, 2011	79	Team Teal:VB hosts Coastal Carolina Tues. at 6pm. Women's Soccer plays Friday & Sunday, Men's Soccer plays Saturday. Visit <a href="http://www.UNCWTeamTeal.com">www.UNCWTeamTeal.com</a> for more info.
Sep 19, 2011	80	Team Teal:Join us Sept. 20 at 7pm for Teal Tuesday at Buffalo Wild Wings at 206 Old Eastwood Rd near Home Depot. Specials, prizes & more. <a href="http://www.UNCWsports.com">www.UNCWsports.com</a>

Total: 233

BIRT Report Viewer

152.20.240.57:8081/birt/frameset?\_report=report/CurrentSubscribers.rptdesign

BIRT Report Viewer

Showing page 1 of 1

**POWERPLAY**  
mobile

**Current Subscribers**

Client: uncwsports

Date: 11/12/11

Number of Subscribers: 16

BIRT Report Viewer

152.20.240.57:8081/birt/frameset?\_report=report/MessageCount.rptdesign

BIRT Report Viewer

Showing page 1 of 1

**POWERPLAY**  
mobile

**Message Count**

Client: uncwsports

7/1/11 - 11/12/11

Message Count: 2

BIRT Report Viewer

152.20.240.57:8081/birt/frameset?\_report=report/MessageLog.rptdesign

**BIRT Report Viewer**

Showing page 1 of 1

**POWERPLAY**  
mobile

## Message Log

Client: teamteal

9/6/11 - 9/19/11

Phone Number	Exchange Type	Time	Message
+12059372873	broadcast	Sep 6, 2011 3:21 PM	Team Teal:Six home events starts Fri. at 6pm when XC hosts Seahawk Invitational & includes 9/11 Remembrance Day when Men's Soccer hosts Wofford Sunday at 3:30!
+12059372873	broadcast	Sep 12, 2011 12:14 PM	Team Teal:VB hosts Coastal Carolina Tues. at 6pm. Women's Soccer plays Friday & Sunday, Men's Soccer plays Saturday. Visit <a href="http://www.UNCWTeamTeal.com">www.UNCWTeamTeal.com</a> for more info.
+12059372873	broadcast	Sep 19, 2011 11:32 AM	Team Teal:Join us Sept. 20 at 7pm for Teal Tuesday at Buffalo Wild Wings at 206 Old Eastwood Rd near Home Depot. Specials, prizes & more. <a href="http://www.UNCWsports.com">www.UNCWsports.com</a>

BIRT Report Viewer

152.20.240.57:8081/birt/frameset?\_report=report/NewSubscribers.rptdesign

BIRT Report Viewer

Showing page 1 of 1

**POWERPLAY**  
mobile

**New Subscribers**

Client: spa

11/3/11 - 11/12/11

Number of New Subscribers: 7

BIRT Report Viewer

152.20.240.57:8081/birt/frameset?\_report=report/OptOuts.rptdesign

BIRT Report Viewer

Showing page 1 of 1

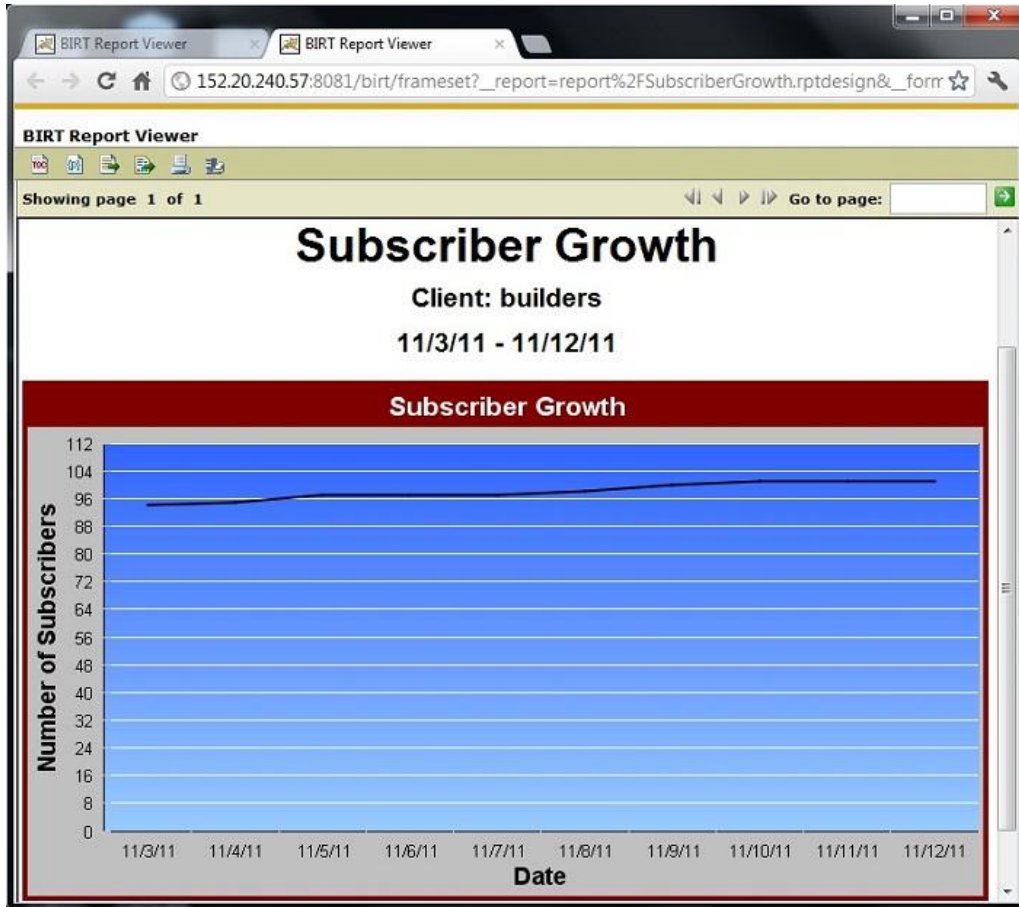
**POWERPLAY**  
mobile

**Opt-Outs**

Client: food

7/1/11 - 11/12/11

Number of Opt-Outs: 1



## Appendix B: Microsoft SQL Server Stored Procedures

### *Available Dates from Message*

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE AvailableDates
    @Client text,
    @Phone text
AS
BEGIN
    SET NOCOUNT ON;
    EXEC ('SELECT CONVERT(VARCHAR(10), Date, 101) AS date FROM
OPENQUERY(MOBED2_LINKED, ''SELECT DATE(time) as Date FROM message WHERE
clientId = '''' + @Client + '''' AND phone = '''' + @Phone + ''''
GROUP BY DATE(time) ORDER BY DATE(time)'')')
END
GO
```

### *Subscriber Growth*

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE SubscriberGrowth
    @Client text,
    @StartDate text,
    @EndDate text
AS
BEGIN
    SET NOCOUNT ON;
    EXEC ('SELECT CONVERT(VARCHAR(10), date, 101) as date, subscribers
FROM OPENQUERY(MOBED2_LINKED, ''SELECT DATE(time) as date, subscribers
FROM subscribecount WHERE DATE(time) >= STR_TO_DATE(''''' + @StartDate
+ ''''', "%m/%d/%Y") AND DATE(time) <= STR_TO_DATE(''''' + @EndDate +
''''', "%m/%d/%Y") AND clientId = '''' + @Client + ''''')')
END
GO
```

### *Subscriber Count Dates*

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE SubscribeCountDates
    @Client text
AS
```

```

BEGIN
    SET NOCOUNT ON;
    EXEC ('SELECT CONVERT(VARCHAR(10), Date, 101) AS date FROM
OPENQUERY(MOBED2_LINKED, 'SELECT DATE(time) as Date FROM subscribecount
WHERE clientId = '''' + @Client + '''' GROUP BY DATE(time) ORDER BY
DATE(time)''')')
END
GO

```

### *Opt-Outs by Client*

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE OptOutByClient
    @Client text,
    @StartDate text,
    @EndDate text
AS
BEGIN
    SET NOCOUNT ON;
    EXEC ('SELECT total FROM OPENQUERY(MOBED2_LINKED, 'SELECT
COUNT(DISTINCT phone) AS total FROM message WHERE clientId = '''' +
@Client + '''' AND DATE(time) >= STR_TO_DATE(''''' + @StartDate +
''''', "%m/%d/%Y") AND DATE(time) <= STR_TO_DATE(''''' + @EndDate +
''''', "%m/%d/%Y") AND body LIKE "%stop" AND exchangeType = "mo"''')')
END
GO

```

### *New Subscribers*

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE NewSubscribers
    @Client text,
    @StartDate text,
    @EndDate text
AS
BEGIN
    SET NOCOUNT ON;

    EXEC ('SELECT newsubscribers FROM OPENQUERY(MOBED2_LINKED, 'SELECT
(SELECT subscribers FROM subscribecount WHERE clientId = '''' +
@Client + '''' AND DATE(time) = STR_TO_DATE(''''' + @EndDate + ''''',
"%m/%d/%Y")) - (SELECT subscribers FROM subscribecount WHERE clientId =
'''' + @Client + '''' AND DATE(time) = STR_TO_DATE(''''' + @StartDate
+ ''''', "%m/%d/%Y")) AS newsubscribers''')')
END

```

GO

### *Messages by Phone*

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE MessagesByPhone
    @Client text,
    @Phone text,
    @StartDate text,
    @EndDate text
AS
BEGIN
    SET NOCOUNT ON;
    EXEC('SELECT phone, clientId, exchangeType, time, body FROM
OPENQUERY(MOBED2_LINKED, ''SELECT phone, clientId, exchangeType, time,
body FROM message WHERE phone = '''' + @Phone + '''' AND clientId =
'''' + @Client + '''' AND DATE(time) >= STR_TO_DATE(''''' +
@StartDate + ''''', ''m/%d/%Y'') AND DATE(time) <= STR_TO_DATE(''''' +
@EndDate + ''''', ''m/%d/%Y'') ORDER BY messageId'')')
END
GO
```

### *Message Count by Client*

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE MessageCountByClient
    @Client text,
    @StartDate text,
    @EndDate text
AS
BEGIN
    SET NOCOUNT ON;
    EXEC('SELECT clientId, total FROM OPENQUERY(MOBED2_LINKED,
''SELECT clientId, COUNT(*) AS total FROM message WHERE clientId =
'''' + @Client + '''' AND DATE(time) >= STR_TO_DATE(''''' +
@StartDate + ''''', ''m/%d/%Y'') AND DATE(time) <= STR_TO_DATE(''''' +
@EndDate + ''''', ''m/%d/%Y'') GROUP BY clientId ORDER BY clientId'')')
END
GO
```

### *Current Subscribers by Client*

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE CurrentSubscribersByClient
    @Client text
AS
BEGIN
    SET NOCOUNT ON;
    EXEC('SELECT clientId, CONVERT(VARCHAR(10), date, 101) as date,
subscribers FROM OPENQUERY(MOBED2_LINKED, ''SELECT clientId, DATE(time)
as date, subscribers FROM subscribecount WHERE DATE(time) = (SELECT
MAX(DATE(time)) FROM subscribecount WHERE clientId = '''' + @Client +
'''' ) AND clientId = '''' + @Client + ''''''')')
END
GO
```

### *Coupons by Client*

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE CouponsByClient
    @Client text,
    @StartDate text,
    @EndDate text
AS
BEGIN
    SET NOCOUNT ON;
    EXEC('SELECT CONVERT(VARCHAR(10), date, 101) AS date, coupons,
body FROM OPENQUERY(MOBED2_LINKED, ''SELECT DATE(time) as date,
COUNT(*) as coupons, body FROM message WHERE clientId = '''' +
@Client + '''' AND exchangeType = "broadcast" AND DATE(time) >=
STR_TO_DATE(''''' + @StartDate + '''', "%m/%d/%Y") AND DATE(time) <=
STR_TO_DATE(''''' + @EndDate + '''', "%m/%d/%Y") GROUP BY DATE(time),
body ORDER BY DATE(time), body''')')
END
GO
```

### *Phones by Client*

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE PhonesByClient
    @Client text
AS
```

```
BEGIN
    SET NOCOUNT ON;
    EXEC('SELECT phone FROM OPENQUERY(MOBED2_LINKED, ''SELECT phone
FROM user WHERE clientId = '''' + @Client + ''''''')')
END
GO
```

## Appendix C: BIRT Implementation Source Code

### *Current Subscribers Query*

```
SELECT      subscribers, DATE(time) as date
FROM        subscribecount
WHERE       DATE(time) <=
           (
             SELECT MAX(DATE(time))
             FROM   subscribecount
           )
           AND clientId = ?
ORDER BY    DATE(time) DESC
LIMIT 1
```

### *Broadcast Dates*

```
SELECT      DISTINCT(DATE(time)) AS Dates
FROM        message
WHERE       clientId = ? AND
           exchangeType = "broadcast"
ORDER BY    DATE(time)
```

### *Text Offers Sent*

```
SELECT      DATE(time) as date,
           COUNT(*) as coupons,
           body
FROM        message
WHERE       clientId = ? AND
           exchangeType = "broadcast" AND
           DATE(time) >= ? AND
           DATE(time) <= ?
GROUP BY    DATE(time), body
ORDER BY    DATE(time), body
```

### *Total Text Offers*

```
SELECT      COUNT(*) as Total
FROM        message
WHERE       clientId = ? AND
           exchangeType = "broadcast" AND
           DATE(time) >= ? AND
           DATE(time) <= ?
```

### *Available Subscriber Dates*

```
SELECT    DISTINCT(DATE(time)) AS Dates
FROM      subscribecount
WHERE     clientId = ?
ORDER BY  DATE(time)
```

### *Subscriber Counts*

```
SELECT    DATE(time) AS Date, subscribers
FROM      subscribecount
WHERE     clientId = ? AND
           DATE(time) >= ? AND
           DATE(time) <= ?
ORDER BY  DATE(time)
```

### *Dates of Opt-Outs*

```
SELECT    DISTINCT(DATE(time)) AS Dates
FROM      message
WHERE     clientId = ? AND
           body LIKE "%stop" AND
           exchangeType = "mo"
ORDER BY  DATE(time)
```

### *Opt-Outs Total*

```
SELECT    COUNT(DISTINCT(phone)) AS OptOuts
FROM      message
WHERE     clientId = ? AND
           DATE(time) >= ? AND
           DATE(time) <= ? AND
           body LIKE "%stop" AND
           exchangeType = "mo"
```

### *Dates of Current Subscribers*

```
SELECT    DISTINCT(DATE(time)) AS Dates
FROM      subscribecount
WHERE     clientId = ?
```

### *New Subscribers*

```
SELECT      s2.subscribers - s1.subscribers
FROM        subscribecount AS s2,
              (
                SELECT      s1.subscribers, s1.clientId
                FROM        subscribecount AS s1
                WHERE      DATE(time) = ? AND
                           s1.clientId >= ?
                ORDER BY DATE(s1.time) ASC
                LIMIT 1
              ) AS s1
WHERE      DATE(time) <= ? AND
           s2.clientId = s1.clientId
ORDER BY  DATE(s2.time) DESC
LIMIT 1
```

### *Dates of Messages by Phone*

```
SELECT      DISTINCT(DATE(time)) AS Dates
FROM        message
WHERE      clientId = ? AND
           phone = ?
ORDER BY  DATE(time)
```

### *Phone Numbers in Message Log*

```
SELECT      phone AS PhoneNumbers
FROM        user
WHERE      clientId = ?
ORDER      BY phone
```

### *Message Log*

```
SELECT      phone, exchangeType, time, body
FROM        message
WHERE      clientId = ? AND
           phone = ? AND
           DATE(time) >= ? AND
           DATE(time) <= ?
ORDER BY  time
```

### *Dates of Messages*

```
SELECT      DISTINCT(DATE(time)) AS Dates
FROM        message
WHERE      clientId = ?
ORDER      BY DATE(time)
```

## *Message Count*

```
SELECT      COUNT(*) AS MessageCount
FROM        message
WHERE       clientId = ? AND
           DATE(time) >= ? AND
           DATE(time) <= ?
```

## *Apache Tomcat: tomcat-users.xml*

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="client"/>

  <user username="builders" password="pass1" roles="client"/>
  <user username="spa" password="pass2" roles="client"/>
  <user username="teamteal" password="pass3" roles="client"/>
  <user username="food" password="pass4" roles="client"/>
  <user username="uncwsports" password="pass5" roles="client"/>
  <!-- All users for PowerPlay Mobile's clients go here. -->
</tomcat-users>
```

## *Client Report Parameter from Tomcat HTTP Basic Web Authentication*

```
reportContext.getHttpServletRequest().getRemoteUser();
```

## *BIRT: web.xml*

```
<?xml version="1.0" encoding="utf-8"?>
<web-app
  xmlns="http://java.sun.com/xml/ns/j2ee"
  version="2.4"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">

  <display-name>Eclipse BIRT Report Viewer</display-name>
  <context-param>
    <param-name>BIRT_VIEWER_LOCALE</param-name>
    <param-value>en-US</param-value>
  </context-param>
  <context-param>
    <param-name>BIRT_VIEWER_TIMEZONE</param-name>
    <param-value></param-value>
  </context-param>
  <context-param>
    <param-name>BIRT_VIEWER_WORKING_FOLDER</param-name>
    <param-value></param-value>
  </context-param>
```

```

<context-param>
  <param-name>BIRT_VIEWER_DOCUMENT_FOLDER</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>WORKING_FOLDER_ACCESS_ONLY</param-name>
  <param-value>>true</param-value>
</context-param>
<context-param>
  <param-name>URL_REPORT_PATH_POLICY</param-name>
  <param-value>domain</param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_IMAGE_DIR</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_LOG_DIR</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_LOG_LEVEL</param-name>
  <param-value>WARNING</param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_SCRIPTLIB_DIR</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>BIRT_RESOURCE_PATH</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_MAX_ROWS</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_MAX_CUBE_ROWLEVELS</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_MAX_CUBE_COLUMNLEVELS</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_CUBE_MEMORY_SIZE</param-name>
  <param-value></param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_CONFIG_FILE</param-name>
  <param-value>WEB-INF/viewer.properties</param-value>
</context-param>
<context-param>
  <param-name>BIRT_VIEWER_PRINT_SERVERSIDE</param-name>
  <param-value>ON</param-value>

```

