Summer 2015 NASA Internship Report

Inzamam Kaleem Rahaman

Contents

1 Intern Background

I recently graduated with first class honours from the University of the West Indies with a BSc in Computer Science. I was placed on the Dean's Honour Role several times while a student at UWI. In addition, I was awarded the IBM World Trade Prize for the best year I performance, the FUJITSU Trinidad Ltd. Prize for the best year III performance, and the Dr. Margaret Bernard-Medullan Prize for the highest GPA of a graduating Computer Science student in 2015.

During my time at UWI, I was an active member of the UWI Computing Society (UWICS), and was elected as the secretary of UWICS for the academic year 2014-2015. Under the guidance of Mr. Kyle DeFreitas, I conducted several workshops during this period on topics ranging from LaTeXto functional programming. Moreover, I worked for the Department of Computing and Information Technology as an undergraduate tutor at a weekly help desk alongside fellow students Stefan Hosein and Nicholas Mendez. During the July-August vacation of 2014, I also worked as a Research Assistant under the AgriNet project, applying data analytics and web development techniques to agricultural price data.

2 Summary of Internship Experience

Project: **NASA World Wind Project**

The Bay Area, comprising of several municipalities south of San Francisco, is the crucible where many of technologies that percolate our everyday lives were developed. The NASA Ames Research Center epitomises this sense of innovation. Consequently, as an aspiring scientist, the opportunity afforded to me by NIHERST to intern at the NASA ARC under the NASA I^2 programme was indeed a great one.

During my internship, I was mentored by Mr. Patrick Hogan and Mr. Randolph Kim of the NASA World Wind Project. My work during this internship was centered around the Web World Wind SDK, the latest member of the World Wind line of SDKs for virtual earth technologies. My typical work hours were 9:00 AM to 5:00 PM. Daily, my group would have a meeting at about 3:00 PM to discuss the progress made during the day as well as the goals for next day. From the 3rd week onwards, each of us were afforded a stint as team leader, allowing each of us to develop our leadership and project management skills.

Aside from the work I undertook, the ARC and the Bay area offered other avenues for intellectual enrichment. Every summer, the ARC is home to a series of lectures called the Director's Colloquim. Speakers across multiple disciplines and backgrounds, from animal behaviour to the history of science, are invited to give a short 1hr lecture on their work. Moreover, the Bay Area is home to the Computer History Museum. The CHM houses artifacts from across computing history, items from working replicas of Napier Bones to Seymour Cray's notebooks. For a CS major is it definitely a must visit.

3 Overview of Projects

During my internship, I worked under the NASA World Wind Project. Data needs to be stored, processed, and represented to the user in order for said user to draw meaningful insight that can engender actionable policies. Until data can be appropriately manipulated to be amenable to human cognition, data remains an untapped mine of knowledge.

Geospatial data refers to any data that can be associated to a point or region on the Earth. Subsumed into this class of data are datasets that represent the locations of cars, the epicentres and hypocentres of earthquakes, and the distribution of services in an urban space. Since geospatial data can be intuitively represented on a globe, various virtual globe technologies have been developed to help captalise on this rich class of data.

However, many pieces of virtual globe software are inflexible: facilitating a limited number of transformations and accommodating only specific sub-clasess of geospatial data. NASA's World Wind project seeks to redress this by providing an extensible, modular SDK that software developers can use to build more complex and varied solutions for geospatial data visualisation.

With the meteoric permeation of the Internet into our daily lives, many technologies once relegated to the desktop are being migrated to the web, thereby allowing for easy distributed access by end-users and rapid deployment. The Web World SDK, latest addition the NASA World Wind SDK family, is designed to help software developers create rich geospatial applications that can be served from the web to accommodate the technological paradigm shift arising from the Internet's increasing influence.

My work during my NASA I^2 internship was primarily undertaken under the umbrella projects: QuakeScape, an pedagogical web application that visualises earthquake data, and CitySmart, a web application for both urban policy makers and the general public that consolidates several components of the Open Street Map stack. In addition, modules from these projects that provided software services that could be potentially applicable to other projects were migrated into the core components of Web World Wind. Work was also done to subvert disruptive browser caching behaviour and to facilitate the manipulation of files from inside the web browser using blob storage.

3.1 QuakeScape

3.1.1 Background

No discussion of plate tectonics in a secondary school science class is complete without addressing one of the most harrowing consequences of our planet's tectonic structure - earthquakes. Typically, students learn best when exposed to concrete examples that reify the principles studied in the classroom. The copious amounts of earthquake data recorded by institutions such as the United States Geological Survey can serve as an excellent resource to develop educational tools to help students grasp geological concepts. QuakeScape was developed to supply a solution for educating students about earthquakes in accord with this realisation.

