Liability Driven Investing – Investment Process

JP Morgan Chase & Co.

A Major Qualifying Project submitted to the faculty of

WORCESTER POLYTECHNIC INSTITUTE (WPI)

in partial fulfillment of the requirements for the

Degree of Bachelor of Science

On December 20, 2014

Written by:

Tianhai Chang

Qiao Li

Saad Riaz

Submitted to:

WPI Advisors: Jon Abraham, Xinming Huang & Kevin Sweeney

JPMC Sponsors: Manish Ghayalod, Gautam Manvar, Ashish Tiwari, Sivaraman Krishnamurthy & Mohankumar Rajagopalan

Abstract

Companies need assistance with retirement plans in order for them to pay off their liabilities. JP Morgan Chase & Co. is a corporation that encompasses a wide range of financial aspects. Retirement plans is an important one. It is vital for JPMC to be able to make such monetary facets easily understandable by its clients, taking into account the investments, economic market and various other things. Our group worked on the investment process that is driven by the liability costs of the JPMC's clients. This process involves a lot of intricate steps including making up to date client portfolios, running fast processes and having a manageable data in store at all times. Each one of us worked in a different area; The Funds of Funds is where client portfolios are created, in the Liability Driven Investment the discounted liability is calculated, and the Data Warehouse is what provides the data. The work we did aids the Client Portfolio Managers in communicating with the clients, to make sure that the clients get correct and up to date information.

Authorship

This Major Qualifying Project was completed after collaboration from all three members of the group. Specifically, Qiao worked alongside two JPMC employees in the Funds of Funds group, while Saad worked with couple other employees on Databases. Both of these groups were part of a bigger group called Global Multi-Asset Group. Tianhai and Saad also worked on the Liability Driven Investment group, which deals more with the outlook for the Client Portfolio Managers. There was a significant amount of work divided equally amongst all three, and only the LDI overlapped. The report and the presentation as well were completed through a combined effort by all three group members.

Executive Summary

Our project is about retirement plans. We know companies need retirement plans in order to pay off their liabilities. There are two kinds of retirement plans that the companies normally offer including the Defined Contribution Plan and the Defined Benefit Plan. A Defined Contribution Plan is a type of retirement plan in which the employer, employee or both make contributions on a regular basis for the employees' own benefit. The Defined Benefit Plan on the other hand is a type of pension plan in which the employers promise a specified benefit amount on retirement. Unlike the DCP, which depends directly on individual investment returns, the DBP is more risky for employers; therefore the goal of this project was to assist JP Morgan Chase & Co. (JPMC) in the investment process involved that helps clients manage their Defined Benefit Plans.

In order to fulfill an obligation and helping companies reach the commitment they have made with their employees, we need to know what to invest in, when the right time to invest is, and where exactly to make these investments. This investment process at JPMC has been divided into three divisions including the Funds of Funds (FOF), the Liability Driven Investment (LDI), and the Data Warehouse. Funds of Funds creates portfolios that manage assets to reach the liabilities; Liability Driven Investment takes the future obligations and find out the discounted liability value. Liability in this case is the amount that is present on the budget sheets of the clients that they want to reach by the end of certain period of time, and asset is the amount or whatever the company keeps aside that they want to grow in order to reach that liability cost. Both the FOF and LDI need data from Data Warehouse to calculate and meet the liability costs.

Within these three divisions, Qiao worked within FOF to build a fee calculator, Tianhai worked in the LDI group to construct a market data page. Both of these tasks help analyze the scenario behind the investment process that is driven by the liability costs of the company. In order to make sure that the data used in these scenarios is of a high quality, Saad worked with a team in the Data Warehouse to perform operational re-engineering within JPMC's Oracle database. Additionally, in order to further improve the processes that take place, Tianhai and Saad also worked together to write a new procedure, including some JUnit testing, to detect bugs within the code called regression testing.

In FOF, to make a fee calculator, the fee data was added into JPMC's vast Oracle database and this data was then displayed at the front-end user interface. The fee calculator was built separately. This has the capability to show a fee percent per portfolio. This allows the Client Portfolio Managers to show their clients how much fee is going to be charged to their accounts. It is important to know this fee, since it is a negative value added to the total asset growth when calculating the required amount, which is needed in order to reach the liability. It is a continuous cost that the clients (companies) pay to JPMC, and the fee calculator makes

is easier for the CPMs as well as the company to stay on top of their costs and other balances. Due to the fee calculator, the CPMs will not have to manually note down the fee and calculate it.

Within the Liability Driven Investment group, a new web page was developed on a pre-existing software. The current LDI Web Workstation consists of four different pages, each of which has a different function. These functions include calculating the liability of a company and showing graphical representations and summary of the company's accounts and also some archived information. This LDI Web Service is constructed by web developers for the use of CPMs in their communications with the clients. A new page called *Market Data* was constructed, which has the ability to give the CPMs the opportunity to upload the data by themselves into the database and view it in graphs and other forms by simply clicking an upload button.

As mentioned earlier, both of these processes require company's data including various types of account information. This data is stored within the Oracle database that JPMC uses. This is one of the biggest databases utilized by any company as JPMC is one of the largest hedge funds in the world, meaning it has a significant number of clients. We also worked on improving the quality of this data. This is called Operational Re-engineering, since the way data is represented is changed, but nothing new is added. Beforehand, some specific data present within the database was available as two different tables for United States and Europe each. There are a number of tables that had both a US copy and an EMEA (European) copy. This meant that even though some information was different, a lot of the information stored was the same, and caused overlaps. This in turn causes the processes to run slower, and difficult to manage. Hence, the tables were merged to prevent the overlaps, and make the data within the database cleaner and easy to manage. All of this work was done using the Oracle SQL Developer.

Lastly, a new method to check for errors within the written code was developed. Since JPMC is a large corporation, a lot of different developers work on the same code at different times. It is important that the initial code, or the basic coding procedures remain the same, and the calculations done are also the same. Hence, a new method called Regression Testing was developed to ensure that the code has not been changed between two time periods.

Overall, the project involved various challenges, and unlike having one overall goal, our team tackled with many different objectives within the project with the main goal focused on helping the Client Portfolio Managers. Various tasks were performed, and many different programs and software languages were utilized to ensure the individual goals were met.

Acknowledgements

We will like to thank the following individuals and organizations for their continuous support and contributions to make this a successful project.

- The project lead sponsor Manish Ghayalod for helping us understand our project goal, providing constructive criticism throughout the project and helping us construct our final presentation.
- The GIM Solutions team at JPMC including Gautam Manvar, Ashish Tiwari, Sivaraman Krishnamurthy, Mohankumar Rajagopalan, Alex Yemets, Meredith Sells, Ashwini Rupert and M. Mazar Mushtaq for providing us resources throughout the project, helping us develop our solutions and providing us with direction and technical advice.
- The project advisors Professor Jon Abraham, Professor Xinming Huang and Professor Kevin Sweeney for their assistance and guidance in helping us develop our project, and write this report.
- Michael Shaddow for helping us settle in the new environment and being an excellent liaison between JP Morgan and WPI throughout the project.
- The Interdisciplinary and Global Studies Division of Worcester Polytechnic Institute for organizing the logistics of coming on-site for the project.
- Worcester Polytechnic Institute for providing us with the opportunity to have a remarkable experience at the New York City Wall Street Project Center.

Table of Contents

Contents

Table of Figures

1. Introduction

JP Morgan Chase & Co. has various divisions, and our team worked specifically with the Global Multi-Asset Group within the Asset Management division. The group continuously works on making the lives of the Client Portfolio Managers easier, by creating programs that can be utilized by them, and also ensures that the information being provided to them is accurate. So when the CPMs communicate with the JPMC clients, they have the correct and up-to-date information.

The investment process involved in making the clients reach their liability costs includes various different parts or steps. The GMAG team works in this process to provide the clients better solutions for their issues or any obstacles they face. The objective of our team was to work as a part of this group and help them achieve their goals.

When we joined the project team, they were in their final month, or as they called it 'Final Sprint'. It was critical to achieve the enlisted goals by the end of the year, and everything remaining within the project had been given a high priority. Hence, our presence was an additional help to the group as we were able to work with the team and provide them with additional IT support. We assisted the team in a wide variety of things and different aspects, all of which are taken into account by the other team members. The team consisted of twenty members, who worked from various locations including New York City, New Jersey, as well as India. Through the usage of conference calls every day, everyone was kept up to date, and individual goals were achieved daily.

2. Background

2.1. J.P. Morgan Chase & Co.

J.P. Morgan Chase & Co. (JPMC) is currently the largest bank in United States of America, and the 6th largest bank in the world with about $2.4 trillion dollars in assets. It operates in more than 60 countries and has over quarter million employees. JP Morgan is a leader in investment banking, and also provides banking and financial services to the customers, small businesses and many of the prominent corporate and governmental clients all over the world.

J.P. Morgan & Co. was a result of combination of JPMorgan Chase and Morgan Stanley; these two have been established as financial institutions for over a century and were one of the largest global banking institutions. It was started by one of the most powerful bankers of his era, John Pierpont Morgan, who also helped in the organization of U.S. Steel and General Electric. There were various reasons for his success, but the reasons for the company's survival and growth are advocated to the fact that he was able to sustain the American financial markets significantly during several economic crises in the early 1920s. J.P. Morgan was able to set up the first central bank of United States that now manages 30% of the world's financial transactions.