```

</context-param>
<context-param>
  <param-name>HTML_ENABLE_AGENTSTYLE_ENGINE</param-name>
  <param-value>>true</param-value>
</context-param>
<context-param>
  <param-name>BIRT_FILENAME_GENERATOR_CLASS</param-name>
  <param-
value>org.eclipse.birt.report.utility.filename.DefaultFilenameGenerator
</param-value>
</context-param>
<filter>
  <filter-name>ViewerFilter</filter-name>
  <filter-
class>org.eclipse.birt.report.filter.ViewerFilter</filter-class>
</filter>
  <filter-mapping>
    <filter-name>ViewerFilter</filter-name>
    <servlet-name>ViewerServlet</servlet-name>
  </filter-mapping>
  <filter-mapping>
    <filter-name>ViewerFilter</filter-name>
    <servlet-name>EngineServlet</servlet-name>
  </filter-mapping>
<listener>
  <listener-
class>org.eclipse.birt.report.listener.ViewerServletContextListener</li
stener-class>
</listener>
<listener>
  <listener-
class>org.eclipse.birt.report.listener.ViewerHttpSessionListener</liste
ner-class>
</listener>
<servlet>
  <servlet-name>ViewerServlet</servlet-name>
  <servlet-
class>org.eclipse.birt.report.servlet.ViewerServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>EngineServlet</servlet-name>
  <servlet-
class>org.eclipse.birt.report.servlet.BirtEngineServlet</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>ViewerServlet</servlet-name>
  <url-pattern>/frameset</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>ViewerServlet</servlet-name>
  <url-pattern>/run</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>EngineServlet</servlet-name>
  <url-pattern>/preview</url-pattern>

```

```

</servlet-mapping>
<servlet-mapping>
  <servlet-name>EngineServlet</servlet-name>
  <url-pattern>/download</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>EngineServlet</servlet-name>
  <url-pattern>/parameter</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>EngineServlet</servlet-name>
  <url-pattern>/document</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>EngineServlet</servlet-name>
  <url-pattern>/output</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>EngineServlet</servlet-name>
  <url-pattern>/extract</url-pattern>
</servlet-mapping>
<jsp-config>
  <taglib>
    <taglib-uri>/birt.tld</taglib-uri>
    <taglib-location>/WEB-INF/tlds/birt.tld</taglib-location>
  </taglib>
</jsp-config>