Moreover, while professional seismologists would have a better understanding of than secondary school students just learning about the existence of the crust and mantle, the visualisation of earthquake data can

still serve as a useful tool, both in the identification of discrepancies and in the task of educating the general populace on science.

3.1.2 Summary of tasks

- The USGS provides a publicly available REST API. A wrapper was written that acts as a proxy to this API. This proxy, exposed the USGS data as an array of **Earthquake** objects representing earthquake data. The cconstruction of **Earthquake** objects was off-loaded to a factory object in accord with the Factory Design Pattern.

- JavaScript's prototype-based object system facilitates a sort of pseudo-duck typing. Consequently, objects need only provide the subset of the contract of some type A to be used instead of being explicitly constructed as a member of A to be used in code that expects an object of type A.

  World Wind **Renderable**s need to satisfy two contracts:

  - The existence of a **render** method that accepts a **DrawContext**.
  - The existence of an accessor to a display switch.

  Consequently, by implementing a **render** and **display**, the **Earthquake** objects could be treated as **Renderable**s. To this end, different representation schemes were considered, and two prototypes were developed: a 3D Cylinder, and a 2D animated circle. These prototypes were then encapsulated in the **Earthquake** objects to use in rendering the earthquake data. Height and radius represented magnitude in the cylinders and circles respectively, and colour represented relative age.

- Given that earthquakes occur near tectonic plate boundaries, any application the seeks to represent earthquake data on a virtual globe must present tectonic plate boundaries to the user. Using appropriately licensed data, code to draw tectonic plate boundaries was written into a Layer.

- A touring system was developed that allowed for the software developer to easily define a tour and to allow for software developers to define tours of **Renderable** objects by simply defining binary comparison function that determines the relative ordering between pairs of objects.

- Various other filters were developed and implemented into the UI to enable to user to richly interact with the displayed earthquake data.

3.2 CitySmart

3.2.1 Background

With increasing interest in sustainable development, new technologies need to be developed to help in the appropriate management of resources. Hence, it is vital that software be developed to visualise resource use in urban spaces if we are to make progress in making our cities more sustainable. However, with urban management

data being of many types and coming from many sources, it can be difficult to get a clear, holistic picture a city that is needed to make informed policies.

The OpenStreetMap (OSM) stack is one of many crowd-sourcing initiatives undertaken with the intention of improving the state of open data. In CitySmart, various components of the OpenStreetMap stack were consolidated into a single application in order to leverage the data provided more readily for the analysis of urban spaces.

3.2.2 Summary of tasks

- Nodes in OSM represent some point of interest. The retrieval of nodes is accomplished by querying some OSM endpoint using either XML or the OSML DSL. Due to being easier to debug, a wrapper that generated OSML was built and used to communicate with the OSM Turbo endpoint. Using this wrapper, data on services provided within a geographical areas, called amenities, were queried and returned, packaged as **Amenity** objects created by a Factory object.

- Geocoding is the process of translating an address to a longigude-latitude coordinate. OSM provides a geocoding service called Nominatim. A wrapper was written to abstract away access to Nominatim. Using this wrapper, user supplied addresses were translated to coordinates that were then used to retrieve data from various endpoints.

- Using the OSRM (OpenStreetMapRouting) facilities in accord with the Nominatim geocoder, the ability to display routes between locations was added. Route objects were constructed to decouple route display from the underling representation of the routes.

- All of the above were integrated with a UI to facilitate easy user interaction based on a structured natural language query.

- A wrapper was built to retrieve data from the OpenStreetMapBuilding component of the OSM stack that allowed for building outlines to be drawn and colour-coded in accord with the building type (e.g. retail, fire station, restaurant, etc....).

  Given the massive influx of data, CitySmart initially scaled very poorly. To address this, the sampling bounding boxes used were shrunk and a cache based on an RTree was built to minimise the number of calls being made to variuos endpoints.

4 Lessons Learned

- A better understanding of how to construct modular and extensible software systems, in particular web applications written in JavaScript.

- An improved understanding of WebGL and related graphics concepts.

- Knowledge of the Web World Wind SDK, its components, and its architecture

- An understanding of geo-spatial concepts such as LiDAR, RTrees, and geocoding.
- An understanding of the OSM Stack.
- The ability to communicate well with domain experts in fields other than Computer Science and Software Engineering.
- A better understanding of project management.
- An appreciation of how Computer Science, Data Science, and Software Engineering can contribute to sustainable development.

5 The Way Forward

Given the current environmental and economic challenges facing the nations of the world in the current globalised context, there is an increasing need to ensure that our development is sustainable. However, sustainable development requires us to have a deep understanding of our current context and an understanding of where our various strategies might take us. Sustainable development requires us to be well-informed - to be data-driven.

In light of this, I hope that I can contribute to Trinidad and Tobago by improving the state of Data Science, through both educating others and developing useful technologies and techniques that can be leveraged to help us develop more data-driven policies.