The JPMC was founded in 2000, after the merger of J.P. Morgan & Co. and Chase Manhattan Corporation. Within a financial institution there are a many groups that help the institution provide more services as compared to a normal bank that acts as a money holder. JPMC is currently divided into two separate entities, J.P. Morgan and Chase, where each serves a different purpose. JP Morgan is responsible for various businesses including investment banking, asset management, private banking, private wealth management and treasury and securities services. Chase, on the other hand, provides credit card services in North America, as well as commercial banking.

JP Morgan Chase & Co. has for a long time remained a prominent member and contributor of the financial industry. It plays a very significant role in the international market, whilst JPMC Chief Executive Officers serve as some of the most influential businessmen and women in the world. (Annual Report, 2005, 2006)

2.2. JP Morgan's role in investment

JPMC plays a significant role in investment. The hedge fund, a portfolio of investments that applies different investment strategies to reach the goal of generating high returns, unit of JPMorgan Chase is currently the second largest in Unites States. This involves investments both internationally and domestically.

Companies come to JPMC to ask for assistance in investing their money. JPMC also aids other international financial institutions to invest within the US. In order to gain significant returns, there are teams within JPMC that manage the investment opportunities depending on the reasons of investment. This project is applied only to the Asset Management division of JPMC, which is the major investment entity of the company.

2.2.1 Divisions within Asset Management

Asset Management involves using assets as investment opportunities, and helping companies, institutions or governments to invest in various different assets. Assets are the where money can be invested, and returns can be expected. There are many different types of assets that the Asset Management division of JPMC invests in; including equity, fixed income, cash liquidity, currency, real estate, infrastructure, hedge funds and private equity.

Additionally, JP Morgan Chase & Co. plays an imperative part in the asset industry, with assets under management of $1.7 trillion (as of September 2014).

According to Forbes, the current Chief Executive Officer of J.P. Morgan Asset Management, Mary Erdoes, is currently the 66th most powerful woman in the world. The Asset Management Technology Team is responsible for the technical issues, advancements and programming within J.P. Morgan Asset Management sector. In order to run a well-organized corporation, this sector was recently divided into five business areas, as shown in the Appendix A.

GIM Solutions manage nearly $80 billion for clients, and includes for different groups. The Global Multi-Asset Group (GMAG) is regarded as a pioneer in asset allocation solutions space. It offers diversified investment solutions for some of the world's leading pension plans, corporations and institutional investors. It also offers strategic advice to Chase. Institutional Asset Management Group is a go-to resource for helping institutional client advisors deliver a robust analysis, pertinent research and tailored approaches to their clients. Global Access Portfolios (GAP) includes a Global Access Investment Team that provides solutions to private clients and institutions in various countries. They take a careful and sophisticated approach to asset allocation across different asset classes. Lastly, J.P Morgan Advisory Program offers innovative and well managed solutions. (Analyst Training Series, 2014)

Our project is part of a plan that the Global Multi-Asset Group has been conducting within the GIM Solutions.

Global Fixed Income is an investment management area that has more than 270 investment professionals and about $843 billion in assets under management worldwide. The commitment to clients and in-depth research provided by this sector has made JPMC one of

the most recognized global leaders in fixed income investment management. ***Global Equities*** is globally focused on providing global insights with local delivery. The teams have portfolio managers, research analysts and traders that expertise in small, medium and large market capitalization equities. It includes more than 440 investment professionals. ***Global Alternative Investment*** has more than $150 billion in AUM and different other engines across Global Real Assets, Private Equity, J.P. Morgan Alternative Asset Management, Highbridge and Gávea. Other sectors include ***Banking*** and ***Global Wealth Management Brokerage***, which assume different financial responsibilities than the other four. (Analyst Training Series, 2014)

As mentioned earlier, this project was part of the GIM Solutions sector within Asset Management. The GMAG team has a goal laid out every month, and we were able to provide significant assistance in achieving that goal. The GMAG team has been working on a specific task that is described in section 2.3. It also explains the reasoning behind the project, and the work done by the GMAG team in general.

2.3. Retirement Plans

Companies need retirement plans in order to pay off their liabilities. These liabilities include the contracts and promises the companies make with their employees. Even for small companies, this is a huge financial burden, and it needs proper management; for large companies one form of liability is divided into smaller parts for easier administration. Hence, companies oftentimes need assistance of financial institutions to take care of their payments to the employees for them, manage the money and make sure that the future payments are assured. Processes like these involve much estimation, and a lot of calculations, and the banks and other financial institutions have employees who work in this sector. JP Morgan Chase & Co. (JPMC) is one of the top investment banks in the country, and hence companies both big and small come to JP Morgan to take assistance related to their retirement plans with the employees. There are two major types of retirement plans that companies offer; Defined Benefit Plan and the Defined Contribution Plan. Both of these plans involve different risks and possibilities for the employer and the employee.

2.3.1. Defined Benefit Plan

The Defined Benefit Plan is a type of a pension plan where the employer formulates the employee's income and comes up with a specific amount that he promises to pay the employee after retirement. It is an employer-sponsored retirement plan where the employee benefits are sorted out based on a formula using factors such as the duration of employment and earnings history. In this retirement plan, the risk of investment and management of the employee's portfolio are entirely under the control of the company itself. Hence, this makes

it a high risk plan for the company (Investopedia, 2014). If the market does not fluctuate or drop significantly, the DBP ends up being very profitable for the company.

Since the payouts are dependent on the return of invested funds, the company will need to take part of its earnings, if there is any shortfall in funding the employee. One of the benefits of this plan is that this plan provides a predictable benefit. Thus, the benefits are not dependent on the asset returns. Since this is a costly plan, when adopting the Defined Benefit Plan, there are also restrictions on when and how you can withdraw these funds without penalties. It is an administratively complex plan, where generally the employer makes contributions.

This project looks into the portfolio management and construction for the companies that have adopted a Defined Benefit Plan. Additionally, it gains insight into how the logistics of this plan can be explained to the clients, the companies, and what can be done to assist them meet their liability needs with profits.

2.3.2. Defined Contribution Plan

The Defined Contribution Plan is a retirement plan in which a certain amount or percentage of money is set aside each year by a company for the benefit of the employee (Investopedia, 2014). Separate accounts are set up for the employees, or participants of the plan, and the benefits are based on the amounts credited to these accounts. In this plan, the future benefits are not guaranteed, as it depends on the investment earnings. Only the employer contributions are defined. When adopting this plan, the employee agrees to contribute a defined portion of his earnings to an account. Depending on the company, this amount is matched by the employer, either all or part of it.

The restrictions as to when and how you can withdraw these funds are much more severe, and there mostly penalties are placed if the person changes the plan or withdraws the amount before a certain period. Usually, the company retracts the amount matched if the plan is cancelled or the money is withdrawn.

Defined Contribution Plan includes two examples, given as follows. In both of these plans, to some degree the employee is responsible for choosing the types of investments towards which the money in the plan is allocated.

- **401(k) Plan**

In this plan, the retirement savings contributions are provided by the employer, and before taxation, these contributions are deducted from the employee's paycheck. Hence, this is a tax-qualified plan, and since 2014, the maximum annual pre-tax annual contribution is about $17500.

- o **Individual Retirement Accounts (IRAs)**

This is a retirement plan is a form of a trust that can be set up by an individual mainly for their own benefit, or for the benefit of their beneficiaries. This also includes a retirement payment, which is usually purchased on contract with a life insurance company.

3. Methodology and Outlook

The GMAG Team within GIM Solutions we worked with compromised of about seventeen employees. The team, led by Manish Ghayalod, works in group and individually to attain the results set out by everyone. Four of the people worked from the JPMC offices in India, while the rest worked in the New York City offices, both 4 New York Plaza on Wall Street and 270 Park Avenue building in Midtown. The entire team worked towards the same goal, of achieving excellence in helping clients meet their liability costs. In order to understand how the retirement system is related to the technical part that our team worked on, it is important to know the clients and teams involved, the role they play and how the different divisions within the GIM Solutions team function.

3.1. Steps involved in achieving the goal

Our project is geared towards helping companies with Defined Benefit plan pay off their liabilities with the least possible risk and maximum profits. What the team is creating makes it easier for JPMC, Client Portfolio Managers (CPM) in specific, to communicate with their clients. Hence, big and small companies that have come to JPMC for assistance in making their investments and building profits for them to meet the demands for their Defined Benefit retirement plan are part of the clientele for whom certain applications that GIM Solutions is working on are constructed for.

Companies with the Defined Benefit plan have an obligation to fulfill after making a commitment to their employees. JPMC provides assistance to help them fulfill this commitment. As mentioned in section 2.3.1, in a DB plan, there is a certain percentage that the company agrees to pay the employee after retirement, if he or she works with the company for a certain period of time. This is very risky and expensive for the company; hence they plan to invest some money years before they have to pay the retired employee. In order to figure out how every part of the team is linked together, the process is divided into four parts.

Part 1 – Estimation

First, to understand what role the company plays when involved with JPMC, a random company is selected. Let us say that company has certain employees who are retiring after a decade time. It sets a goal to have *x* amount of dollars by the end of the decade, in order to fulfill this commitment it made to the employees. Hence, the company wants to know if it would need to make an investment now, and if it should, how much money should it invest. If the money is invested now, how would that be different from investing money after a few years, and putting the current money for some other use? Additionally, it needs to decide the level of risk the company is willing to take. There are a lot of different things that needs to be

considered while making decisions about investment, like interest rates, discount curves, market growth and so on. JPMC is a financial giant, and hence this entire process is done with the help of the GIM solutions team at JPMC. Once the company's logistics are noted down, they are added to a database, so other teams may utilize the information.