<!-- Define a security constraint on this application -->
<security-constraint>
  <web-resource-collection>
    <web-resource-name>Entire Application</web-resource-name>
    <url-pattern>/*</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <!-- This role is not in the default user directory -->
    <role-name>client</role-name>
  </auth-constraint>
</security-constraint>
<!-- Define the login configuration for this application -->
<login-config>
  <auth-method>BASIC</auth-method>
  <realm-name>BIRT Report Viewer</realm-name>
</login-config>
<!-- Security roles referenced by this web application -->
<security-role>
  <description>
    The role that is required to log in to the BIRT Report Viewer
  </description>
  <role-name>client</role-name>
</security-role>
</web-app>

```

## Appendix D: PowerPlay Mobile Transaction Database

<b>ctyrole</b>			
	Column Name	Condensed Type	Nullable
*	name	varchar(25)	No
*	role	varchar(25)	No

<b>couponrecord</b>			
	Column Name	Condensed Type	Nullable
*	phone	varchar(20)	No
*	clientid	varchar(20)	No
*	last	date	No

<b>player_input</b>			
	Column Name	Condensed Type	Nullable
*	phone	char(12)	No

<b>xmfiapps</b>			
	Column Name	Condensed Type	Nullable
*	id	int	No
*	name	varchar(50)	No
*	description	varchar(254)	Yes
*	xmfi	text	No

<b>ctypocale</b>			
	Column Name	Condensed Type	Nullable
*	name	varchar(50)	No

<b>missingadjusted</b>			
	Column Name	Condensed Type	Nullable
*	phone	varchar(12)	Yes

<b>player</b>			
	Column Name	Condensed Type	Nullable
*	field	char(2)	No
*	value	int	No

<b>dubidd</b>			
	Column Name	Condensed Type	Nullable
*	field	char(2)	No
*	value	int	No

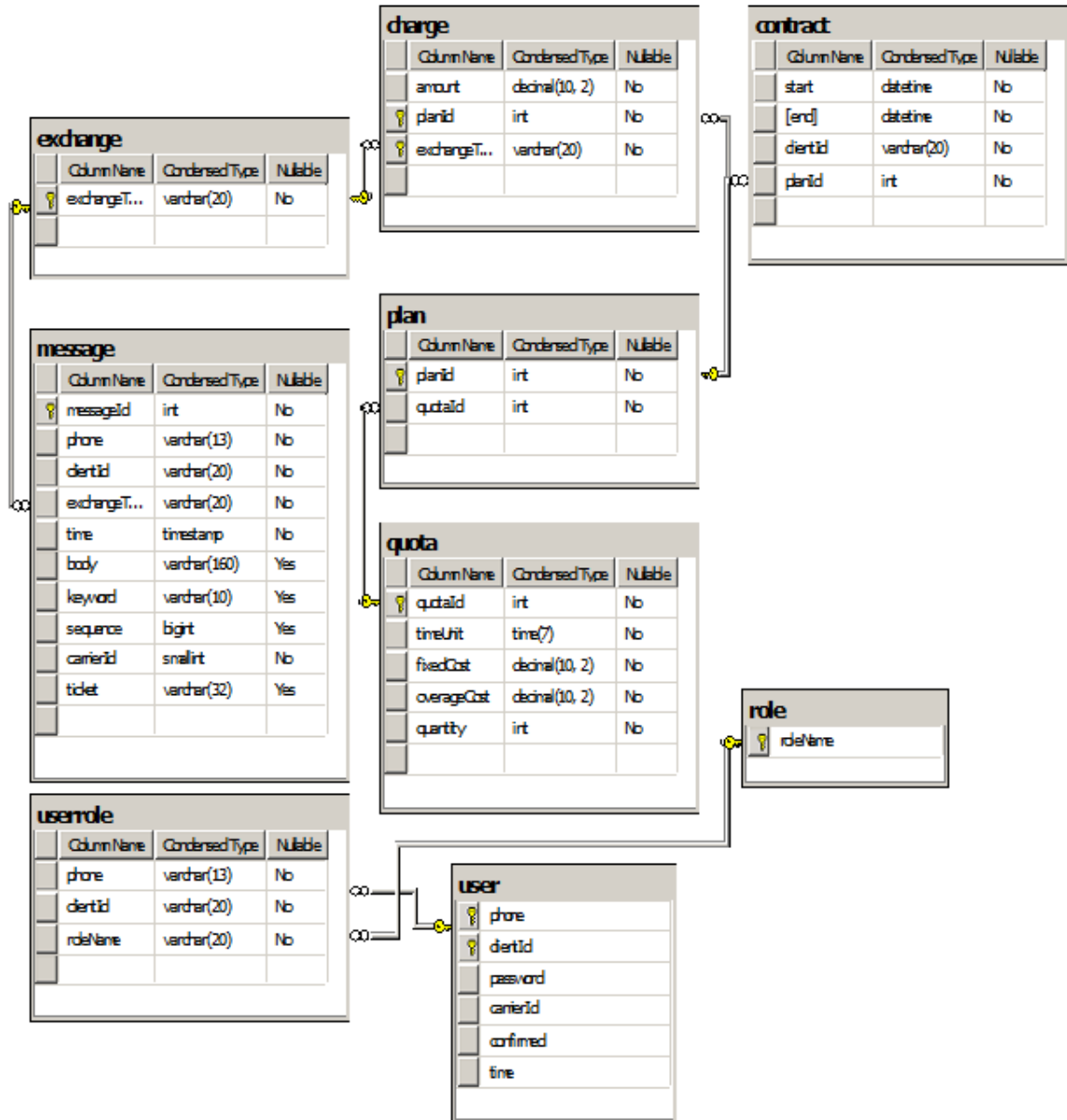
<b>intercept</b>			
	Column Name	Condensed Type	Nullable
*	name	varchar(20)	No
*	timeout	int	Yes

<b>ctypuser</b>			
	Column Name	Condensed Type	Nullable
*	name	varchar(25)	No
*	password	varchar(25)	Yes

<b>coupons</b>			
	Column Name	Condensed Type	Nullable
*	id	int	No
*	name	varchar(50)	Yes

<b>drawing</b>			
	Column Name	Condensed Type	Nullable
*	phone	varchar(12)	No
*	name	varchar(50)	No
*	info	int	Yes
*	selected	text	Yes
*	business	varchar(254)	Yes

<b>itsd</b>			
	Column Name	Condensed Type	Nullable
*	field	char(2)	No
*	value	int	No



vote_input			
Column Name	Condensed Type	Nullable	
phone	char(12)	No	

uniqueid			
Column Name	Condensed Type	Nullable	
id	varchar(15)	No	
client	varchar(20)	No	

subscribecount			
Column Name	Condensed Type	Nullable	
id	int	No	
clientId	varchar(20)	No	
subscribers	int	No	
time	timestamp	No	

nominate			
Column Name	Condensed Type	Nullable	
id	int	No	
time	timestamp	No	
host	varchar(100)	Yes	
business	varchar(100)	No	

couponuser			
Column Name	Condensed Type	Nullable	
personid	int	No	
couponid	int	No	
other	varchar(50)	Yes	

xmlinteractive			
Column Name	Condensed Type	Nullable	
clientId	varchar(20)	No	
phone	varchar(12)	No	
node	varchar(20)	Yes	

wavebc			
Column Name	Condensed Type	Nullable	
id	int	No	
bc_id	int	No	
sequence	int	No	
lat	int	No	
lon	int	No	
heading	int	No	
datetime	int	No	
speed	int	No	

trial			
Column Name	Condensed Type	Nullable	
phone	varchar(12)	No	
node	varchar(20)	Yes	

sent			
Column Name	Condensed Type	Nullable	
clientId	varchar(20)	No	
time	timestamp	No	
type	int	Yes	
body	varchar(160)	Yes	
phone	varchar(13)	Yes	
code	varchar(20)	Yes	

xmlarchive			
Column Name	Condensed Type	Nullable	
clientId	varchar(20)	No	
phone	varchar(12)	No	
xml	text	Yes	

vote			
Column Name	Condensed Type	Nullable	
field	char(2)	No	
value	int	No	

test			
Column Name	Condensed Type	Nullable	
member	int	Yes	
committee	varchar(10)	Yes	

coupon2			
Column Name	Condensed Type	Nullable	
id	int	No	
clientId	varchar(20)	Yes	
type	int	No	
text	varchar(160)	No	
weight	float	Yes	
notMoreThan	int	Yes	
notBefore	date	Yes	
notAfter	date	Yes	
code	varchar(20)	Yes	
startHour	varchar(10)	Yes	
endHour	varchar(10)	Yes	
startDay	varchar(10)	Yes	
endDay	varchar(10)	Yes	
initDays	int	Yes	
clientName	varchar(50)	Yes	
fromExpire	int	Yes	

contestant			
	Column Name	Condensed Type	Nullable
	phone	varchar(12)	No
	frname	varchar(20)	No
	lrname	varchar(30)	No
	email	varchar(60)	Yes
	gender	char(1)	Yes
	age	tinyint	Yes
	income	tinyint	Yes
	abc	tinyint	Yes
	[break]	tinyint	Yes
	cds	tinyint	Yes
	chicken	tinyint	Yes
	devids	tinyint	Yes
	frontstreet	tinyint	Yes
	gravity	tinyint	Yes
	greatdips	tinyint	Yes
	hamburgerj...	tinyint	Yes
	honeybaked	tinyint	Yes
	jerseyniles	tinyint	Yes
	katys	tinyint	Yes
	lubakou	tinyint	Yes
	pizza	tinyint	Yes
	spatdips	tinyint	Yes
	tan	tinyint	Yes
	ten	tinyint	Yes
	mail	int	Yes
	book	int	Yes
	grocery	int	Yes
	retail	int	Yes
	other	varchar(1)	Yes
	info	char(1)	No
	nicds	tinyint	Yes

coupon			
	Column Name	Condensed Type	Nullable
	clientid	varchar(20)	No
	type	int	No
	text	varchar(160)	No
	weight	float	No
	notMoreThan	int	No
	notBefore	date	No
	notAfter	date	No
	code	varchar(20)	Yes

ageranges			
	Column Name	Condensed Type	Nullable
	id	int	No
	range	varchar(1)	Yes

incomeranges			
	Column Name	Condensed Type	Nullable
	id	int	No
	range	varchar(20)	Yes

event			
	Column Name	Condensed Type	Nullable
	day	date	No
	text	varchar(160)	Yes

client			
	Column Name	Condensed Type	Nullable
	clientid	varchar(20)	No
	cred	varchar(50)	Yes
	name	varchar(100)	No
	servicesUrl	varchar(100)	Yes
	namespace	varchar(100)	Yes
	couponServer	varchar(100)	Yes
	couponMail	int	Yes
	type	int	Yes
	notes	text	Yes
	owner	varchar(25)	Yes

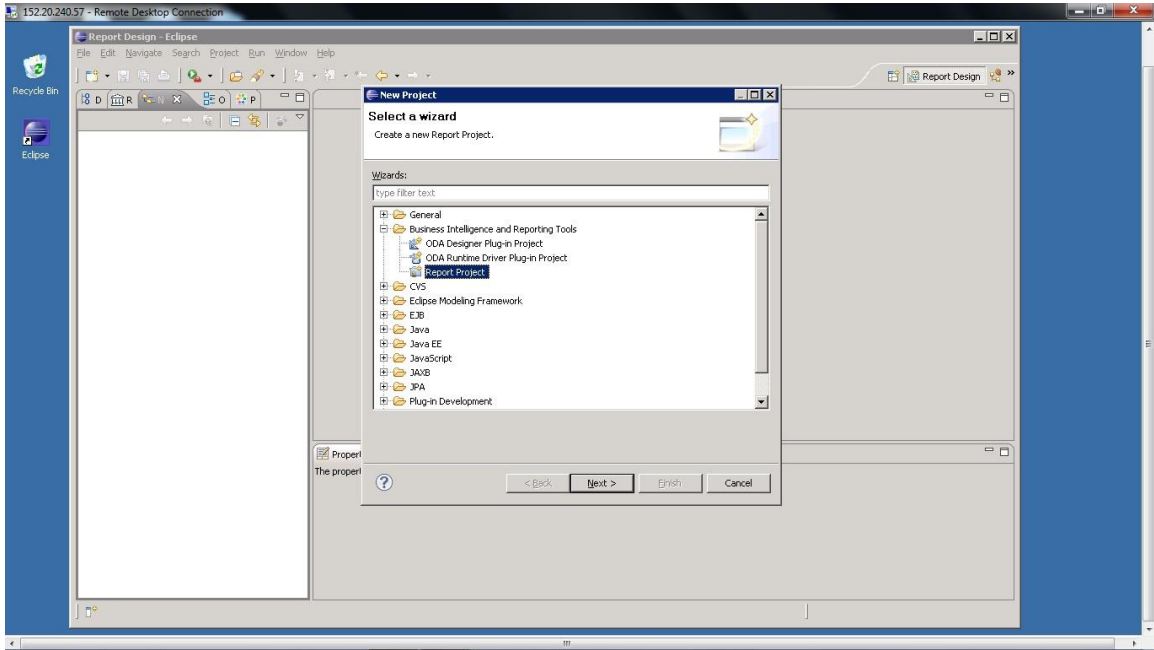
email			
	Column Name	Condensed Type	Nullable