General estimates are made by the JPMC and the company's representatives, to figure out which plan to select. Since the project is for the Defined Benefit plan, the possible outcomes are decided, and reasonable assumptions are made. In order to reach the *x* amount of dollars in a period of ten years, the JPMC takes the responsibility to make that happen by offering different solutions to the company, but the company itself makes the final decision on choosing the path to take. These solutions are provided after going through technical measurements, and are presented in a very viewable format.

Part 2 – Liability Driven Investments

This is where JPMC figures out what the liability (future obligation) is for the company, and to meet that liability, what the assets (required growth) should be, or how much the current asset needs to grow. This step is taken by the Liability Driven Investment (LDI) group, which involves Information Technology (IT) personnel and some Business Associates (BA) as well. The LDI group uses the LDI Web App to represent the information about the future liability costs of the company on a user interface as a part of portfolio construction.

Part 3 – Funds of Funds

Once the values are obtained by the LDI group, JPMC needs to figure out the best method to reach the required amount for the company, and hence there is a need to create a Funds of Funds (FOF) portfolio in order to have the asset reach the liability. This is also a significant part of portfolio construction, and certain user interfaces are used to represent this in detail as well. The user interfaces are also used to make certain assumptions, obtain various graphs and curves that show possible outcomes, future assumptions, and gives the company the best options that it may need to choose. Ideally, the goal is always to take low risk, and gain high profits, and hence JPMC provides feedback to the company about how to reach that goal. These are all the different scenarios that the team has to analyze before proceeding.

Part 4 – Global Database

An important part of the process is making sure the data is accessible and easy to use since the foundation of all the processes is the data used. Nothing can be achieved if the data is incorrect, missing, difficult to use or not understandable. All the data is available globally, and the portfolio managers need access to this data when making user interfaces. Hence, it is also the job of JPMC to make sure the data is well maintained, filtered and of excellent

quality. The Operational Re-engineering group does not involve many employees, since this is something that is not a continuous issue. Once the databases are edited to become more manageable, they are not altered until there is a new idea to make the usage more efficient.

Once these processes or the different aspects of the project are understood, it makes it easier to divide the tasks and take action to fulfill the major goal. Section 3.2 explains in detail what was done for each part, and how it helped to achieve the overall project goal.

3.2. Portfolio Construction and Management

3.2.1 Portfolio Construction

3.2.1.1 Liability Driven Investment (LDI)

Liability Driven Investment is a form of investing which the main goal is to gain sufficient assets to meet all liabilities, both in current and future. As the article mentioned before, Liability Driven Investment Project is always applied to defined-benefit plan. The "guaranteed" pensions that company will provide to its members are the liabilities that need to meet in the future. So to speak, Liability Driven Investment Project would be the first step in Portfolio Construction Project in order to help clients find out the liability to achieve their pension investment plan.

Generally, Liability calculation would involve lots of complex equations and computation, and large amounts of number and tables, thus in order to help Portfolio Manage team better construct LDI analysis and provide a clear view of investment to client, it is necessary to develop a visual functionality platform for Liability Driven Investment, which is the LDI Web App. The figure below shows a general outlook of LDI Web App.
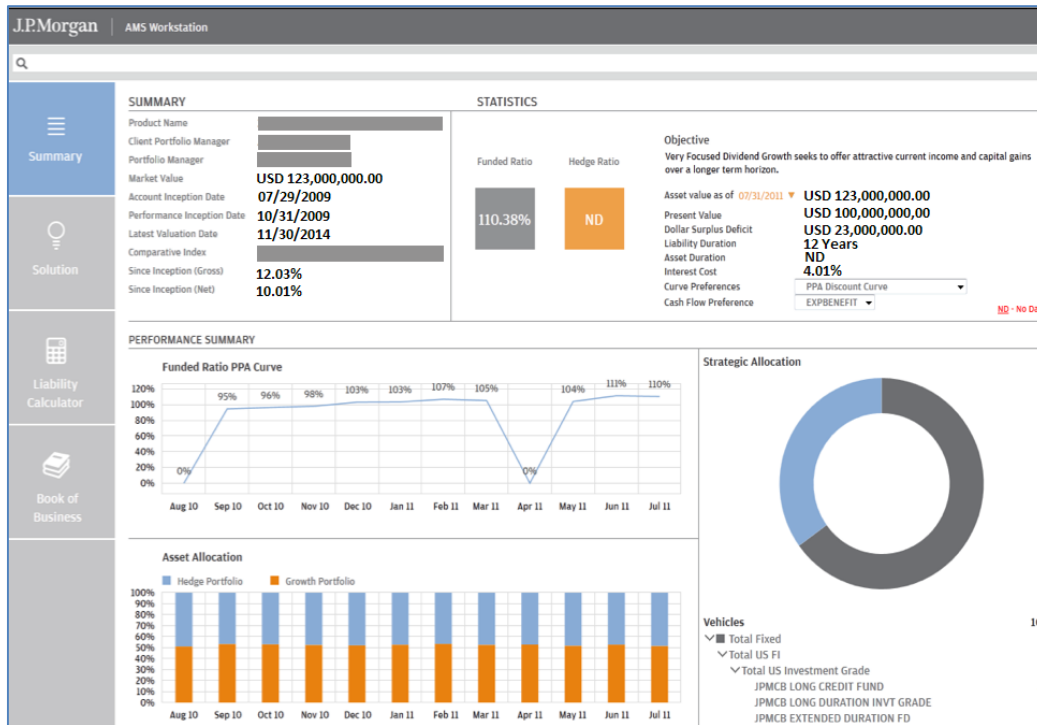
Figure 1: Liability Driven Investment Summary Web Page on AMS Workstation

JPMorgan's LDI team has done the framework and four functional blocks of LDI Web. It can be observed that the LDI Web contains summary, solution, LDI calculator and book of business already. In order to help Portfolio Manage team better analyze scenario condition and modify client information, there is the need to add another function block, Market data page. User will have options of three tabs - ash flow, service cost and discount curve, when entering market data page. Each tab has same functionalities, search bar, edit, delete and upload button. There will be an additional clone function of cash flow tab. The Figure bellow shows a brief outlook of Market Data page. Our team used HTML 5, backbone.js and Spring MVC to design this Web Page.

Backbone provides the structure of Web design with given key-value bind, customer events and lots of build-in API. It will connect your existing API to RESTful JSON interface. Spring MVC provides a Model-View-Controller structure to Web design. It connects front end, user interface and middle office together in a Web design project.
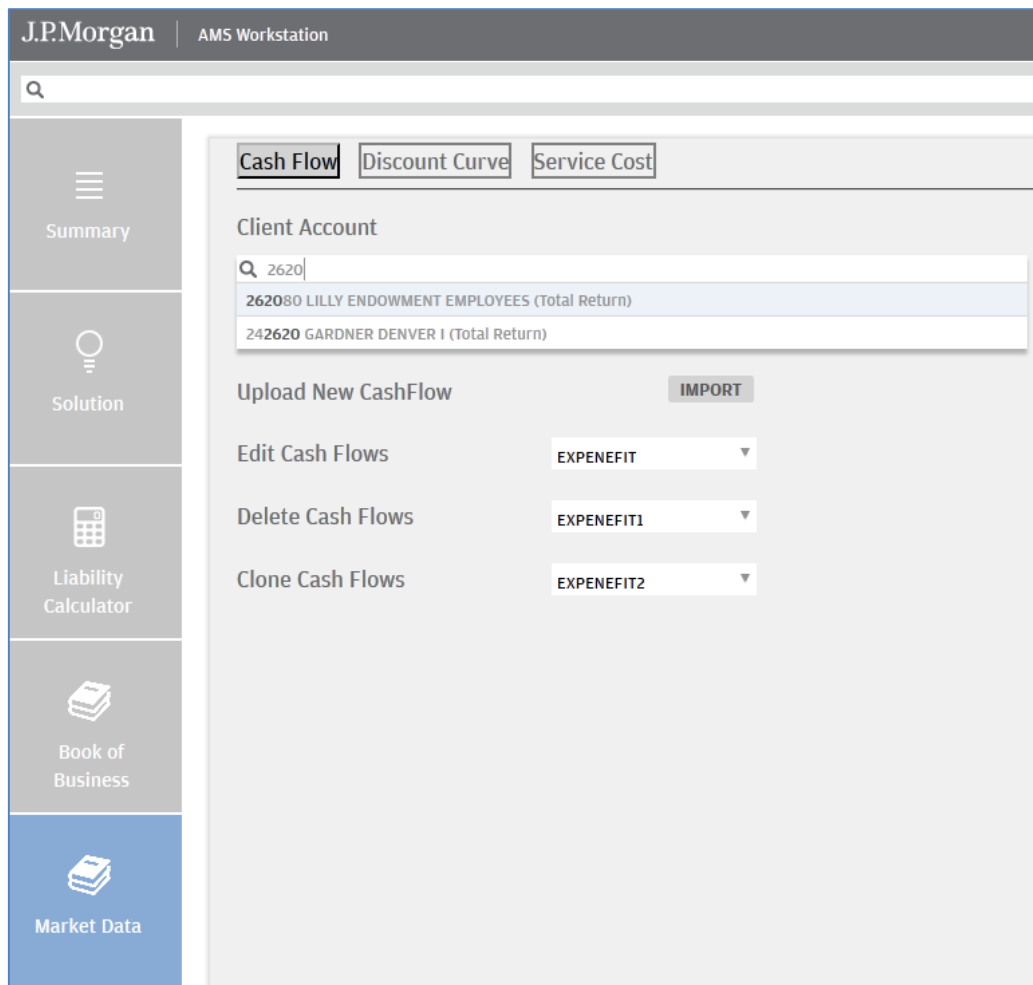
Figure 2: Market Data Page on AMS Workstation

The search bar is the core of *Market Data* block. It allows users to search for a specific client account by typing the account ID or the account name. It is accessible from JPMC's database so that any changes made to the file through market data block will be applied to all the other LDI screens.