	email	varchar(25)	No
	frname	varchar(20)	No
	lrname	varchar(20)	No
	type	varchar(1)	Yes

dubidol_input			
	Column Name	Condensed Type	Nullable
	phone	char(12)	No

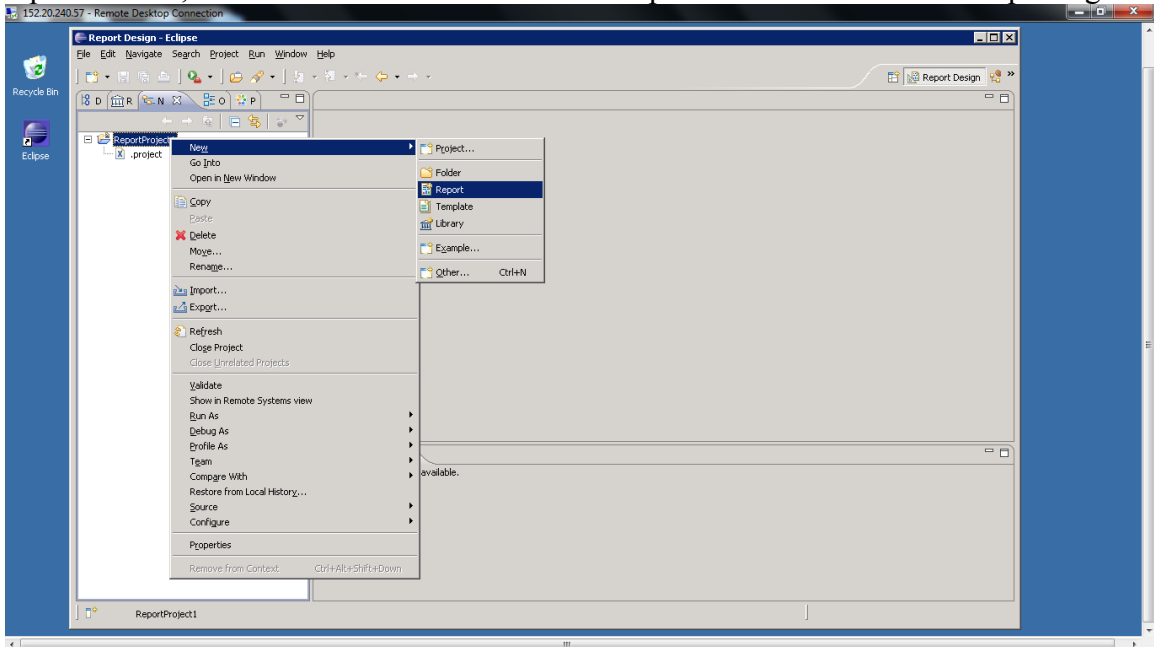
no			
	Column Name	Condensed Type	Nullable
	input	text	Yes

## Appendix E: Developer's Guide

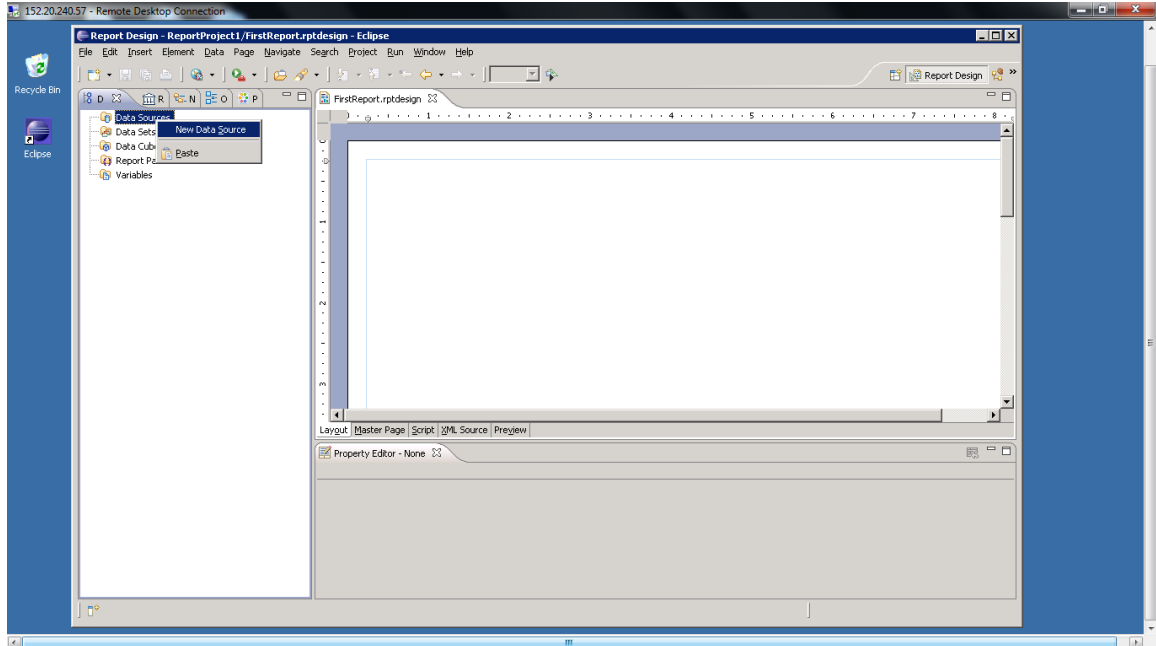
To create a new Report Project, click on **File > New > Project...**, and a “New Project” wizard should pop up. Expand the “**Business Intelligence and Reporting Tools**” folder, click on “**Report Project**”, and click “**Next**”. Give your project a name, and click “**Finish**.”



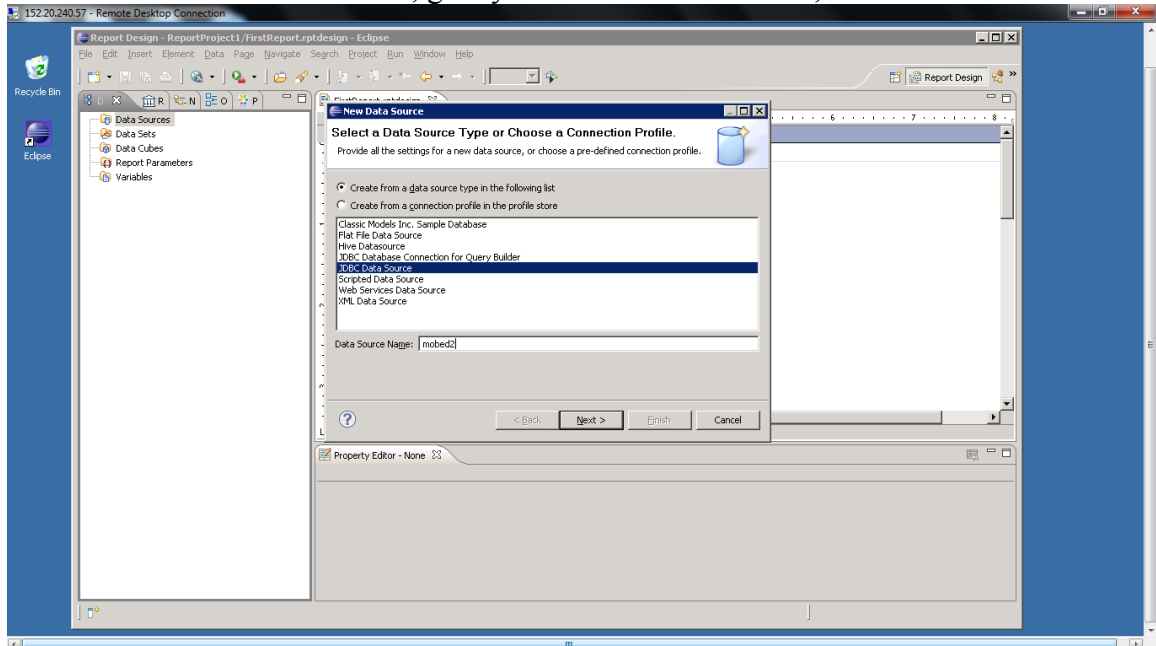
To create a new report, right click on your project, and go to **New > Report**. Give your report a name, and click “**Finish**”. Note that the report name must end with “.rptdesign”.



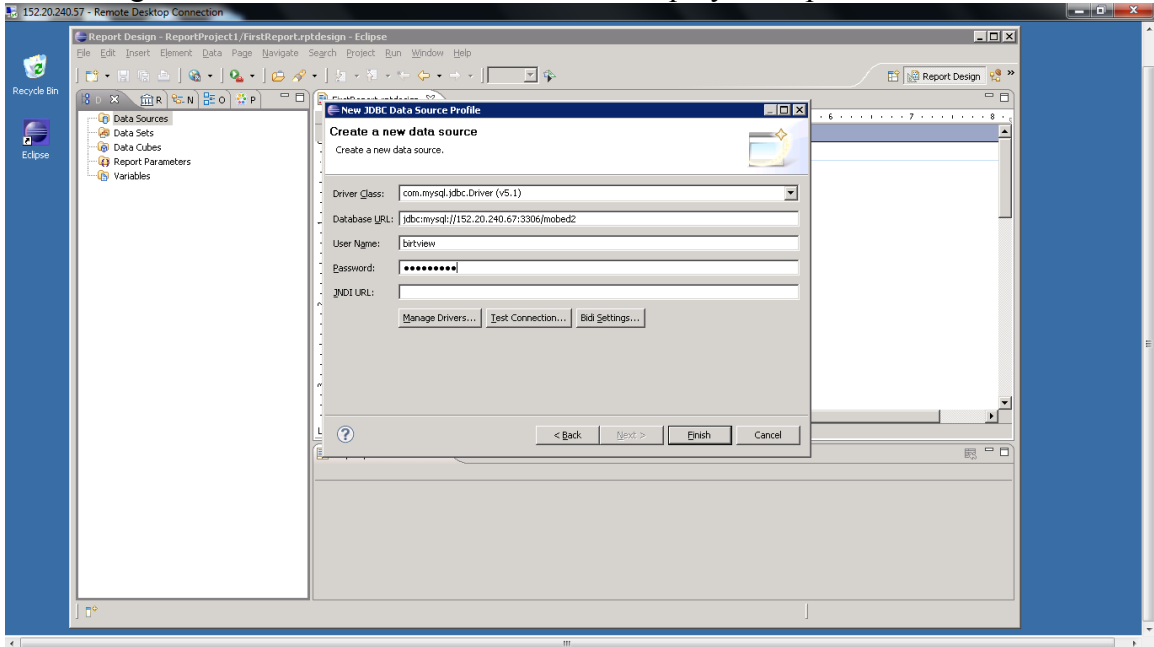
To create a new Data Source, click on your “**Data Explorer**” tab, right click on “**Data Sources**” and click on *New Data Source*.



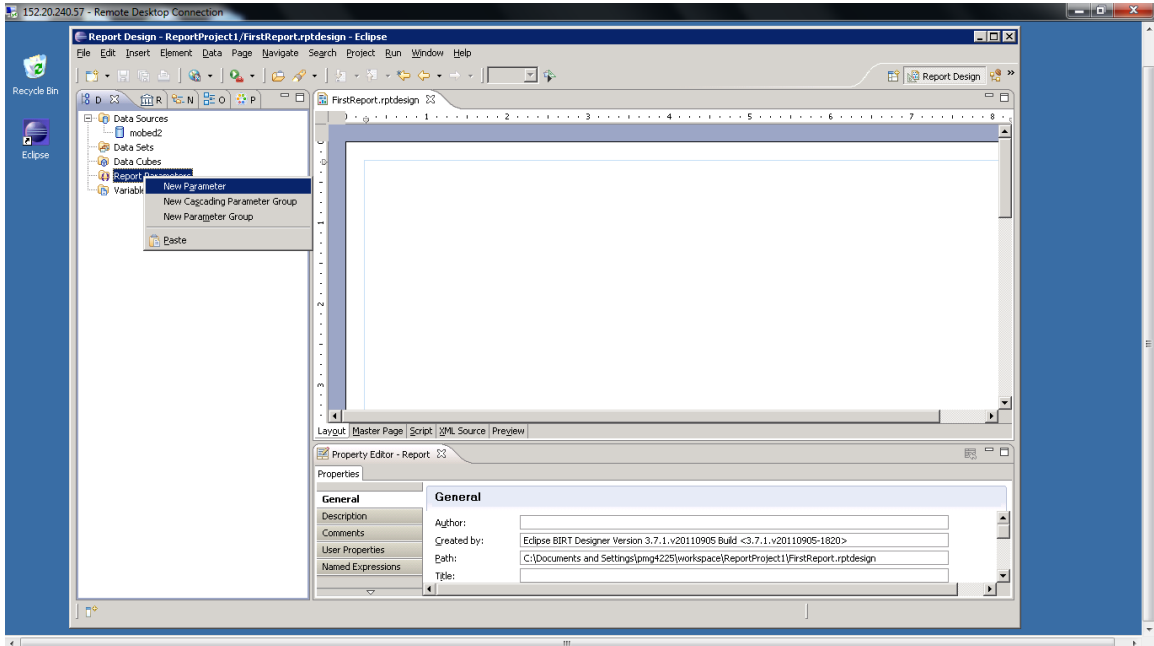
Click on “**JDBC Data Source**”, give your Data Source a name, and click “**Next**”.



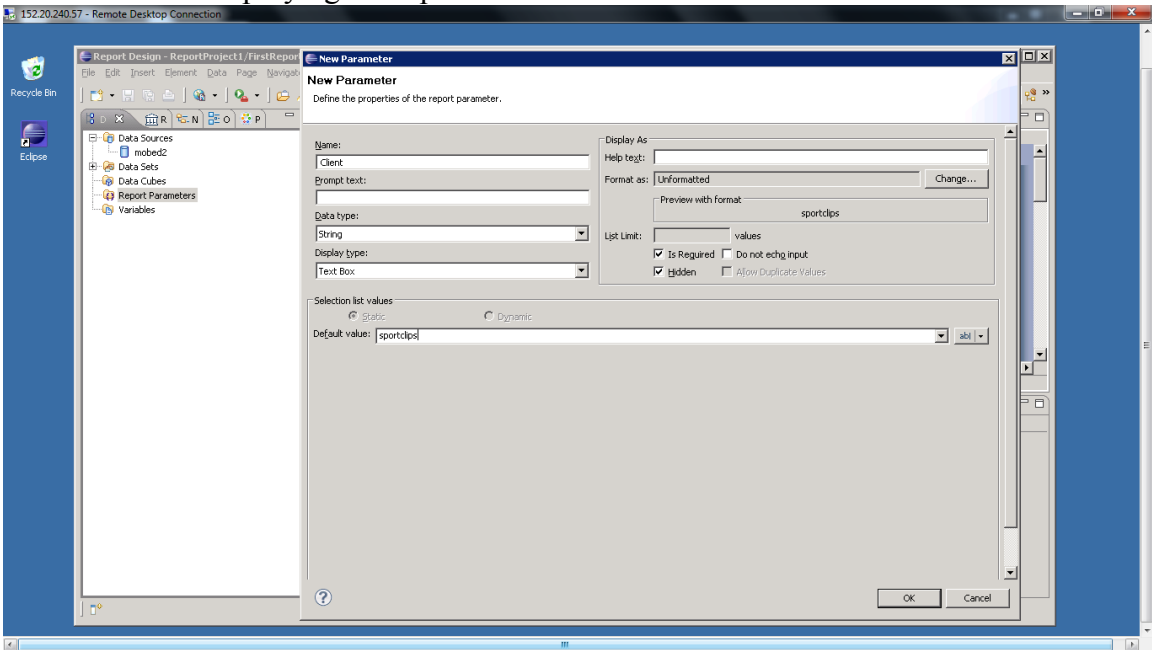
Fill out all of the form except for “JNDI URL”. Since the report will be dropped onto a Linux server running MySQL, we will give the IP address of the server to begin with, and later change the address to “localhost” before we deploy the report.



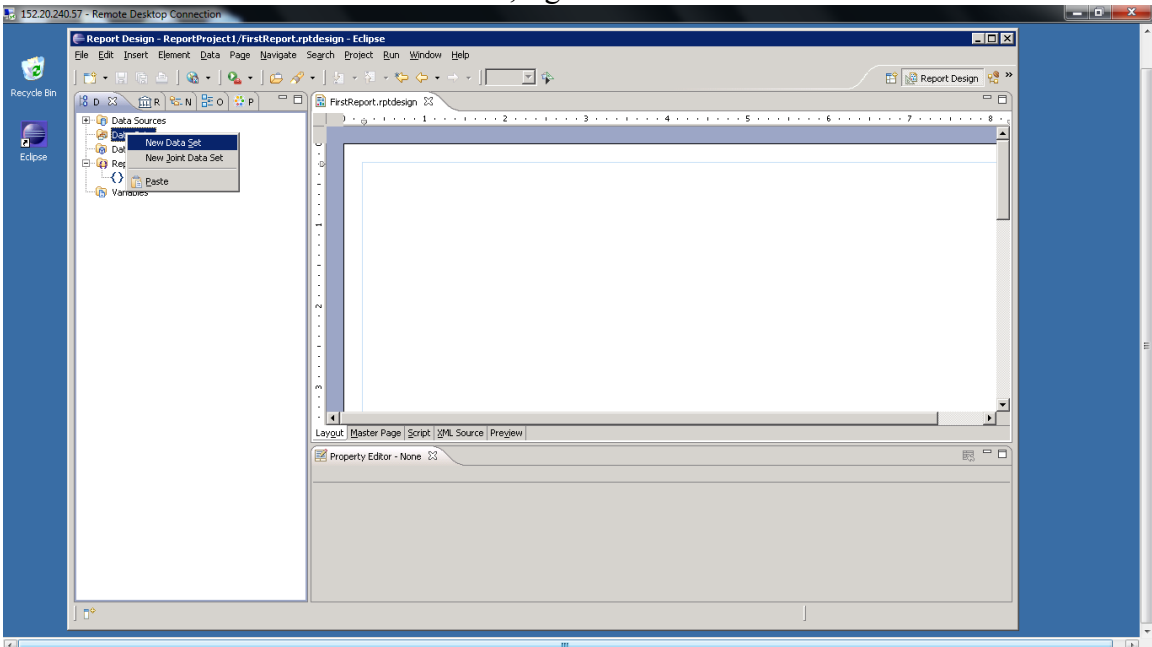
To create a “Client” parameter, right click on “Report Parameters” and click on *New Parameter*.



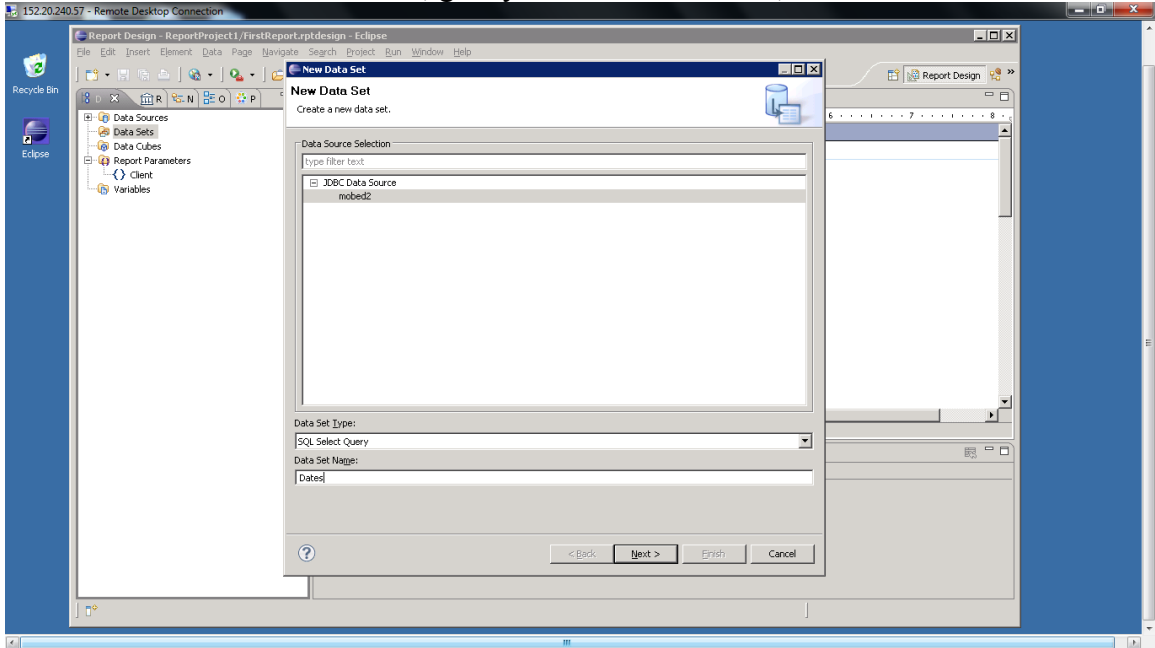
Fill out the form the same as below, and click “OK”. Make sure the “Hidden” check box is checked. We want to set a default value here so that queries to the MySQL server will work correctly for testing in the Report Builder. These default values will need to be removed before deploying the report.



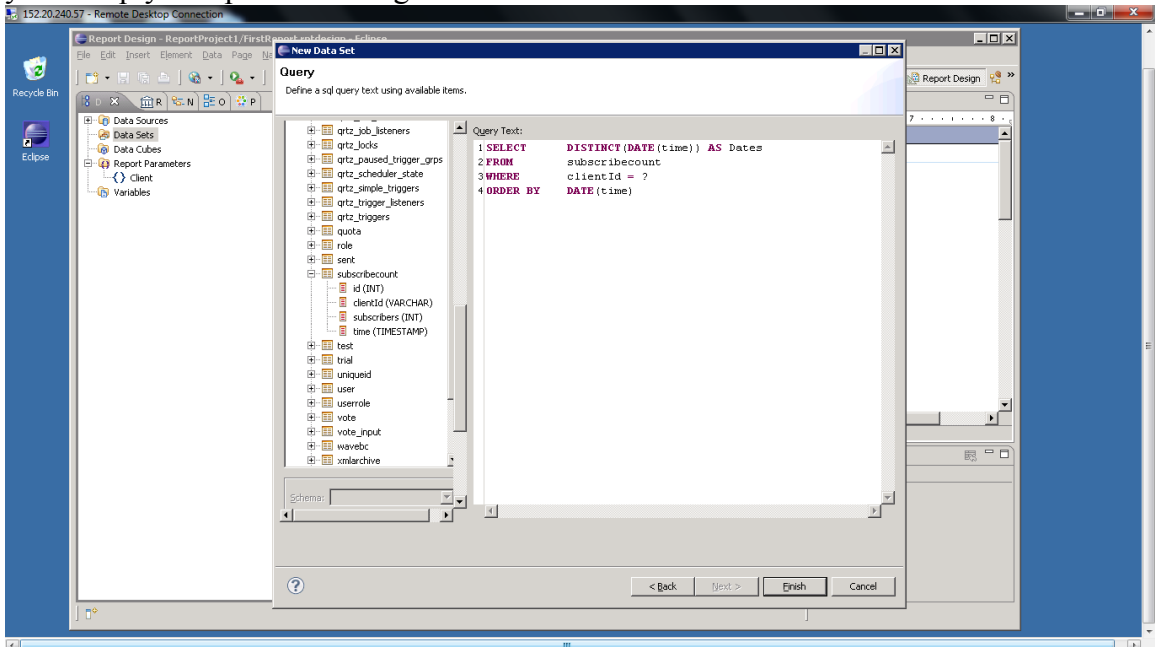
Now that we have a “Client” parameter, we can get a new Data Set for the available dates for that client. To create a new Data Set, right click on Data Sets then click **New Data Set**.



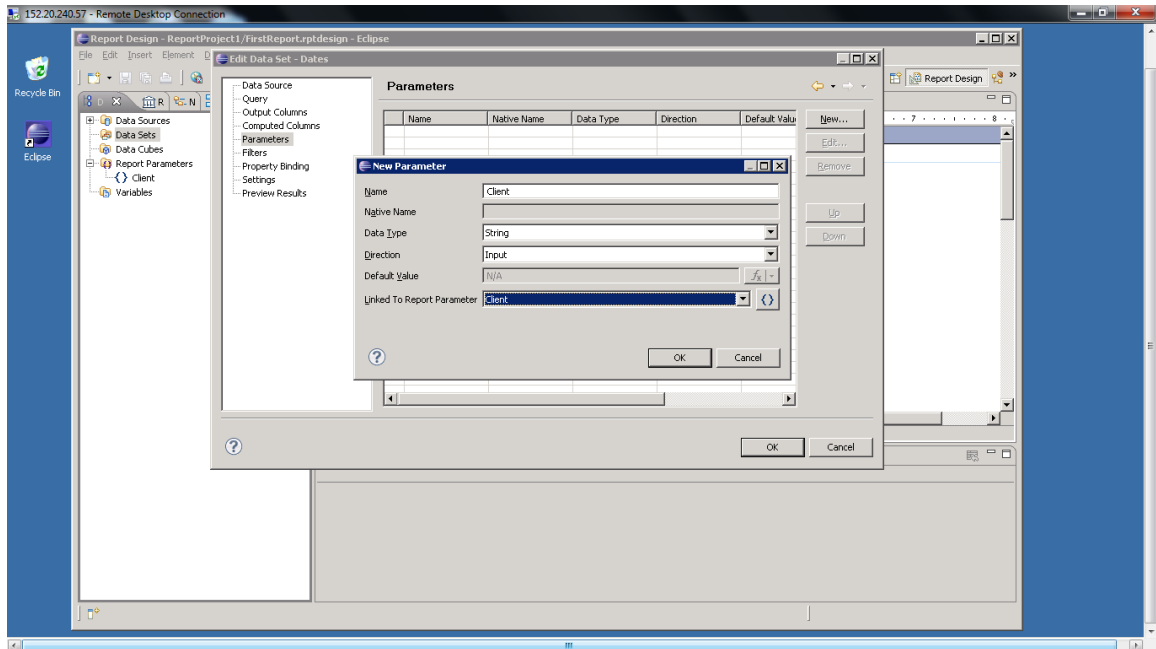
Click on the correct Data Source, give your Data Set a name, and click “Next”.



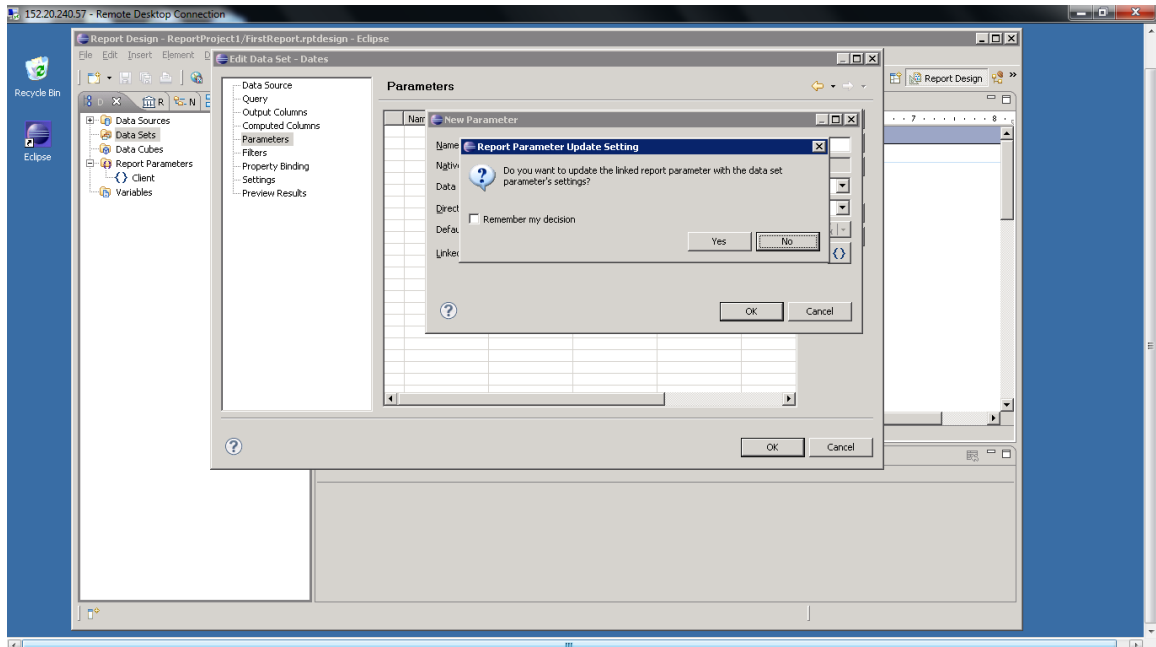
Fill out the query shown below. Use the “?” reserve word for any parameters coming into the query. Once you have entered the query, click “Finish”. You will receive a couple of exceptions after you finish; this is because you have not added parameters to the Data Set yet. Simply accept the warnings to move on.



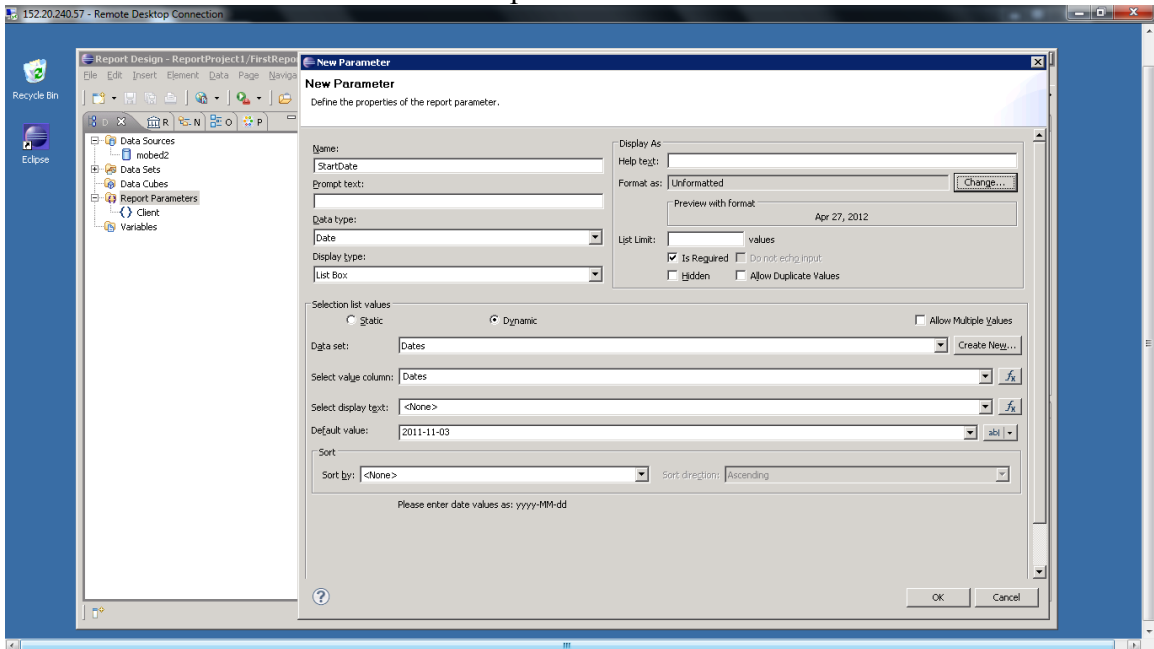
Click on “**Parameters**” and click on “**New**”. Add the “**Client**” parameter to the Data Set, shown below.



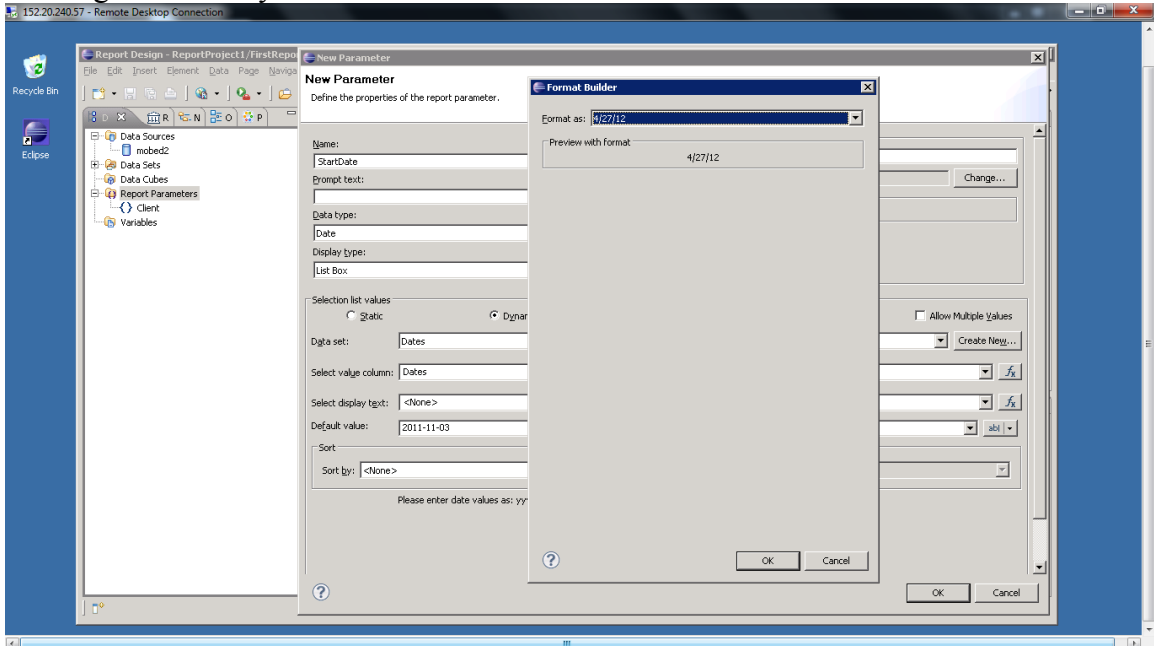
Click on “**No**” if the dialog box appears after linking to a report parameter. Once you have added the “**Client**” parameter, you should be able to “**Preview Results**” with the default value of “**Client**”.



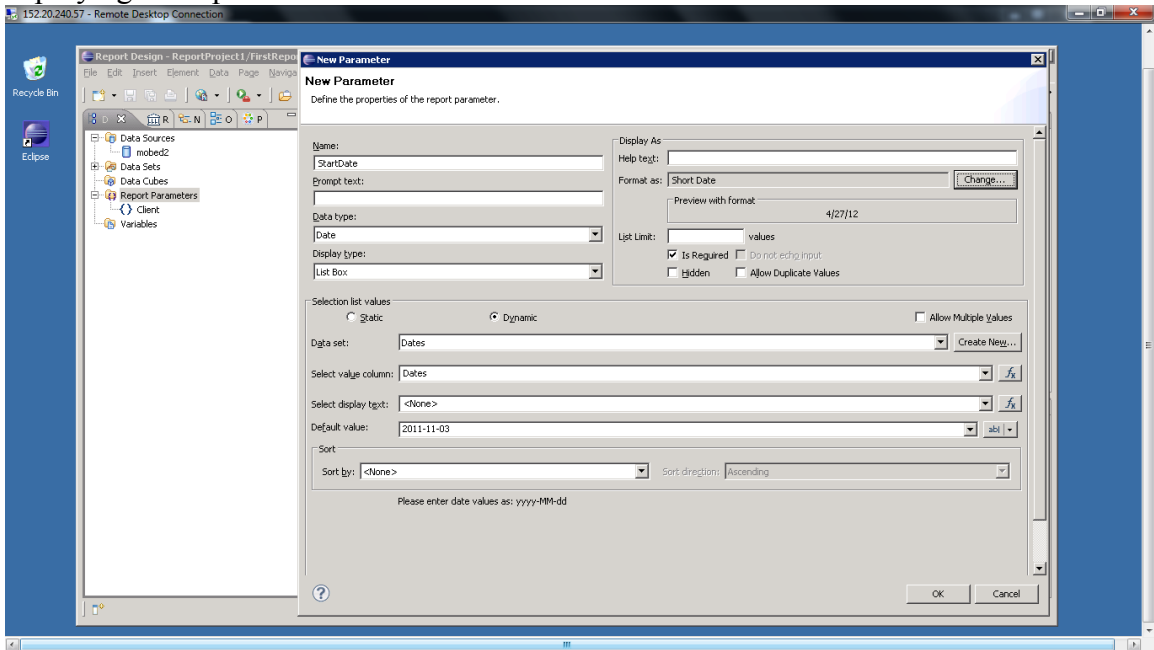
Now that we have a Data Set for “Dates”, we can create report parameters for “StartDate” and “EndDate”. Create the “StartDate” parameter shown below.



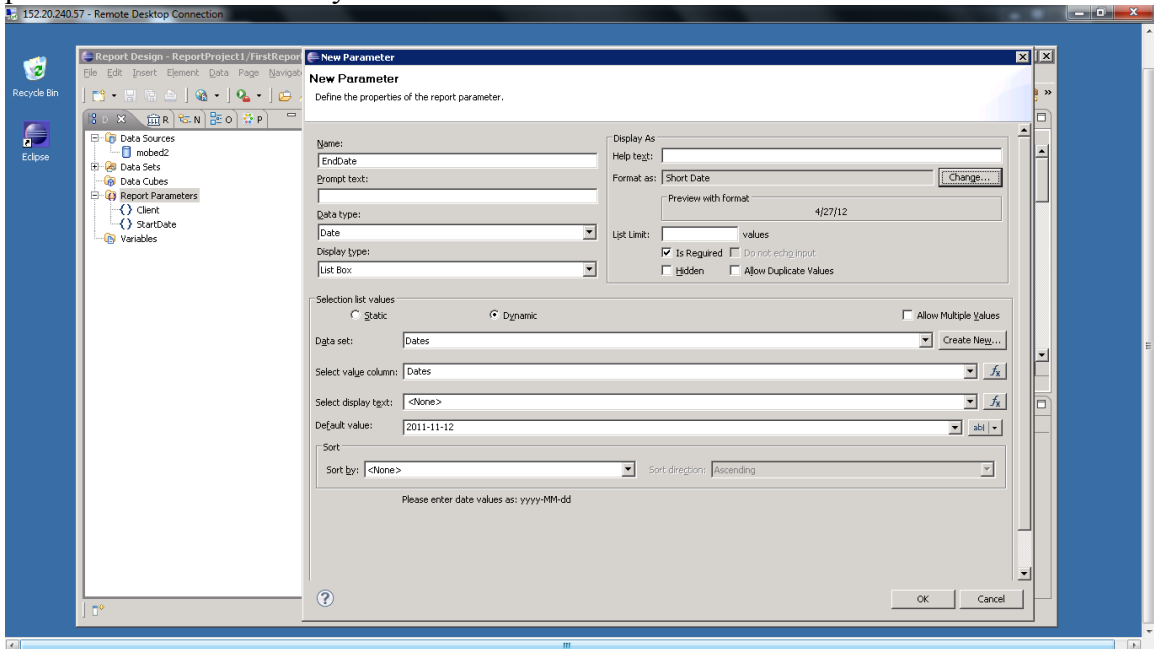
You can change the format of the values by pressing the “Change...” button, and entering the format you would like.



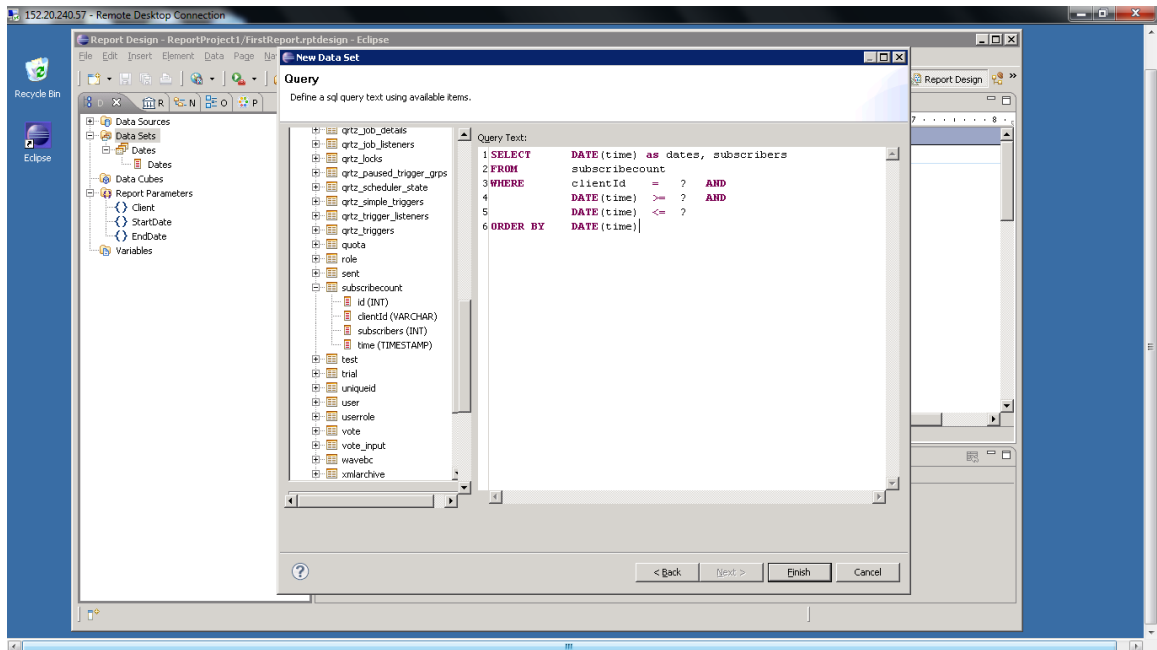
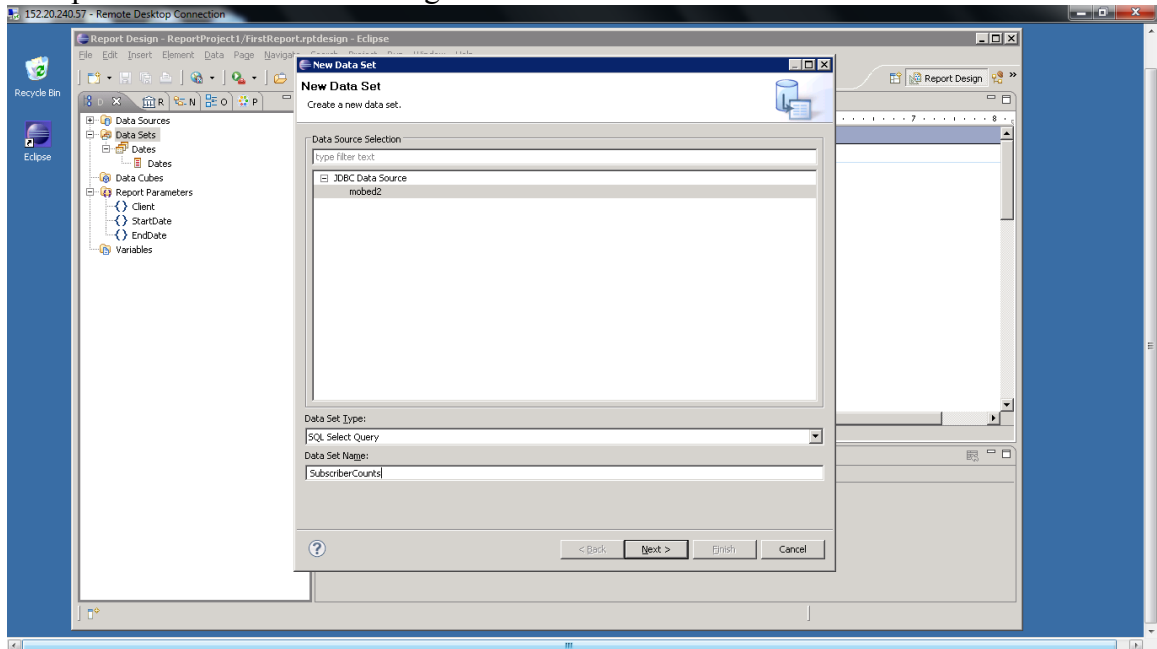
Once you have finished “StartDate”, the form should look like the form below. Remember that a default value is necessary for testing, but will be removed before deploying the report.

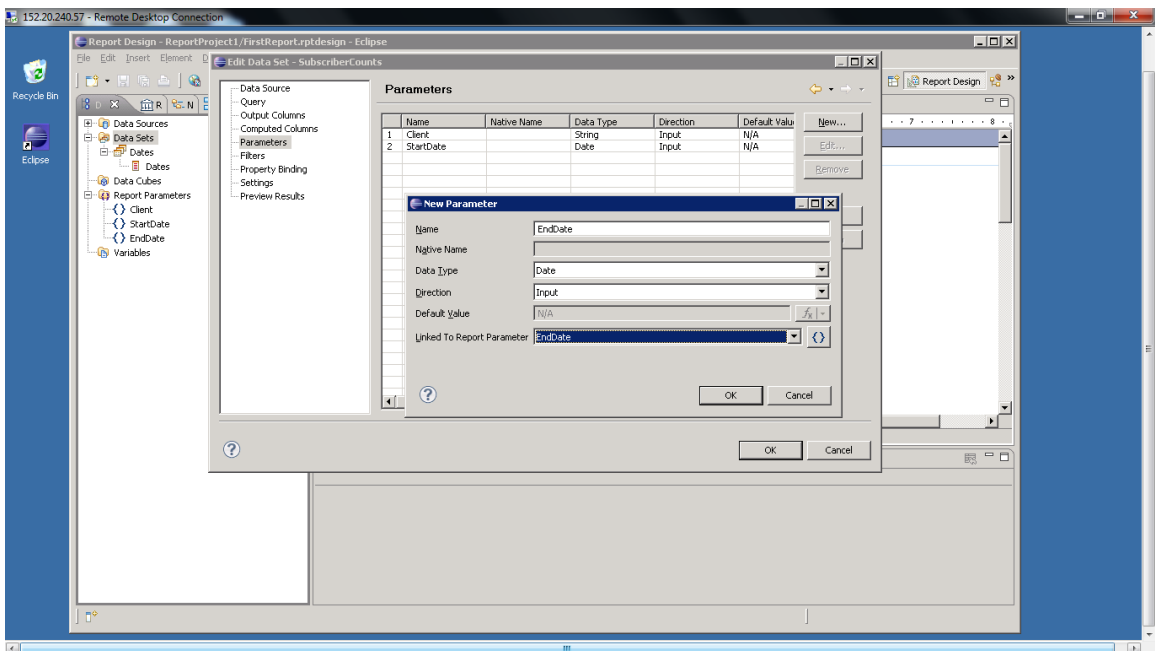
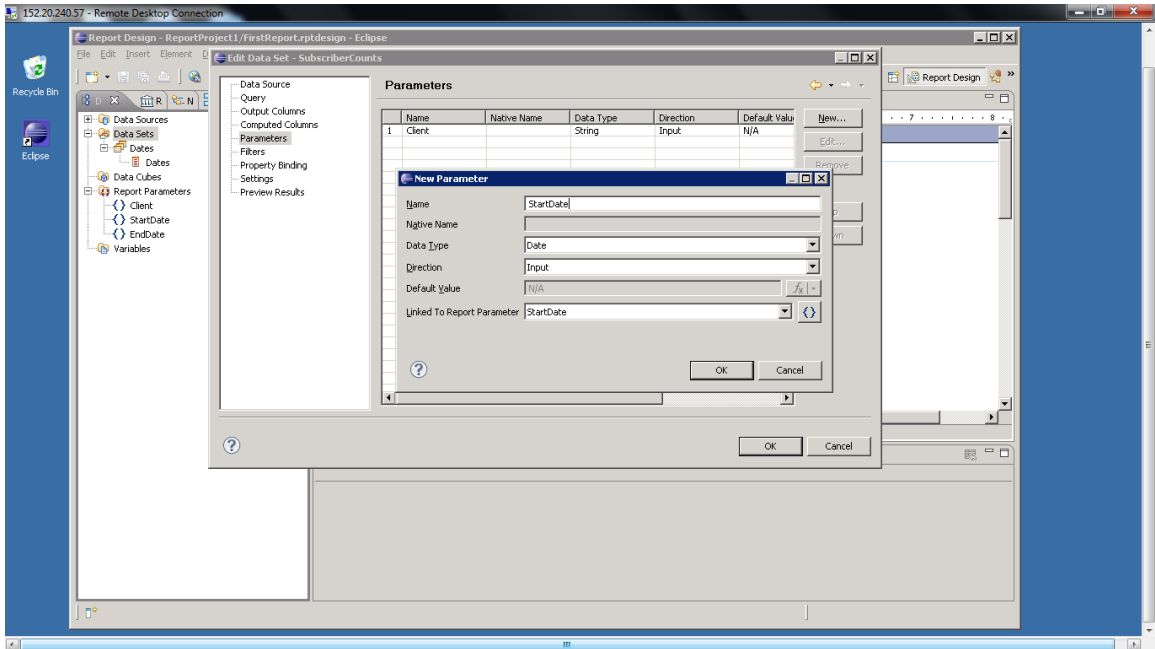


Once you have finished the “StartDate” parameter, you can create the “EndDate” parameter in the same way. The form for this is shown below.



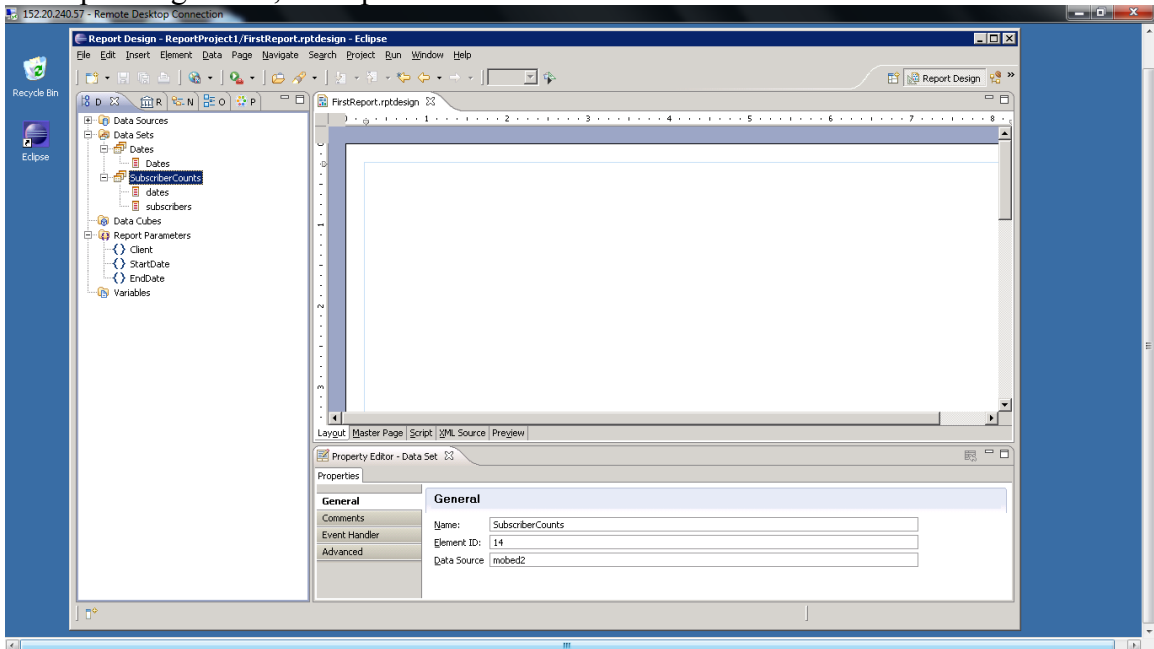
Now that we have our date range, we can create a new Data Set for subscriber counts. This process is shown in the images below.



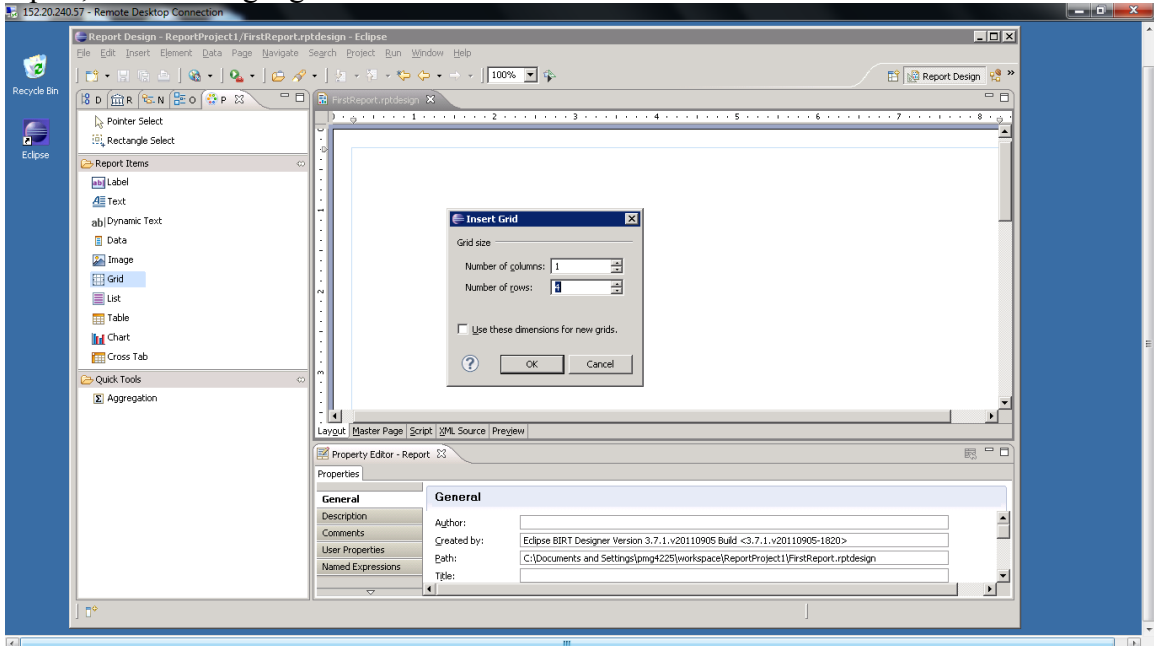


After you have added all of the parameters, the “**Preview Results**” section should give you a table of two columns, “date” and “subscribers”.

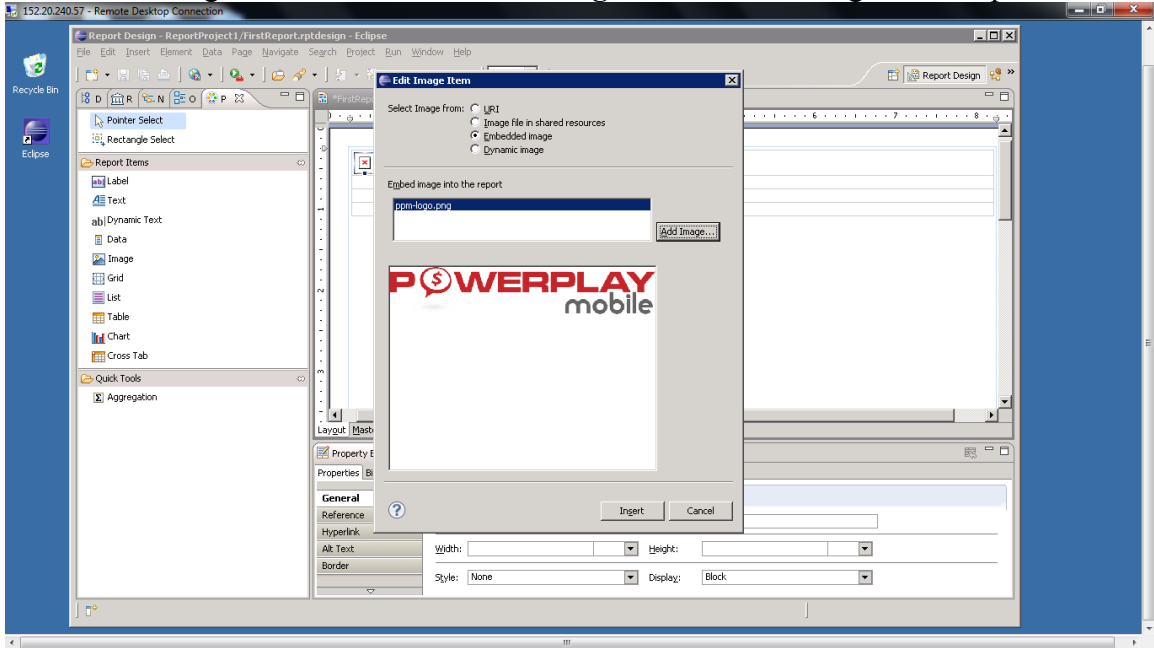
After pressing “OK”, the report should look like the screen shot below.



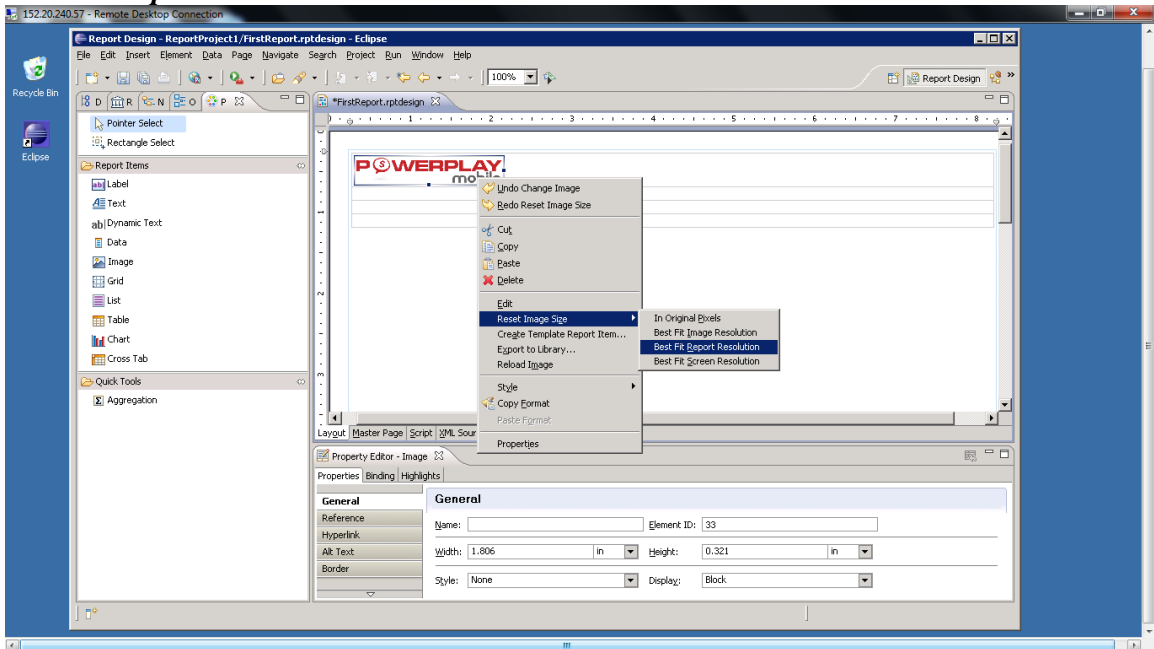
Now that we have all of the data we need, we can begin to add it to the layout. One of the easiest ways to format a layout, is to use a grid. You can add a grid by going to the “Palette” tab, click on “Grid”, and press anywhere inside the report layout. For this report, we are using a grid with 1 column and 4 rows.



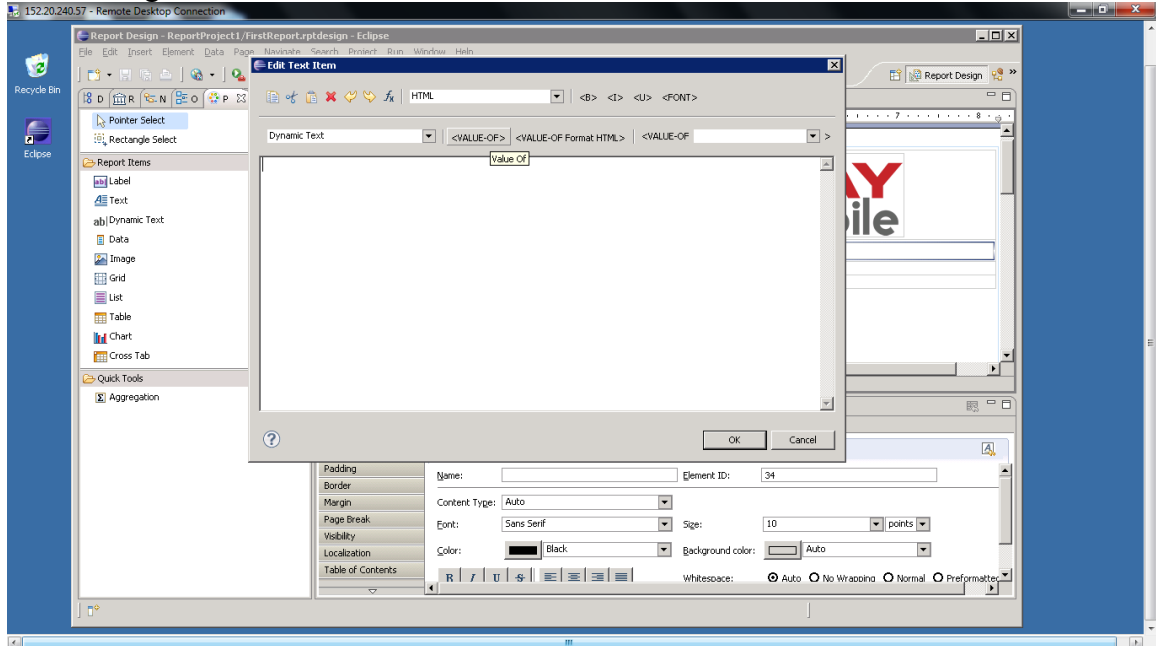
To insert an image into the first row, click on “Image” and then click anywhere in the first row of the grid. Use an “Embedded Image” and add the image to the report.



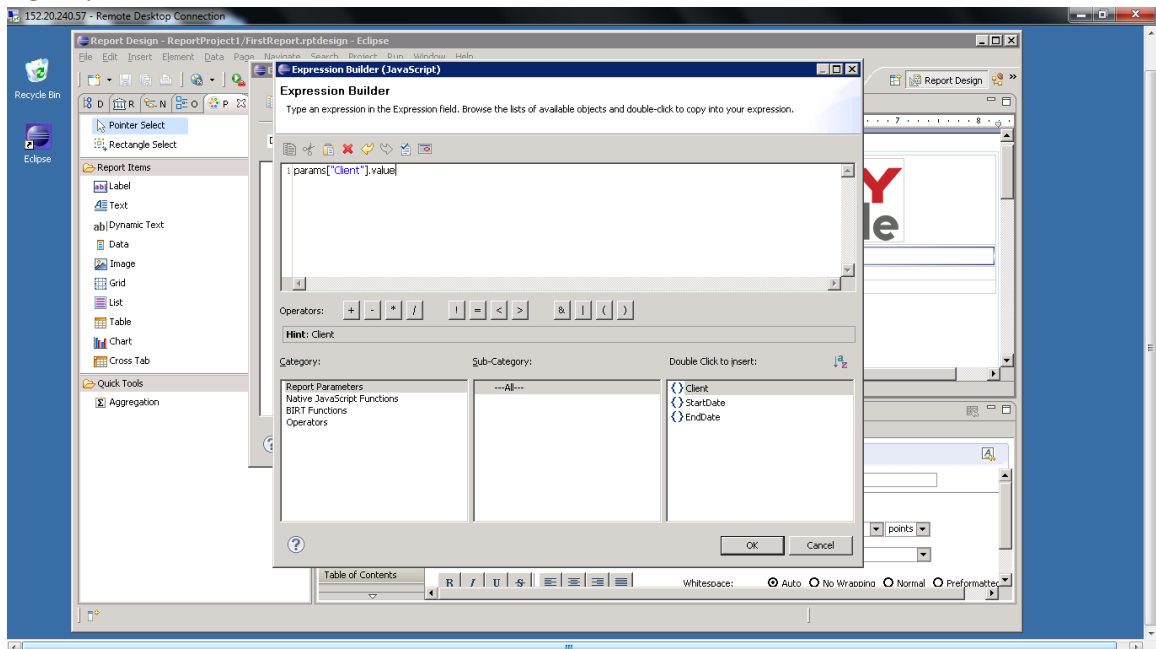
To get the best size of the image, right click on the image and click *Reset Image Size > Best Fit Report Resolution*.



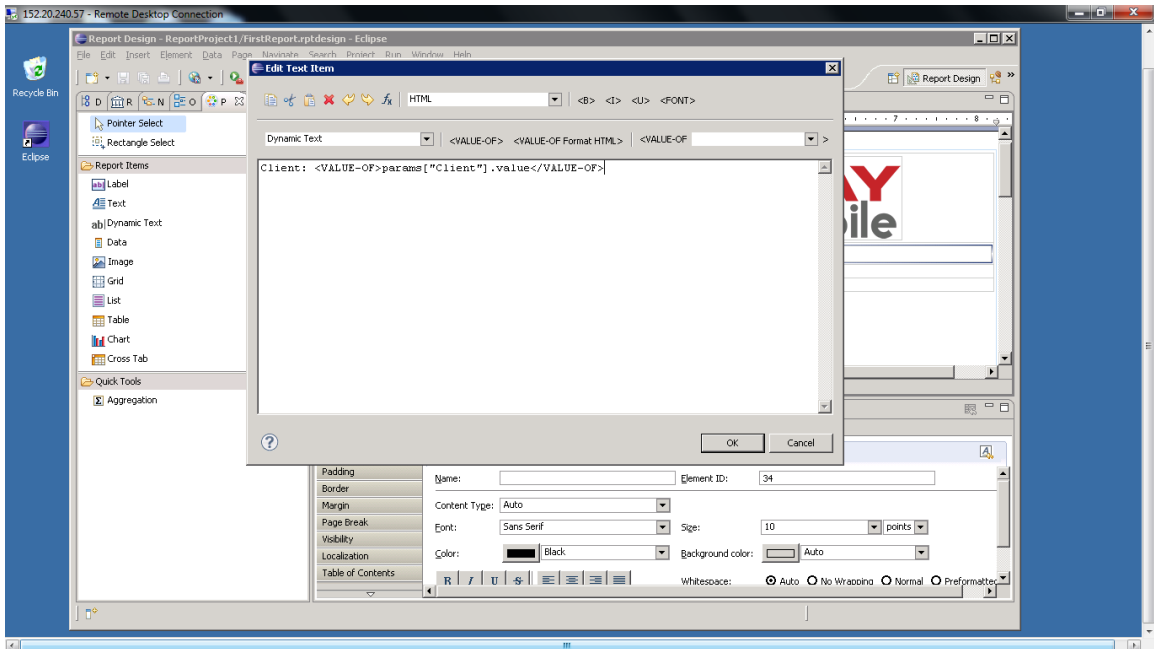
To display the “Client” parameter in the report, click on “Text” and then one the second row in the grid. Make sure the text is set to “HTML” and click on “<VALUE-OF>”.



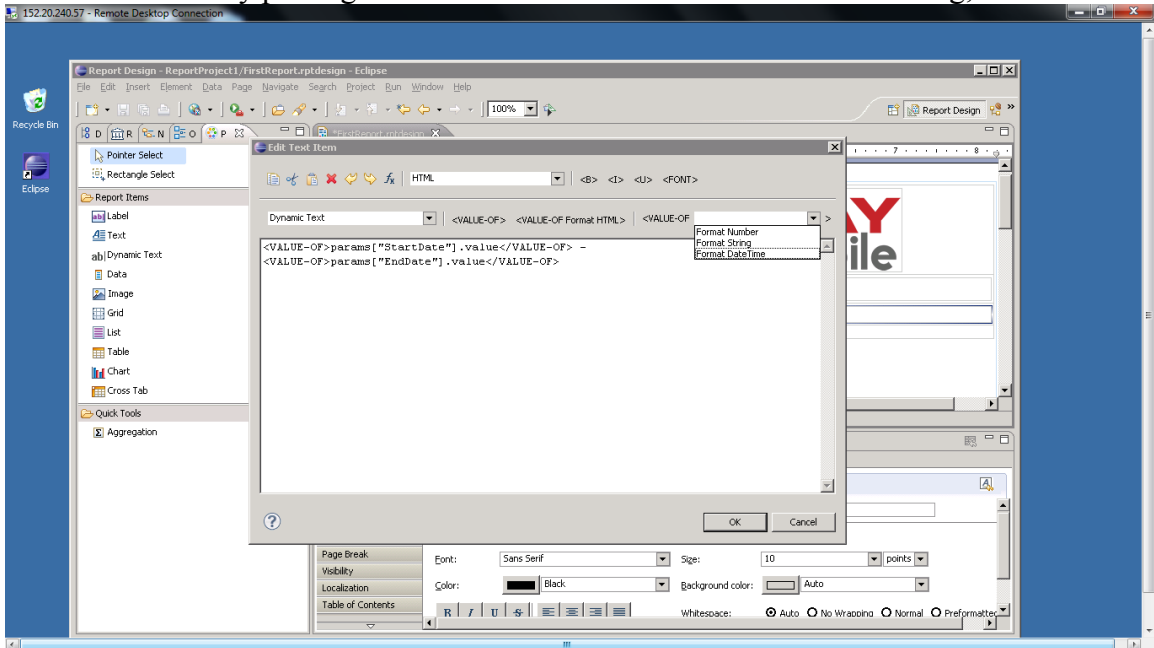
Navigate to the “Client” parameter, and double click on it to insert the value. Then, press “OK”.