Our team used backbone.js and spring MVC to write codes of the search bar. It is connected with JP Morgan's database and front end. When the user is trying to find a specific client account information, the search bar will connect to database and seek the information from database. JPMC team has developed template codes of search bar; our team called this template and modified most of the variables in order to make it compatible with the market data page. After the search bar has been called, our team added some more functions. Whenever users click one of three tabs, the corresponding page will open in a new window and the clicked tab will change to clicked type. In order to achieve this, our team used *event method* in backbone.js. We created events for cash flow, service cost and discount curve separately. The events are defined as clicked events which means whenever the tab is clicked,

the event will be trigged and corresponding function is called. Additionally, *upload*, *edit* and *delete* functions were also added by defining different field sets into each tab's HTML file. When the upload button is clicked, there will be a windows explorer diagnose opened and the users will be allowed to upload any file they want. When the *delete* and *edit* buttons are clicked, it will bring a drop down menu so that users can choose the file they want to delete or edit. The other function of *delete* and *edit* are still in progress and will be finished by JPMC LDI team in the future.

3.2.1.2. Funds of Funds (FOF)

Enhancing Portfolio Construction application

In order to construct a retirement plan, it is necessary to know the current asset in hand that is going to be invested. Funds of Funds helps customers to calculate and estimate how much the investment rate is, and also how much of a risk it has. In order to understand how to calculate this, it is necessary to know what the technical aspect behind this process is.

The FOF group utilizes a three tier architecture, which includes three independent layers: the front end, middle tier, and the database. For the *front end* we have the Windows Presentation Foundation (WPF), which is a graphical subsystem for rendering user interfaces in Windows-based applications by Microsoft. WPF was initially released as part of .NET Framework 3.0, but it is used as a user interface for front-end. The *middle tier* is the foam service, which includes a set of service deployed to a dedicated server in order to connect to the respect of queries to the database. The WCF is used for the *middle tier*. WCF is a framework for building service-oriented applications. By using WCF, data can be sent as asynchronous messages from one service endpoint to another. A service endpoint can be part of a continuously available service hosted by IIS, or it can be a service hosted in an application. An endpoint can be a client of a service that requests data from a service endpoint. The messages can be as simple as a single character or word sent as XML, or as complex as a stream of binary data.

As for the database, which is the *third tier*, the Oracle Database is used. Oracle Database is an object-relational database management system produced and marketed by Oracle Corporation. It stores all the information that is needed for portfolio construction.

The flow chart here clearly shows that data can go in and out through the front User Interface (UI) to database via the FOAM service. For example, one of the items that was planned to be delivered was inputting and viewing the 'fee' data. In order to deliver this item, it was first needed to be input and then viewed back. This information is provided in hand, hence it was easy to upload the data through the AMS workstation, which is the UI for front end, to the *middle tier* FOAM service via a connecter called the *client FOAM*. Once the data gets to FOAM it can be updated to the Oracle Database through the SMART Data Service, which is a data

service used to update data. If the data needs to be viewed back, the step by step procedure can be repeated, from the AMS workstation and to the data extractor in foam service, and the specific data extracted to view from the Oracle Database.
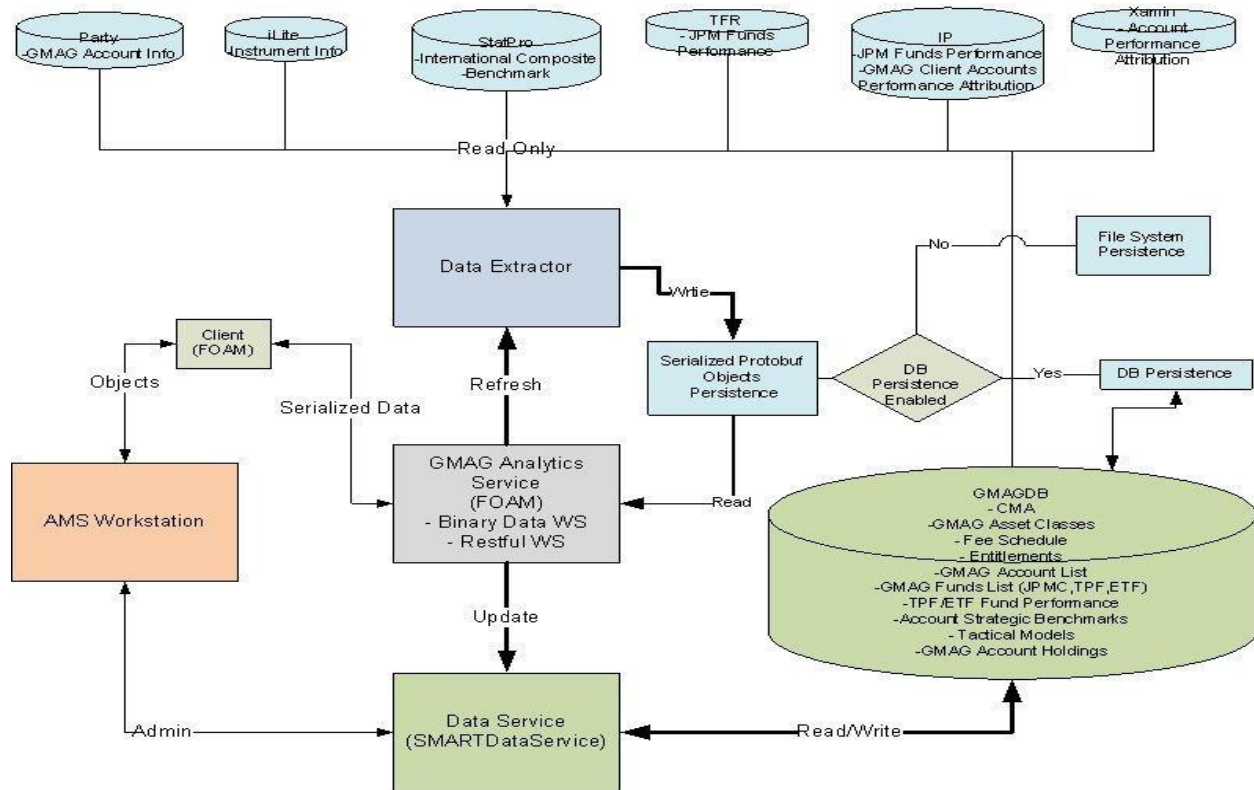

Figure 3: Different layers of the Architecture (A simplified figure can be found in the Appendix)

Why it is necessary to have data extractor to extract data rather than access the database directly? Data extraction is the act or process of retrieving data out of data sources for further data processing or data storage (data migration). The import into the intermediate extracting system is thus usually followed by data transformation and possibly the addition of metadata prior to export to another stage in the data workflow. Usually, the term data extraction is applied when (experimental) data is first imported into a computer from primary sources, like measuring or recording devices. Nowadays electronic devices will usually present an electrical connector through which 'raw data' can be streamed into a personal computer.