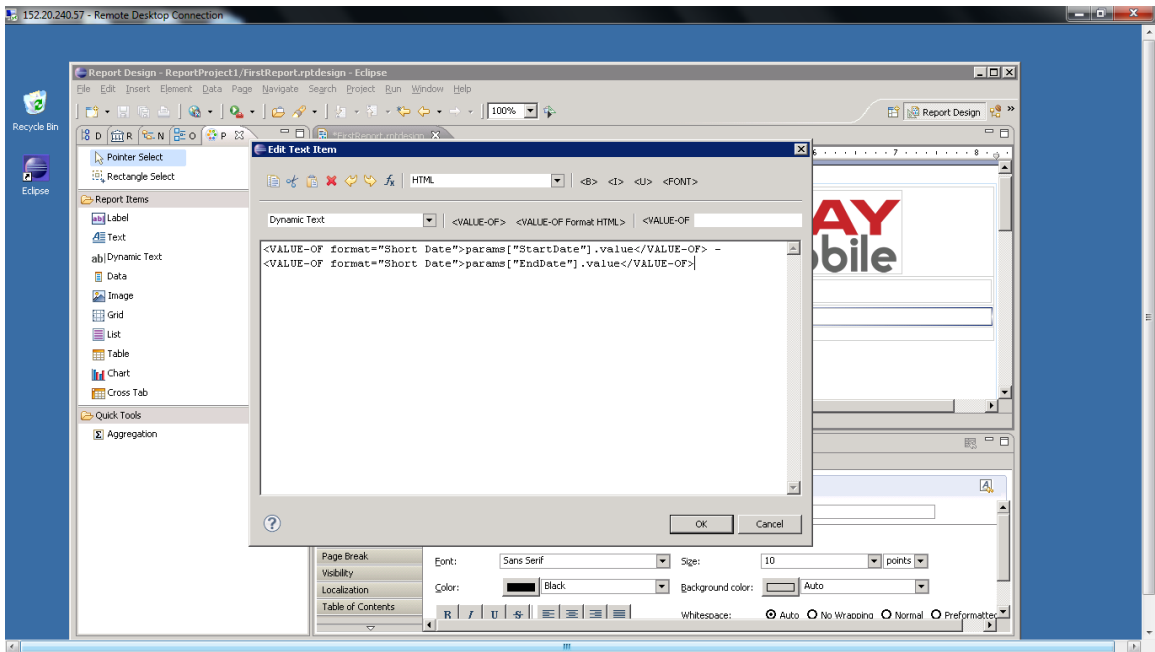
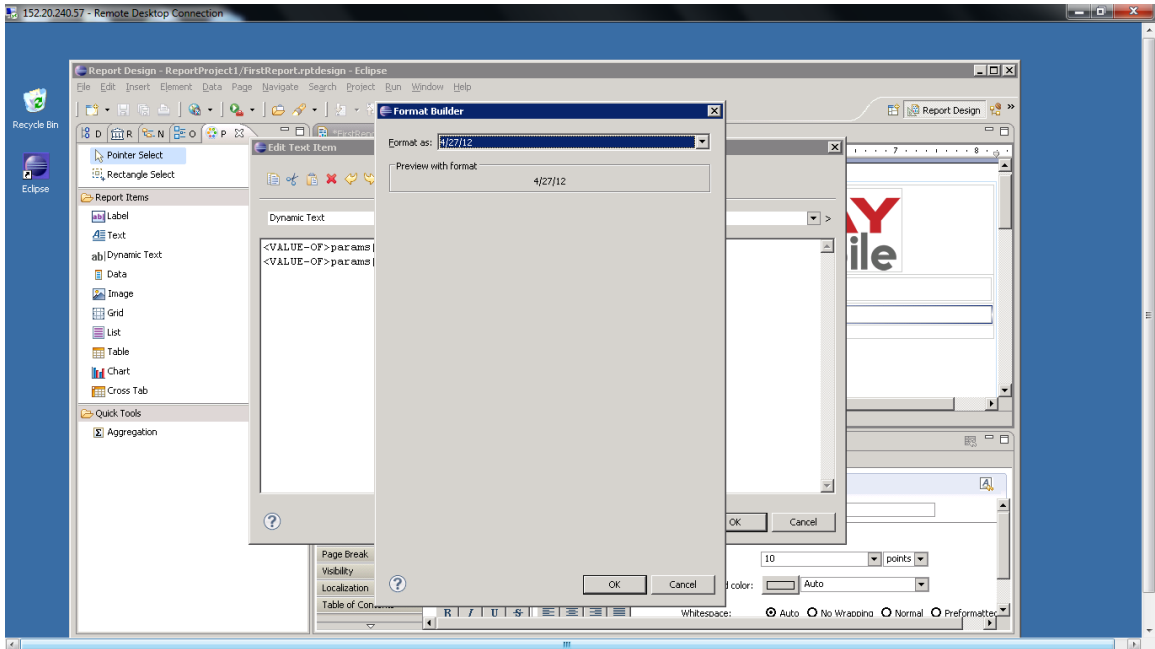


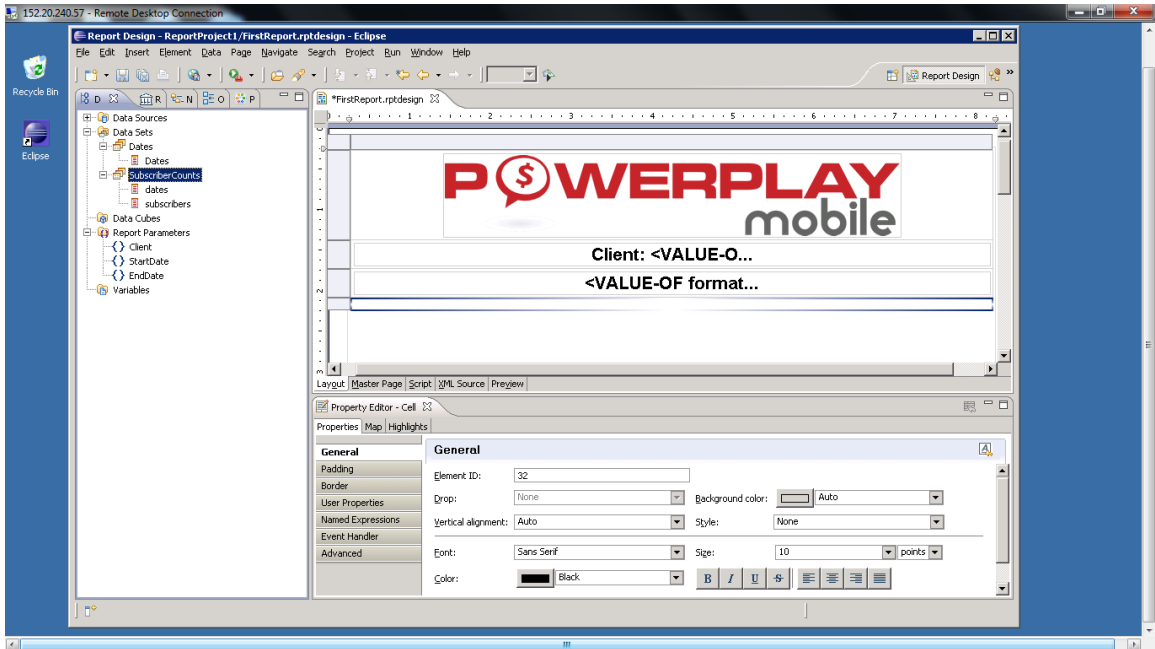
The text should look like the screen shot below. Press “OK” if this is the case.



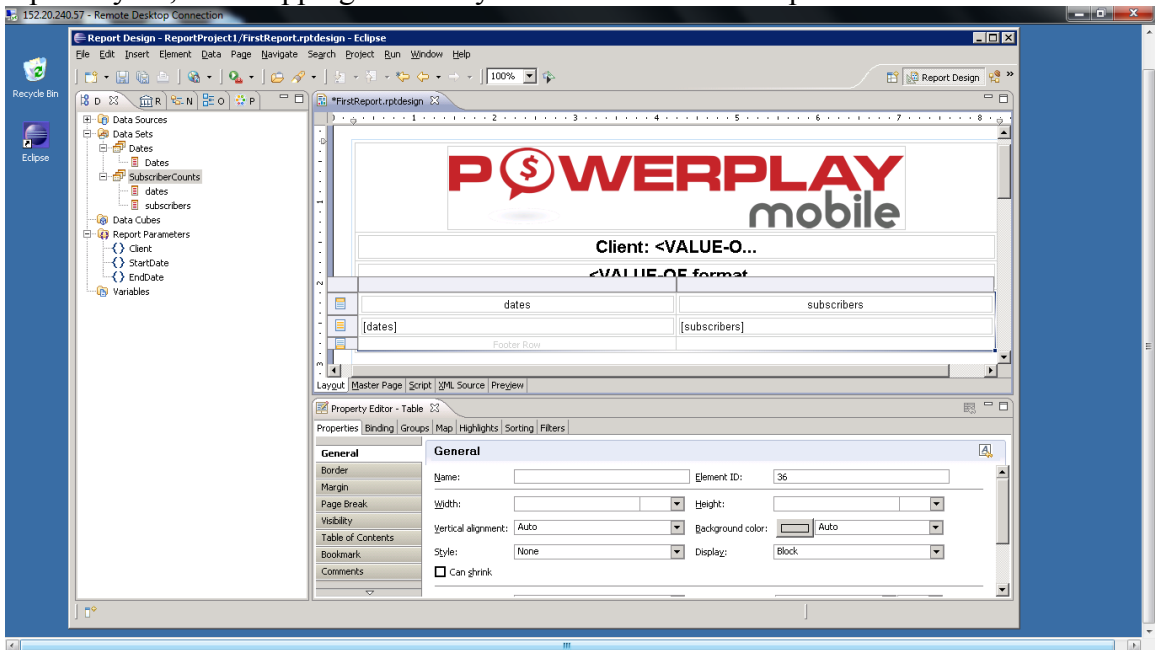
You can create the date range in a similar way in the third row of the grid. You can also format the value by placing a format attribute inside the “VALUE-OF” tag, shown below.



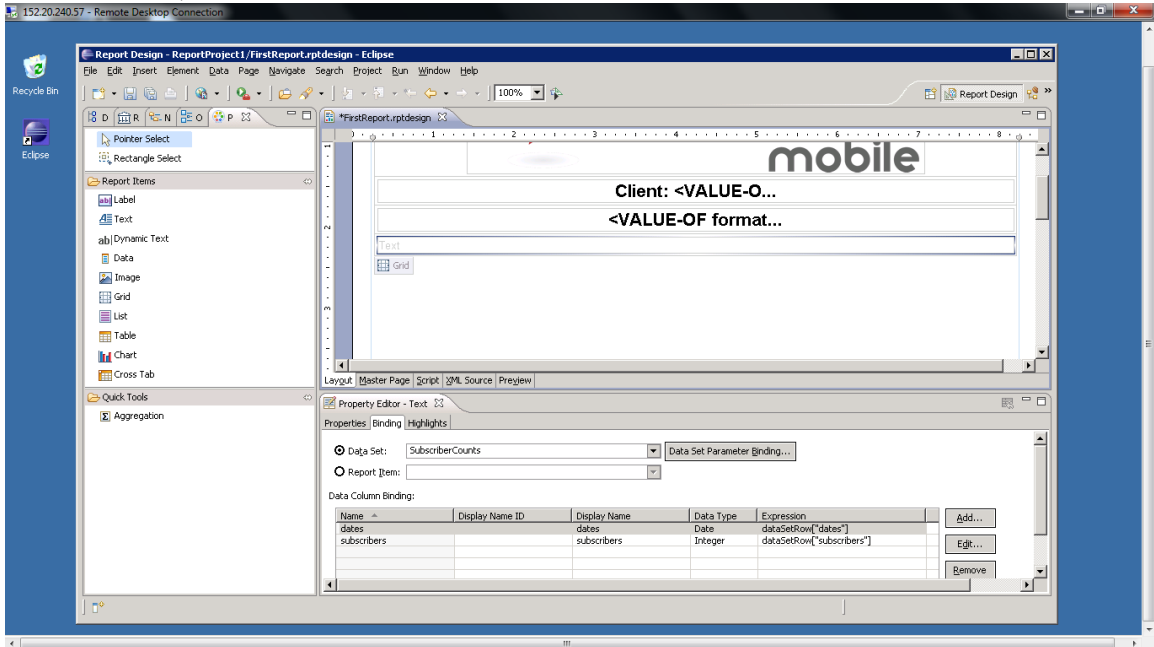




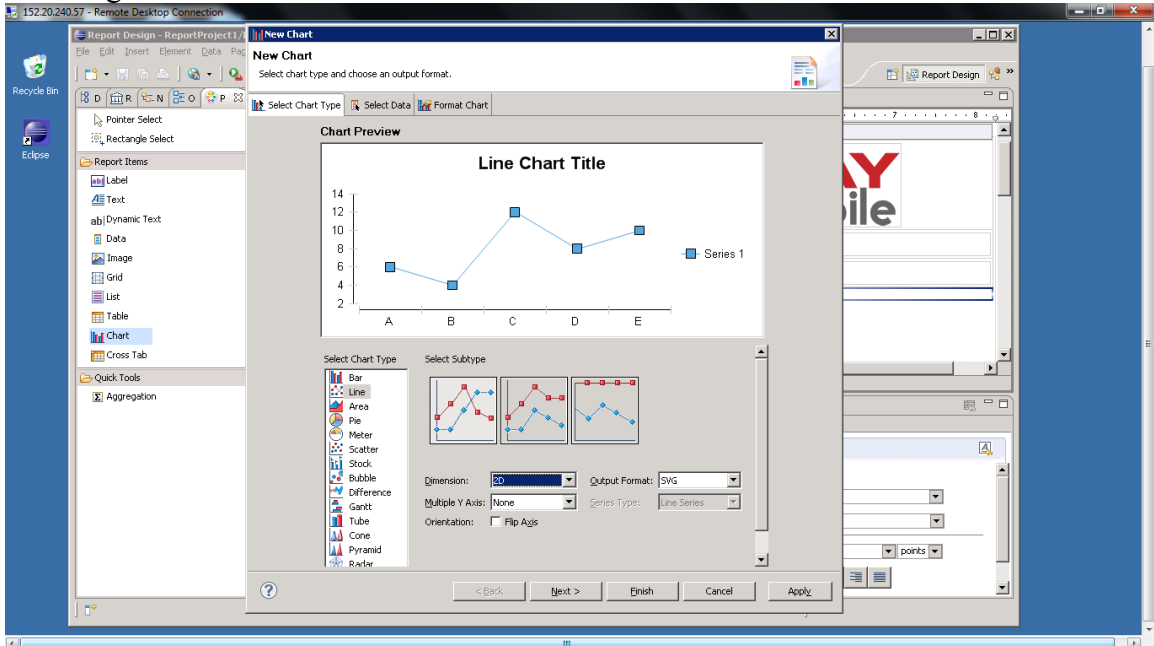
You can add a table of the “SubscriberCounts” Data Set by dragging the Data Set to the report layout, and dropping it where you want the table to be placed.



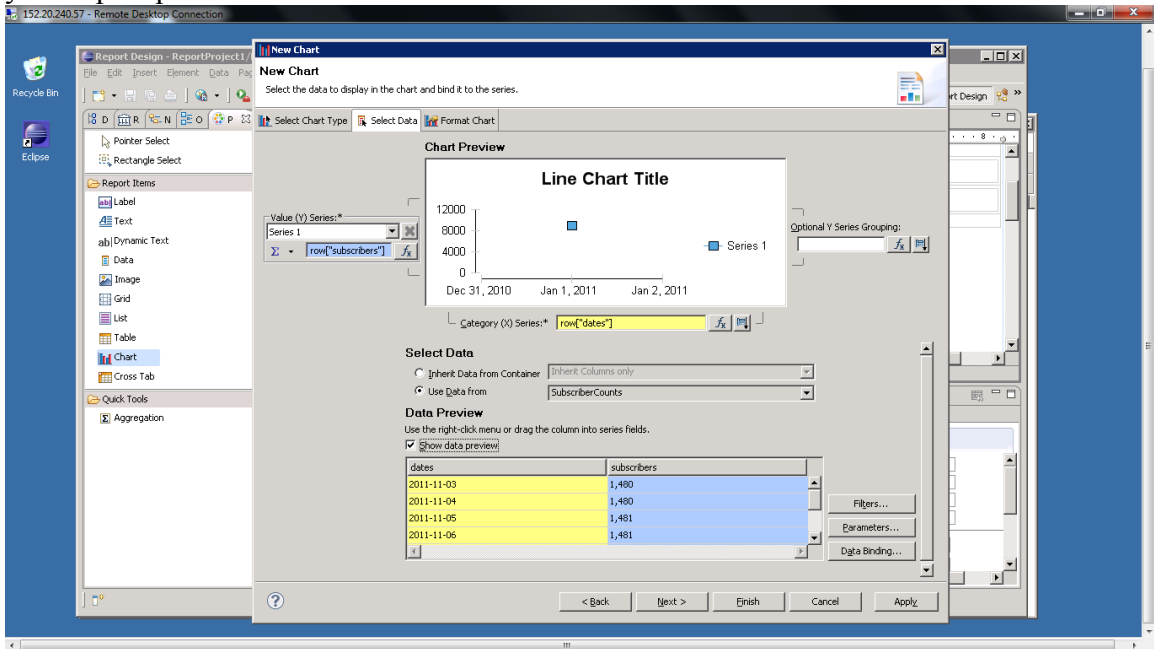
You can also add values from a Data Set to a text block in the same way as a report parameter. However, before you can add the Data Set, you must first bind the Data Set to the text block, shown below.



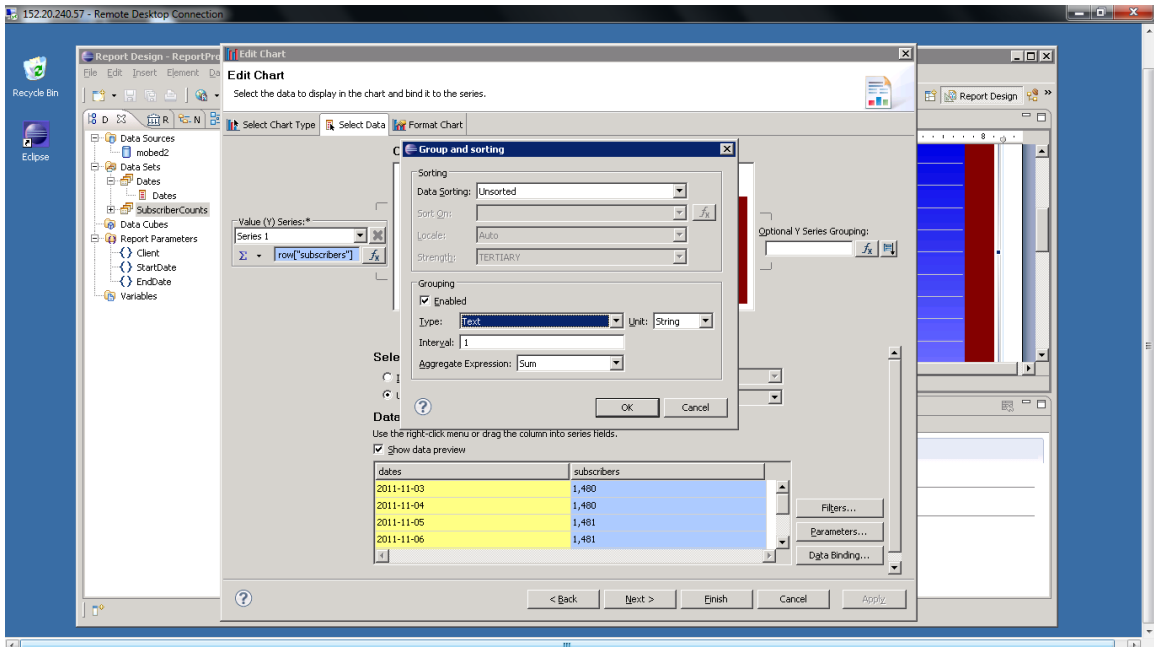
To represent a Data Set to a chart, click on “**Chart**” and choose the type of chart you wish to generate. The screen shot below shows how to create a line chart.



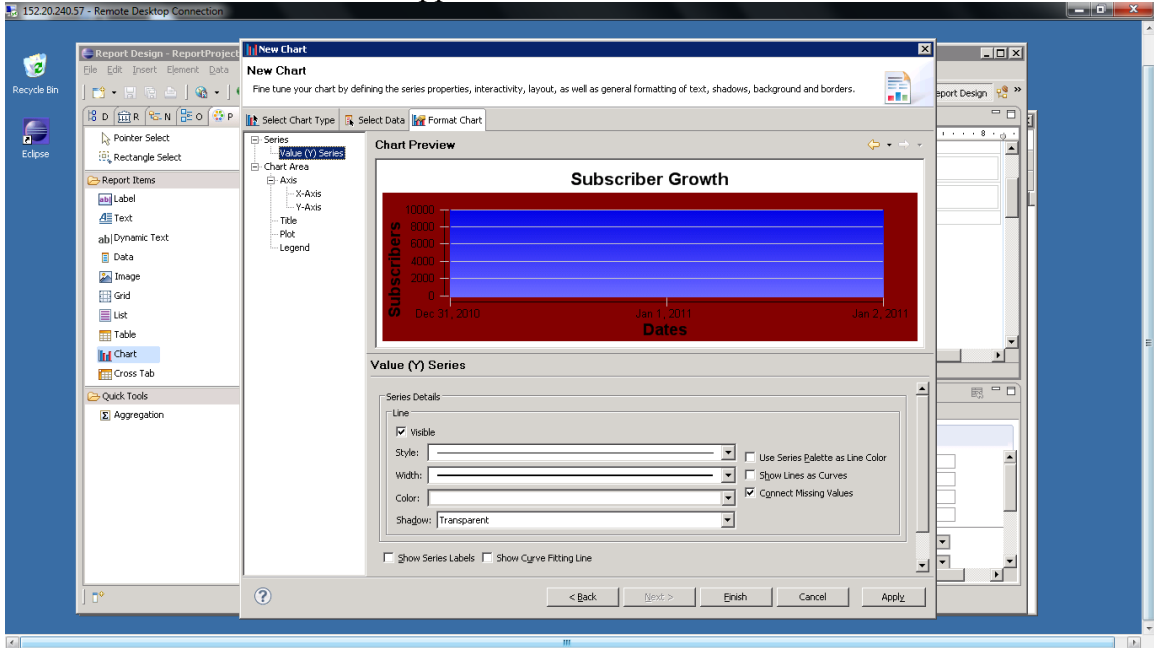
Once you have chosen the type of chart, click “Next” and fill out the form shown below. You should have values in the “Data Preview” section based on the default values of your report parameters.



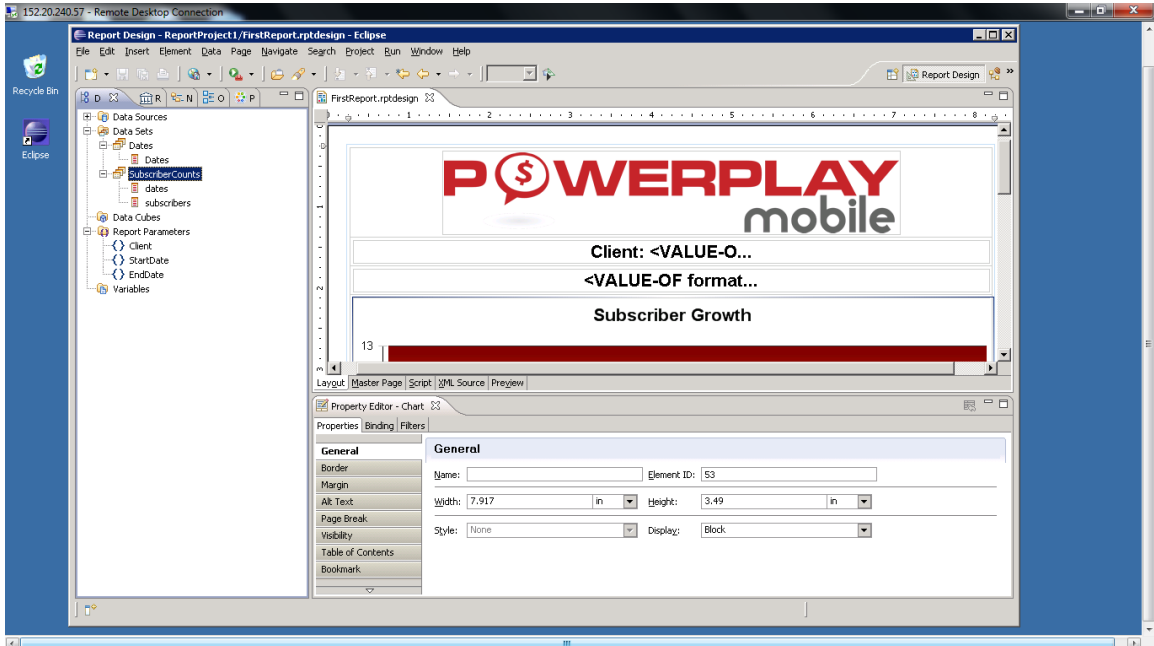
Before continuing to the next tab, click on the second button to the right of the “Category (X) Series” text field. Set the “Type” to “Text”, and verify that the form is field out the same as the screen shot below.



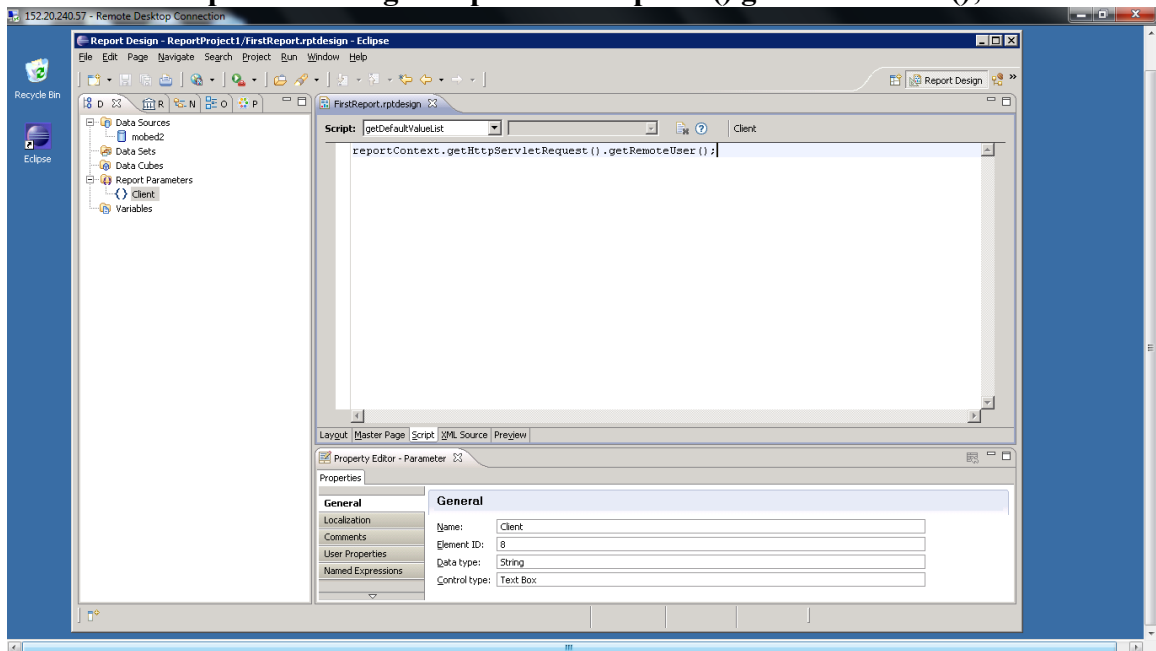
In the next tab, there are several options to change the appearance of the chart. The screen shot below demonstrates some appearance modifications.



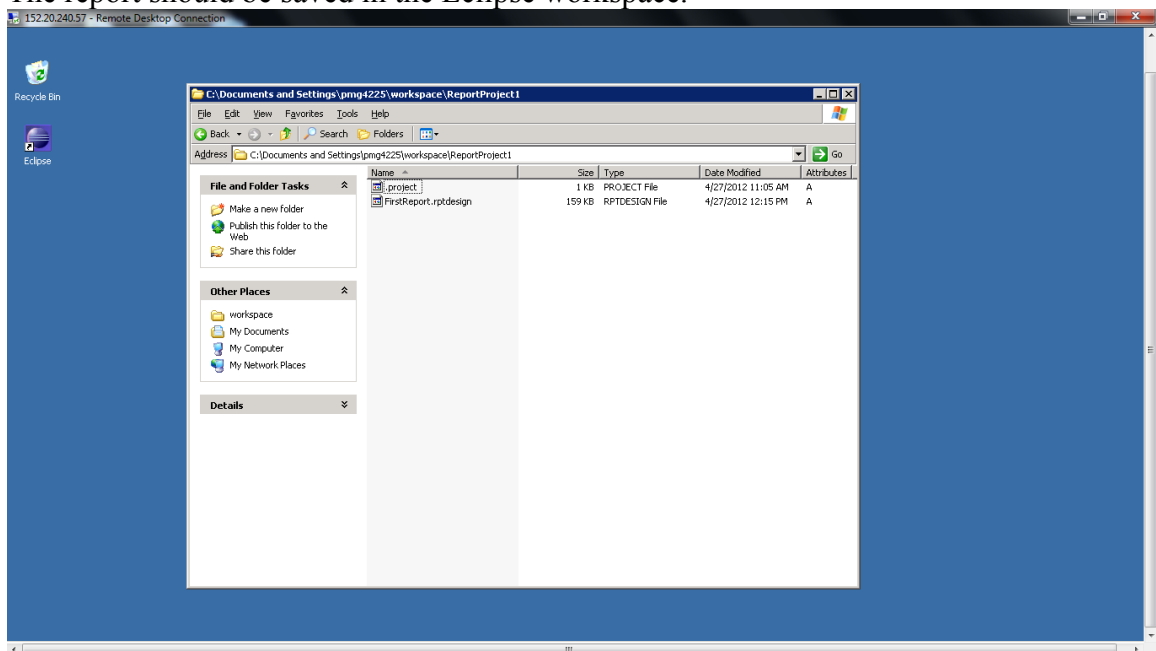
After completing the chart, the report should look like this. To deploy the report to the Apache Tomcat server, remove the default values from all of the parameters and change the Data Source from “152.20.240.67” to “localhost”.



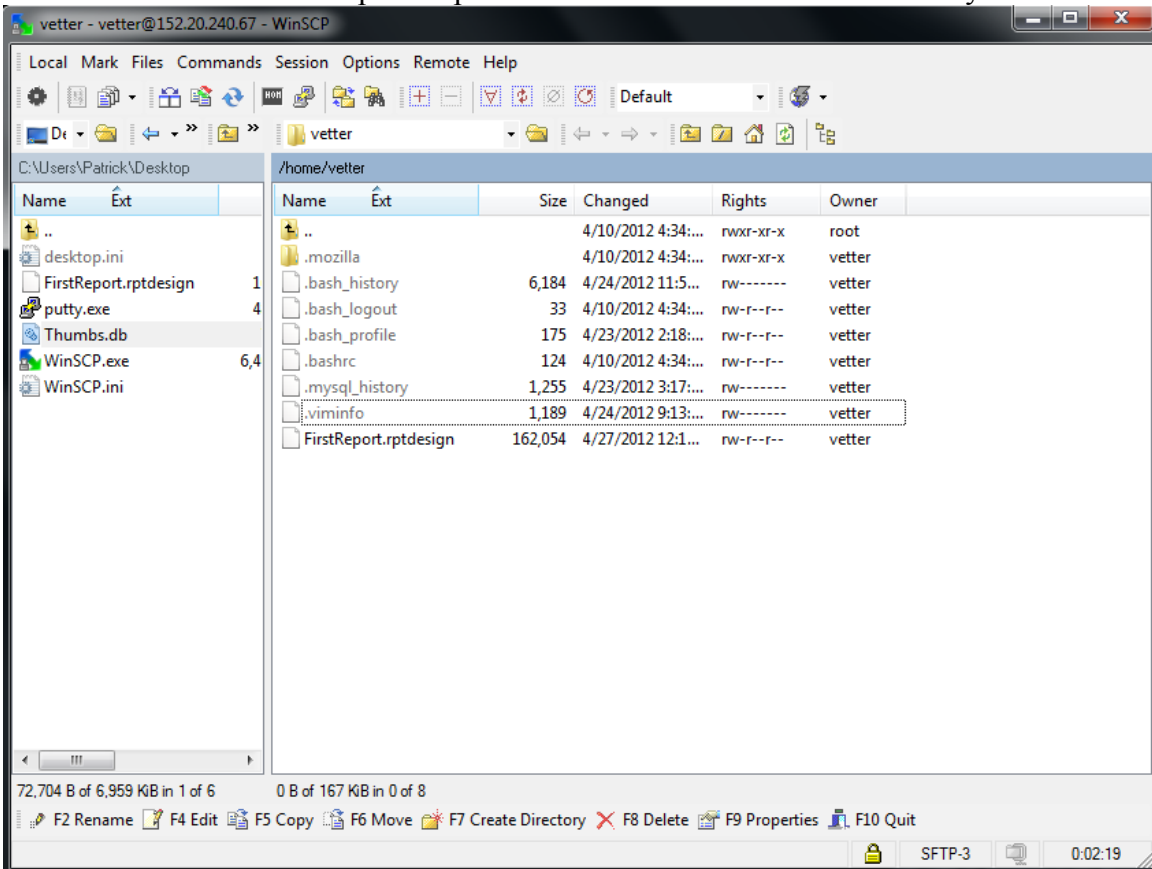
The “Client” parameter also needs to be set to grab its value from the HTTP authentication. To do this, click on the parameter, click on the “Script” tab, and insert the line of code “`reportContext.getHttpServletRequest().getRemoteUser();`”



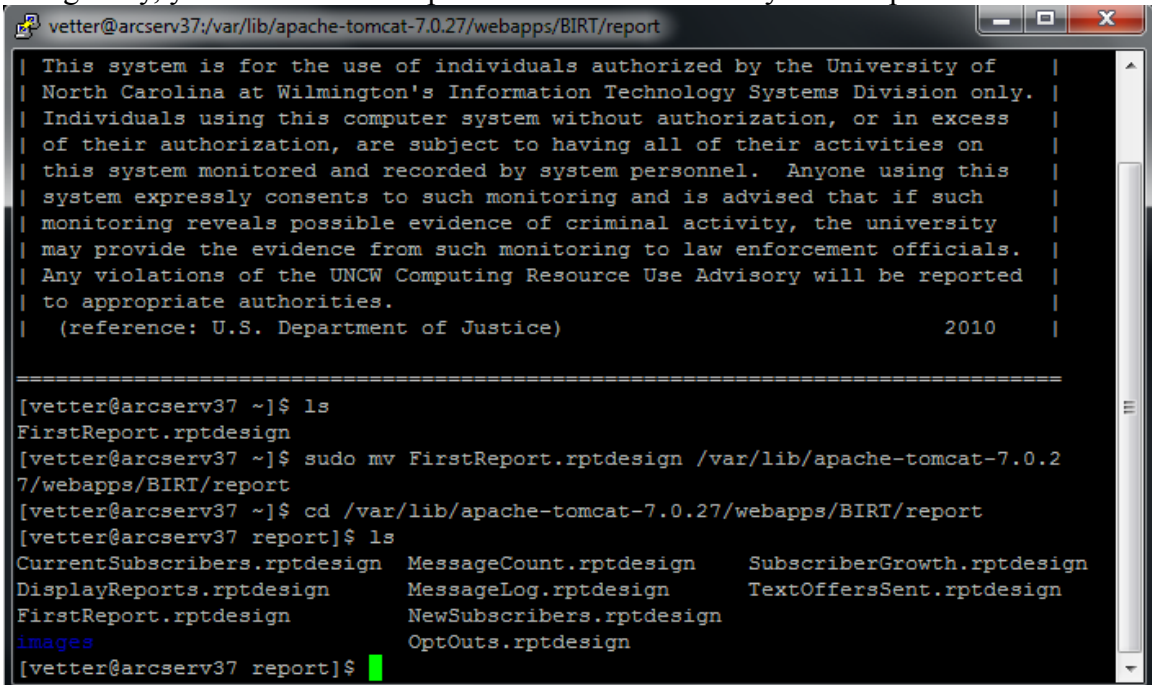
The report should be saved in the Eclipse workspace.



You can use WinSCP to drop the report into the Linux user's home directory.



Using Putty, you can move the report to the correct directory in the Apache Tomcat.



The Tomcat users are located in the tomcat-users.xml file, shown below.