In other words, the data used in this case does not always store in the same place and also the demand is not always the same. If there is no data extraction from the FOAM service, the demand needs to be changed every time when the data from a different location is requested. It is very time consuming and takes a lot of work, that why it is necessary to have the data extraction step, instead of changing the whole demand only the location is changed each time.
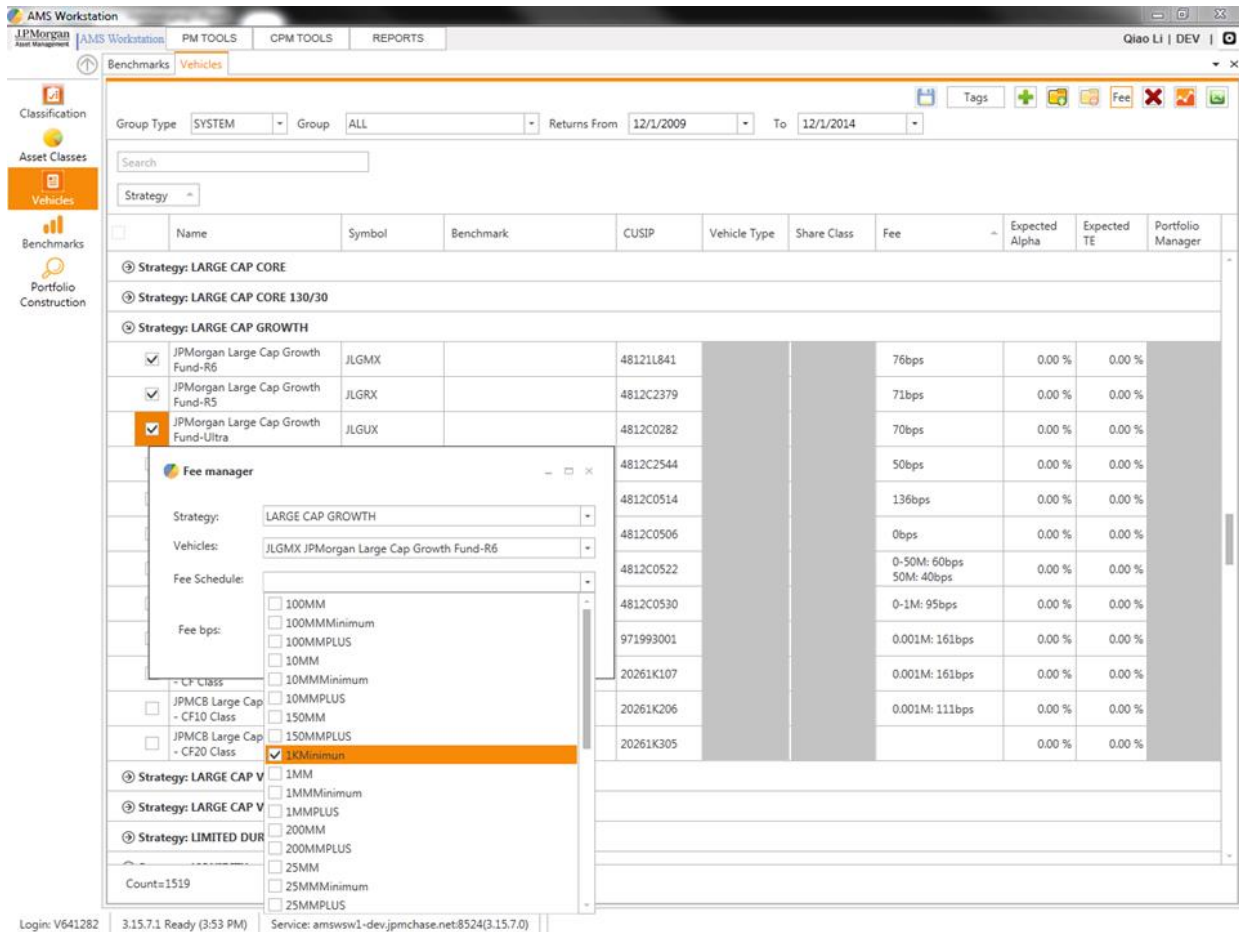
Figure 4: AMS Workstation FOF

As mentioned before our goal was to build a fee calculator for client to easily view fee information. In order to accomplish this feature, fee information needed to be added into the database first. So a fee column was inserted into the AMS workstation. Figure 4 shows a screenshot of the AMS workstation vehicle information. This page shows all the information about each vehicle such as vehicle name, benchmark, manager, and some other information. The fee column added shows the fee percent for each fund.

A fee button was also added at the corner of the page, which has the feature to update the fee information within the account. In order to use this feature, first specific vehicles that the user wanted to manage are selected, the button click produces a fee manager window that contains all the preselected strategies and vehicle names, and the data can be updated by selecting specific fee scheduler (fee range) and the fee percentages. This feature is very useful because before this there was no information related to fee present within the workstation and the database, but now it can be easily viewed and also updated.

AMS Workstation

JPMorgan Asset Management | AMS Workstation   CONSTRUCTION   PM TOOLS   CPM TOOLS   REPORTS   ADMIN   CSP   QIAO LI | DEV

Portfolio Construction

PORTFOLIO CONSTRUCTION USD - 2015 ☐ Historical ● Asset Allocation ● Product Selection ● Combined ○ Fees

Workspace	Types: None	Fund ABC		
Vehicle		Fund ABC	Full Fee	Stack Fee
Portfolio		100.00 %	2.79 %	2.43 %
Equity		80.00 %	1.68 %	1.09 %
US Large Cap		38.60 %	0.48 %	0.26 %
Fund 1A		9.00 %	0.25 %	0.12 %
Fund 1B		5.00 %	0.08 %	0.07 %
Fund 1C		8.00 %	0.16 %	0.07 %
Fund 1D		5.00 %		
US Mid Cap		0.00%(5.99)	0.06 %	0.04 %
Fund 2		4.25 %	0.06 %	0.04 %
US Small Cap		9.90%(3.91)	0.06 %	0.01 %
Fund 3		2.75 %	0.06 %	0.01 %
Multi Cap				
Fund 4A		7.25 %	0.27 %	-0.13 %
Fund 4B		7.25 %	0.22 %	0.24 %
US REITs		4.00 %	0.06 %	0.03 %

ANALYTICS

Summary	Fund ABC
Expected Total Portfolio Arithmetic Return	10.36 %
Expected Total Portfolio Compounded Return	9.38 %
Expected Total Portfolio Volatility	14.84 %
Expected Total Portfolio Sharpe Ratio	0.50
Expected Total Portfolio Duration	-

Login: E758881   3.16.1.1 Ready (1:53 PM)   Service: amswsw1-dev.jpmchase.net:8524(3.15.7.0)

Figure 5: AMS Workstation, Funds

Since the fee data is already inserted into the database, the next step is to build the fee calculator. Figure 5 shows the embryonic form of the fee calculator. The fee calculator goes within the portfolio construction feature and each time when portfolio managers construct a portfolio, they can view the fee information for that portfolio by selecting the fee button. This page does not only show how much fee percent charged per vehicle also shows how much percent charged in total per portfolio.

This way client portfolio manager can calculate how much current assets to invest by deducting the fee amount from the total amount that the clients provide. Also fee is just another way for the customers to view how much they are going to be charged.

3.2.2 Portfolio Management

3.2.2.1. Operational Re-engineering (Databases/ Middle Office)

One of the major ways of the Portfolio usage is through database. JPMC has various databases that store different types of information. The databases include the ORA_DEV and UAT databases, both of which perform similar functions but contain different data. These are databases are very large, and includes all the information about the companies working with

JPMC, and the clients. For our project, most of the portfolio construction and management utilizes the ORA_DEV database.

This database includes many tables with information about the Accounts that JPMC has, where these accounts include the information about the client, their cost, profits, origins and other general data that is needed to make a portfolio. This database is accessible through Oracle SQL Developer, since this is an Oracle stored database. Within the database, different procedures are performed in order to utilize the tables and do unique tasks. Since the data from the database is used in many of the JPMC operations, it is important that the data is stored in a manageable database, that it is clean, usable and of high quality.

One of the tasks provided to us by our sponsor was to perform operational re-engineering to improve the quality of the data by combining the tables that had similar data but were divided only due to certain specificities, in this case the country codes. This meant that there was a lot of data overlap happening, which meant that it took longer to process certain tasks when using the database and also was not a very organized way. The two types of tables that were merged were the EMEA tables and the US data table. EMEA tables were the ones that contained data for the European countries, while the US table contained the data from the United States. Since this division was done recently, and the EMEA tables were made so that they were merely an extension of the preexisting US tables, they contained much of the similar data present in the US tables. The following gives an example of what the EMEA and US tables were called:

- COUNTRY_SOURCE_MAP
- COUNTRY_SOURCE_MAP_EMEA

Where the one without the suffix _EMEA is the US table.

The goal was to merge the EMEA tables with the US by keeping all the US tables and removing the reference of _EMEA tables, while adding missing data from the EMEA into the US tables for easier maintenance in the future.

Since the data from the ORA_DEV database is continuously under use by portfolio managers, it was important to keep it functioning, and not make changes that might introduce bugs or any problems in the code. Hence, the original US and EMEA files were kept, and new tables were made in order to do testing before actually implementing them. These were made with the suffix _NEW or _1, depending on the number of characters in the name of the table. In order to do this, the procedures were altered to produce a _NEW table that met the criteria of a merged table.
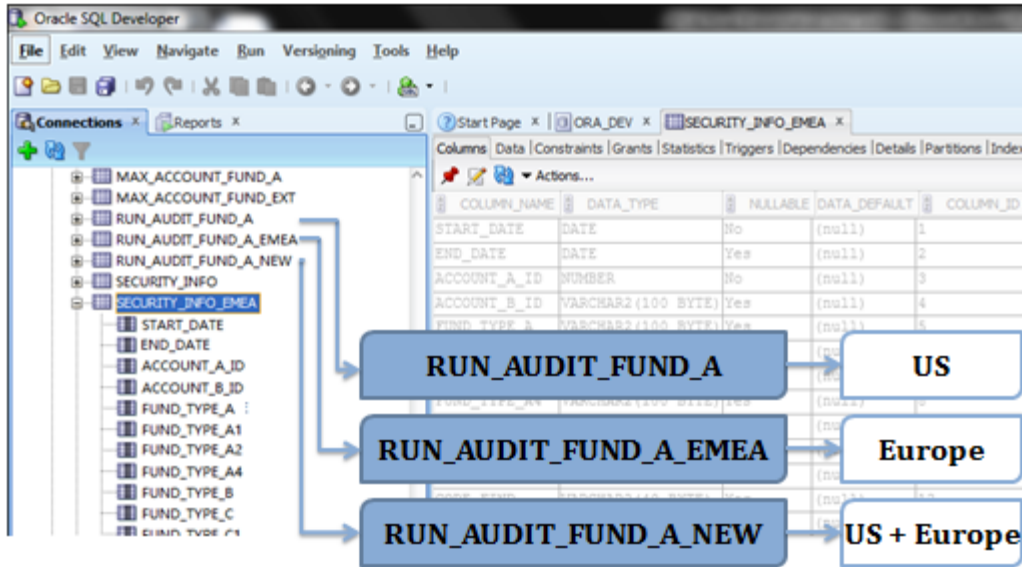
Figure 6: Oracle SQL Developer with US, EMEA and NEW tables

For example, a new package called RUN_AUDIT_FUND_A_NEW was created within the Oracle SQL Database ORA_DEV. The two original packages with similar tables already present include PKG_ RUN_AUDIT_FUND_A and PKG_ RUN_AUDIT_FUND_A _EMEA, both of which created tables for US and EMEA respectively. Step by step the code for the tables within the _NEW package that combines both EMEA and US data was written, without making major edits in the *procedures* within the package. While writing the code, it was made sure that the tables within the original files and the new files matched. The original files being the files that US and EMEA packages included, while the new files were the ones created in the NEW procedure.

In order to look exactly at how the tables needed to be merged, it was important to look at the content of each table, and figure out which fields define the country codes, that were needed in order to combine the tables. When merging the tables together using country codes, one issue that came was that there was an overlap of account information. For a column called SECURITY_ID, that contained the IDs for different accounts, there were different values for each account, but corresponding to that, the SECURITY_NAME had information that was similar in the EMEA and the US tables. The following tables show a part of each column, and how the information is same:

```
select security_id, security_name from security_info
```

	SECURITY_ID	SECURITY_NAME
1	89054	(FoF Converted to USD)
2	83071	
3	80269	
4	80257	
5	75415	**wssdmy Em Mkt Debt (eur)

```
SELECT SECURITY_ID, SECURITY_NAME FROM SECURITY_INFO_EMEA
```

	SECURITY_ID	SECURITY_NAME
1	387009	(FoF Converted to USD)
2	383799	
3	384056	
4	383474	
5	381859	**wssdmy Em Mkt Debt (eur)

Figure 7: The data overlap in EMEA and US tables

As seen in the above table, for different security_id, the information in security_name will be repeated when the data is merged. Hence, in order to avoid this, data claims were made so that there was no repetition. A sample code written for this is shown in appendix XX.

It is important to make sure that the data is easy to manage, since it is utilized in all the user interfaces used for the clients. The following figure shows how this data comes into play when figuring helping companies pay off their liability costs.

3.2.3 Regression Testing

As the Portfolio Manage Team (TM) relies so much on LDI Web App, it would be crucial to ascertain that this Web App is fully functional and working under stable conditions at all times. In order to achieve this goal, it is necessary to perform different types of software tests for the programs, all of which are involved in the LDI project. The one major issue that comes up when testing for LDI projects is that JPMC is a very big corporation, and many people work on similar things over a period of time. In the case of information technology, many different personnel including web developers and technicians do the programming, and their codes are uploaded to a main Oracle database. Both the current and historic files containing source codes, web pages and other documentation are kept uploaded to software called Apache Subversion, abbreviated to SVN. This is where the code files are kept, and it acts as a form of record keeping software. Since there are multiple users utilizing the same

SVN, there is a possibility for errors. As new ideas are brought together, the information technology administrators need to make changes to the web pages and upgrade them. In this case, there is a chance for programs to introduce new errors as developers make changes. Since it is extremely hard for developers to figure out where the faults caused by making new changes are, it is necessary to perform Regression Testing in order to find the bugs.

Regression means when a function or program regresses from stable condition to unstable condition after some changes have been made to project by developers. Regression Testing is a type of software testing that seeks to uncover new software bugs and can help developers determine where the problem is so that they can fix the errors quickly. The Figure below indicates the flow chart of processing Regression Testing.
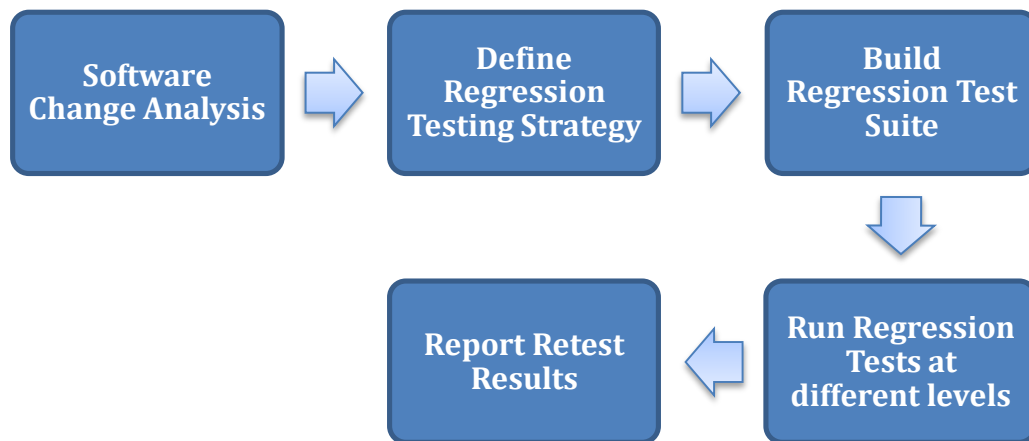
Figure 8: Steps involved in the Regression Test

Firstly, developers need to do an analysis of the software change and figure out the impact of the changes. With the change analysis, developers are able to define a strategy for regression testing. Then the regression test suite could be built properly based on the strategy and the developer should run tests at different levels in order to make sure every program in the project has been tested. Lastly, the regression testing will generate an error report to show where and what the error is.

Since there are hundreds of programs involved within the LDI project, it would be tough to perform regression test for every program within the limited period of eight weeks, but the framework of regression test has been done. The team wrote framework with Java and JUnit. As regression testing is applied to more programs, it is essential for developers to add or change functions within the regression test. This process makes sure that all the errors that are reported by regression testing are due to tested programs but not a compatible problem.

For instance, in the summary page of LDI Web App, there is a part of Statistics which shows the *Goal of Liability*. This part is able to generate the goal of the liability when users put in their *account number*, *asset date*, *curve preference* and *cash flow preference*. Since some computation programs will be involved in the process, it is important to ensure that these programs are not affected by the changes made in other programs. The regression test for this particular program can be divided into three steps.

Firstly, the test will store output data of different input combinations in separate files at a certain time, Time A. This file will not be changed in the future which means that the output data is fixed all the time. In order to make the regression test store the output and generate this file automatically, the team used the snapshot process in Java. This function allows the test to generate files at first and put the specific content into a target file.

Secondly, when the developers want to run the regression test after they have made some changes to the project, that is the current time or Time C, the regression test is able to read output data from the required URL. In order to achieve this goal, the team designed a *RegressionUtils* class in Java. This class includes a *FetchData* function and *Properties* function. *FetchData* uses *HttpGet* method, which is an API of Apache HeepClient. This method means retrieving of information is identified by the Request-URL. So, with this method, *FetchData* function can read the output data from the required URL when it is being called. Since JP Morgan LDI team does not want to use the localhost server when performing the regression test, our team created a webserver property for the regression test. With this property, the regression test will be able to connect to a separate server instead of a localhost server, so that there will be no data conflict generated during regression testing. The team used *InputStream* and *prop.load(in)* method to create a property and this *Property* function will be called at the very first whenever regression test is run.

Finally, the Regression Test will compare the output data stored in the file at Time A with output data read from the URL at Time B. In order to do this, the scanner class in Java was used to read the content in file and a *Gson* method to convert output data into objects which are much easier for developers to compare. Then, *AssertEqual* method in JUnit was utilized to compare output data. The *AssertEqual* methods will automatically compare target objects and show a red color in JUnit console window indicating a mismatch. If there is no mismatch among the objects being compared, the JUnit will show green color in the JUnit console window.

After these three steps are completed, an error report will be generated to show all the mismatched results in console window. Then the developers are able to figure out where the problem is based on the mismatched target and fix errors in the program.

```
[INFO ] c.j.a.l.r.RegressionTest 43 - Running Test Case for = accountno = 543210, asof = 07/31/2011, beiefiitpa.
[INFO ] c.j.a.l.r.RegressionTest 61 - Response from URL={"summary":"Inflation Protection","objective":"'ery·
[INFO ] c.j.a.l.r.RegressionTest 83 - {"summary":"Inflation Protection","objective":"Very Focused Dividind
[ERROR] c.j.a.l.r.RegressionTest 90 - FundedRatio Mismatch FileValue=105.%, UrlValue=105.82% !!!
[ERROR] c.j.a.l.r.RegressionTest 96 - LiabilityDuration Mismatch
[ERROR] c.j.a.l.r.RegressionTest 102 -
[ERROR] c.j.a.l.r.RegressionTest 108 - InterestCost Mismatch FileValue=5.%, UrlValue=5.19% !!!
[ERROR] c.j.a.l.r.RegressionTest 114 - HedgeRatio Mismatch FileValue=10%, UrlValue=ND !!!
[ERROR] c.j.a.l.r.RegressionTest 126 - AssetDuration Mismatch FileValue=1 year, UrlValue=ND !!!
[ERROR] c.j.a.l.r.RegressionTest 132 -
[INFO ] c.j.a.l.r.RegressionTest 135 - Completed !!!
```

Figure 9: Cut out Sample results obtained after running the Regression Test

The figure above indicates the outlook of error report. It is to see how the error report shows the developers that output data is mismatched and where the problem is. For instance, the fourth line indicates Funded Ratio mismatch and gives the exact value of output data stored in file at Time A and the output data read from URL when the regression test is running. The grey part is for block some client information because of JP Morgan's confidentiality.