```
vetter@arcserve37:/var/lib/apache-tomcat-7.0.27/conf
[vetter@arcserve37 report]$ cd /var/lib/apache-tomcat-7.0.27/conf
[vetter@arcserve37 conf]$ cat tomcat-users.xml
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="client"/>

  <user username="builders"      password="pass1"      roles="client"/>
  <user username="spa"          password="pass2"      roles="client"/>
  <user username="teamteal"     password="pass3"      roles="client"/>
  <user username="food"         password="pass4"      roles="client"/>
  <user username="uncwsports"   password="pass5"      roles="client"/>
  <user username="sportclips"   password="pass6"      roles="client"/>
  <!-- All users for PowerPlay Mobile's clients go here. -->
</tomcat-users>
[vetter@arcserve37 conf]$
```

You can start Apache Tomcat with the “**startup.sh**” file, shown below.

```
vetter@arcserve37:/var/lib/apache-tomcat-7.0.27/bin
[vetter@arcserve37 conf]$ cd /var/lib/apache-tomcat-7.0.27/bin
[vetter@arcserve37 bin]$ sudo ./startup.sh
Using CATALINA_BASE:   /var/lib/apache-tomcat-7.0.27
Using CATALINA_HOME:   /var/lib/apache-tomcat-7.0.27
Using CATALINA_TMPDIR: /var/lib/apache-tomcat-7.0.27/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /var/lib/apache-tomcat-7.0.27/bin/bootstrap.jar:/var/lib/apac
he-tomcat-7.0.27/bin/tomcat-juli.jar
[vetter@arcserve37 bin]$
```

To run the report, the URL is:

[http://152.20.240.67:8080/BIRT/frameset?\\_\\_report=report/FirstReport.rptdesign](http://152.20.240.67:8080/BIRT/frameset?__report=report/FirstReport.rptdesign)

The screen shot below demonstrates this report logged in as “**sportclips**” with StartDate = 11/3/11 and EndDate = 11/12/11.