This is all done in order to make sure that the Client Portfolio Managers who utilize the LDI Web Page do not find an issue within the system, since even one wrong value can have a large impact. Since they deal with the clients directly, it is important that all the information that reaches them is correct and utilizes high quality data. The regression tests help to remove any bugs that might be present before the release of the code.

4. Analysis

4.1. Globalization of the Oracle Database

Database is a major part of all the processes involved in the portfolio construction as well as portfolio management. Database, as mentioned earlier, is somewhere all the data is stored. This data includes information such as funds, fee, performance benchmark, strategies, account information, security information, and other data used for portfolio construction. It is important to provide a more organized database outlook, and also make it easily manageable and faster to process since it is used by various user interfaces and calculation purposes that the client portfolio managers utilize to communicate with the companies. In order to obtain exact values, correct percentages and for other investment calculations to be accurate, it is important that the data used be of a higher quality. This quality may be improved by various different methods, and JPMC continuously work towards having an excellent quality data.

Since JPMC is a multination corporation, the data stored contains information from companies all around the globe. It is important that this data be available from anywhere to the JPMC employees, but also be easily filterable. This was one of the tasks completed during

this project, the data tables were merged to provide single tables that were easy to manage and filtered and caused fewer overlaps of the data since similar information was removed.

4.3. Efficiency at workplace

4.3.1. Action Item Tracking

Every organization or corporation has a different way of running, a unique method to perform tasks and keep things in check. The managers have to make sure that the deadlines are being achieved and the goals are being met. It is difficult for a manager to run a team of many employees, hence the companies devise methods to help the managers stay on track.

JP Morgan Chase & Co. applies unique software that is used in tracking the tasks of each individual. This software is used by many different groups within GIM Solutions, including GMAG team and the LDI team. It is called a Jira Item tracking software. Each developer has his own page, as their JPMC account is linked to this software. When a project starts, one person usually takes the responsibility to assign specific tasks to each person. This person knows what each developer is doing, what the business associate wants from the developers and hence puts that information and makes a new task or a *Jira Item*. These items are very specific, and can be given a level of priority, hence the person themselves or the entire group can view the progress of others. These tasks are always part of a bigger project, and since many people are involved in the entire project, in our case twenty employees worked on the GMAG team, the software also provides statistics of the whole project.

Once the task for which the Jira item was created is completed, the person can open the page and mark completed, to show that the task has been done. This way the progress is recorded and up to date.

This is just one of the ways JPMC utilizes its resources to make sure that the employees stay on track and finish their tasks promptly. Whatever needs to be done, is completed in the required amount of time. It is a very useful way to checking who is doing what, and if something is not going right, it can be addresses easily.

4.3.2. Agile vs. Waterfall System

There are two different types of systems present when it comes to company's work efficiency; Agile and Waterfall. Much like construction and manufacturing workflows, waterfall methodology is a sequential design process. This means that in order for the developers to move on to the next step, all the previous steps need to be completed. Once a step has been completed, developers cannot go back to a previous step – not without scratching the whole project and starting from the beginning. There is no room for change or error, so a project outcome and an extensive plan must be set in the beginning and then

followed carefully. This type of methodology relies heavily on the initial requirements. However, if these requirements are wrong in any form, the project fails. Additionally, the temptation to delay thorough testing is often very high, as these delays allow short-term wins of staying on-schedule. The client's needs evolve with time, but this type of a plan does not take them into account. The project will have to be significantly delayed and will impact the budget if the client realizes that they need to make a change.

Agile, on the other hand, follows an incremental approach instead of a sequential process. The only reason it came about was to be a *solution* to the disadvantages of the waterfall methodology. The developers start off with a simplistic project design, and then begin to work on small modules within the project. This work on the modules is done in weekly or monthly sprints, and at the end of each sprint, project priorities are evaluated and everything is tested. These sprints allow for discovery of bugs that might be present, and also client feedback to be incorporated into the design before the next sprint begins. This process, with its lack of initial design and steps, is often criticized for its collaborative nature that focuses on principles rather than process. A good point about this system is that it allows developers to make changes and add features that will keep you up to date with the latest developments in the industry. The evaluations at the end of each sprint ensures that the bugs are caught and taken care of in the development cycle, and will not be present in the final release.

The Global Multi-Asset Group we worked with utilized the agile methodology, as is used by most of the JPMC. Everyday sprint calls were made, one for the LDI group and another one for the FOF and Restructuring. During the calls, each employee discussed what they did the previous day, and what they will be doing the present day. Additionally, they discussed any problems they were facing and asked for help. These calls kept everyone in the loop about what was happening and everyone was able to help someone who required assistance. The reason for using agile methodology is that it is a good system to use when rapid production is more important than the quality of the product, which is the case of JPMC. Also, since the clients are able to chance the scope of the project, it is important that their feedback is heard and changes made accordingly. When the project is started, a clear picture of what the final product should look like is not developed. It comes into form as the project continues and reaches the end. JPMC has skilled developers who are adaptable and able to think independently. Hence, this method does not cause any problems, but is actually preferred by the employees. Lastly, JPMC works in a highly competitive market, with a high rate of change of standards. Hence, agile methodology plays perfectly since it is applied when the product is specifically intended for an industry with rapidly changing standards.

5. Recommendations

We were part of the final *sprint* for the year, this means that the annual goals' deadline was at the end of December; hence a lot of data testing and web construction was going on during our time here. Each of us was able to achieve success in the individual goals we tried to achieve. Initially, there were some issued faced regarding the esoteric nature of the project. Only after understanding the business perspective of the project did everything fall into place and the three smaller project goals were related to one another.

After achieving the goals, we spent some time looking at the issues that we faced, and what else could JPMC do to be become highly efficient and also complete certain tasks that were not completed during our time there. Hence, a list of recommendation was developed.

- Apply the Regression Testing in other processes to provide a wider scope of locating errors
    - Introduce Regression Test as a part of the bill so it is run before every release.
- Run complete database analysis to make sure the data is clean
    - This can be achieved by placing someone in charge of the operational reengineering of databases.
- Provide a more visual representation of the data such as graphs and charts
    - Developers have to put a large number of columns side by side in order to observe all data
    - Using graphs within excel instead of tables
    - The LDI page is an example of where the information is more visual and graphical, but the LDI Web page is mainly for the client portfolio managers to communicate to the clients, something similar could be made for the sole purpose of usage by employees.
    - For example, at JMPC has significant number of clients, and in order to view their various fund amounts, the employees have to open tables and look through. Although, LDI Web service does provide assistance in this, but it shows one company at a time, and it is tough to see general trends happening in the market. Below is an example of how a graph would look like.

Figure 10: Sample Graph representing various companies and their balances

It is very easy to observe certain trends in the market across the various Accounts. JPMC can use graphs like this to present the information related to time, or other factors. It could also make a webpage where the employees will have the option to select any value for either axis, and compare the things they wish to see.

- Another obstacle we faced was that the team members were distributed among two different buildings. While most of the work can be done personally, and there are daily scrum calls, the division of the group still causes communication barriers and also utilizes more time to coordinate the job. One team could be placed within one area, hence if any problems arise, the team is readily available to assist.

References

"J.P. Morgan." *History.com*. A&E Television Networks, n.d.

"About Us." *JPMorgan Chase & Co. (NYSE: JPM)*. N.p., n.d.

"History of Our Firm." *JPMorgan History*. N.p., n.d. Web.

"Hedge Fund Definition | Investopedia." *Investopedia*. N.p., 20 Nov. 2003. Web.

"Asset Management." *Asset Management*. N.p., n.d

"The House of Morgan" by Ron Chernow

"26 U.S. Code § 408 - Individual Retirement Accounts." *26 U.S. Code § 408*. N.p., n.d.

"What Is Windows Communication Foundation." *What Is Windows Communication Foundation*.

"What Is Web Data Extraction." *Loginworks*. N.p., 22 Oct. 2014.

"Agile & Waterfall Methodologies - A Side-By-Side Comparison." *Base36*. N.p., 13 Dec. 2012.

Appendix A: Divisions within Asset Management

Organization Structure: Asset Management Technology
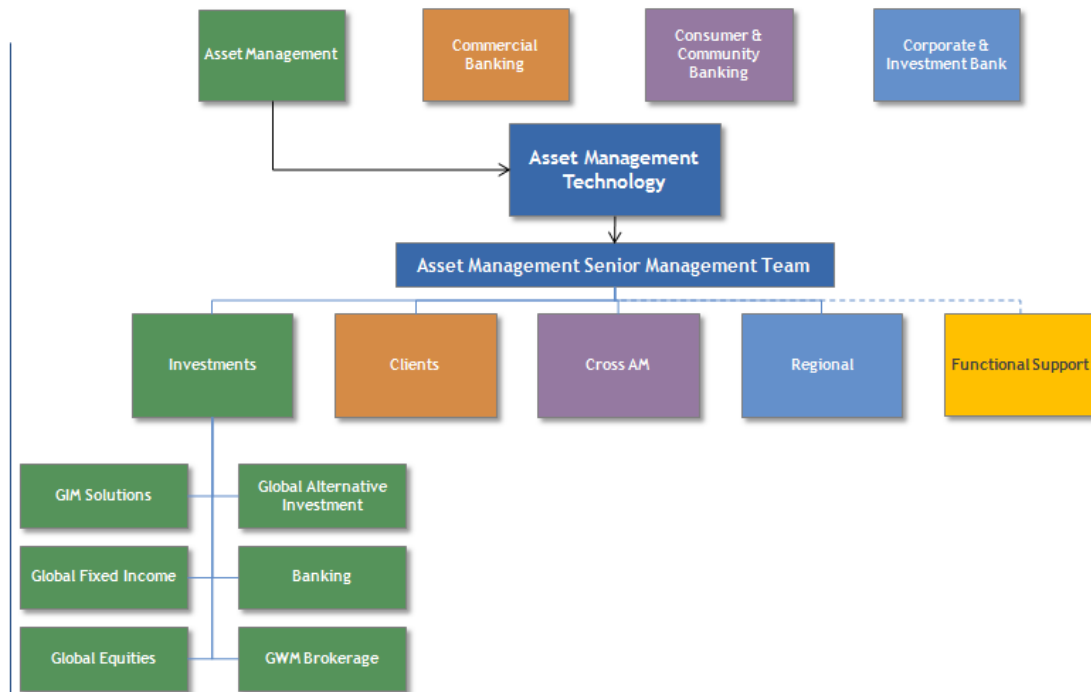
Appendix B: Divisions within GIM Solutions
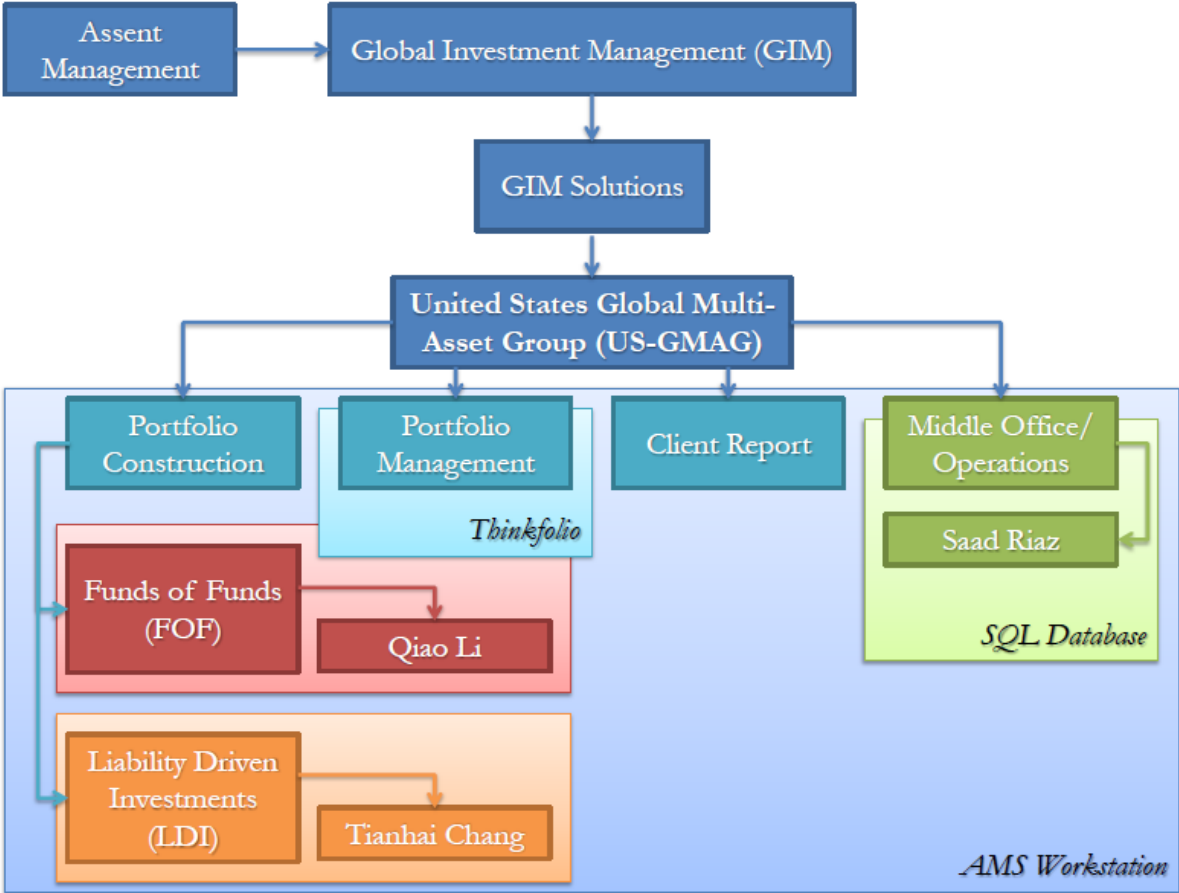
Appendix C: Correlation of the project to the different majors

The Liability Driven Investing – Investment Process is a project that encompasses a vast variety of different skills and goals. This project was a self-developed project in order to meet certain project guidelines. Once we arrived on site, we were placed in the Global Multi-Asset Group that incorporated mainly Information Technology employees, a few Managers and some Business Associates. A total of twenty employees worked in the team. This team worked towards assisting the Client Portfolio Managers to communicate easily with their clients, where the clients have obtained a Defined Benefit Plan and need JP Morgan Chase & Co.'s assistance in meeting their liability costs. This entire process is divided into many divisions, and our team worked within this investment process on smaller but very important and high priority aspects. Since it was the end of their yearly *sprint*, they had to complete the goals very soon, and hence each one of us was placed on a different goal.

Achieving the goals was very tedious, but sustained a big enough learning curve for us. We worked on different software and utilized various programming languages including JavaScript, C#, SQL, Visio and Tomcat. We did learn a lot, but a lot of the programming knowledge correlates to our previous work at Worcester Polytechnic Institute in the field of Electrical and Computer Engineering. The web design classes in college have had taught us how to achieve certain goals that were required by JPMC. Our knowledge on the training processes and engineering economics through Industrial Engineering major and minor also assisted us in determining the regression test flow, and the business aspect of the entire project. Dealing with the Client Portfolio Managers and figuring out the greater scope of the project would not have been possible without the previous knowledge on cash flows, business operations and finance. Lastly, Actuarial Mathematics assisted in determining the three layers of architecture, and how certain aspects of these layers can be woven in the work that we did. Developing the fee calculator was also a challenge, and significant data statistics were needed in order to implement is correctly.

Appendix D: Important Software, Languages and their purposes

Backbone.js: *This is a JavaScript library with a RESTful JSON interface and is based on the model-view-presenter (MVP). It is only dependency is JavaScript library. It gives structure to web applications by providing models with key-value binding and custom events, collections with a rich API of enumerable functions, views with declarative event handling, and connects it all to your existing API over JSON interface. The whole LDI Web App is built based on Backbone.js, thus it is the mandatory software to be familiar with in order to better improve LDI Web App project.*

Oracle SQL Developer: *This is an integrated development environment which helps in the development as well as management of the Oracle Database. The PL/SQL applications can be developed utilizing this software, and queries and scripts may be run to make reports and tables. Since JPMC utilized Oracle database, which is one of the biggest databases there is, it is important that the information within this database is easily manageable. Hence, SQL Developer was utilized to make any changes or edits of the database, and to perform Operational Re-engineering.*

Toad: *This is an Oracle application that JP Morgan uses in order to access database via PL/SQL applications. This application is designed for software developer use only, which needs complicated PL/SQL coding and testing.*

Tortoise SVN: *Programmers use this application. Programmers spend time to make little updates and they may change these update anytime. So they need tortoise in order to refresh or go back to specific version that programmers need to work on.*

Tomcat: *This is an open-source web server and servlet container developed by the Apache Software Foundation (ASF), and provides a HTTP web server environment for Java code to run in. Tomcat provides as a web server when building LDI Web App. Since LDI Web App is fully built based on Web service, Tomcat is essential for build LDI Web app no matter in terms of testing and Snapshotting.*

HTML 5: *It is the core technology markup language of the internet used for structuring and presenting content for World Wide Web. Since LDI Web App is the light web application, HTML 5 is one of the most important technology to be relied on. The structure of LDI Web App is built with HTML 5.*

Cascading Style Sheets (CSS): *This is a style sheet language used for describing the look and formatting of a document written in a markup language. Since LDI Web App is built with HTML 5 which is a markup language, the CSS is used for editing the style and format of LDI Webpage.*

AMS Workstation: *This is an application is a front user interface that JP Morgan uses to show backend information, which is stored in Oracle database. JPMC does most of its calculations, representations as well as communications with the clients using this AMS Workstation. By communicating with clients it means that information such as account ID, security ID and other forms of numerical values are obtained from here and showed to the client. Everything it utilizes, it uses from the Oracle database.*

Appendix E: Final Presentation

The data presented on this slide is not real data, and is only used as an example

The data presented on this slide is not real data, and is only used as an example

46

Future Recommendations

Apply the Regression Testing in other processes to provide a wider scope of locating errors

- Introduce Regression Test as a part of the bill

Run complete database analysis to make sure the data is clean

Provide a more visual representation of the data such as graphs and charts

- Developers have to put a large number of columns side by side in order to observe all data

J.P.Morgan
Asset Management