Prajna – Migrant Boats Challenge

Edward Swing*

Vision Systems & Technology, Inc.

ABSTRACT

The Prajna Project is a Java toolkit designed to provide various capabilities for visualization, knowledge representation, geographic displays, semantic reasoning, and data fusion. Rather than attempt to recreate the significant capabilities provided in other tools, Prajna instead provides software bridges to incorporate other toolkits where appropriate.

This challenge required the development of a custom application for visual analysis. By applying the utilities within the Prajna project, I developed a robust and diverse set of capabilities to solve the analytical challenge.

KEYWORDS: Information Visualization, Software Toolkit, Knowledge Representation

INDEX TERMS: D.2.11 [Software Engineering]: Software Architectures - Domain-specific architectures; [Computer Graphics]: Methodology and Techniques - Interaction Techniques.

1 INTRODUCTION

This challenge involved a set of records of migrant boats, representing attempts to leave Isla del Sueño. Each record contained geographic and temporal information, including coordinates representing the location of either the landing or interdiction, and the date of landing.

The challenge included questions about the choice of landing sites, interdiction patterns, and successful landing rates. Secondary data included number of deaths, vessel type, and the passengers themselves.

VSTI develops software for diverse customers who face numerous analytical challenges. The solutions developed for this challenge should apply to other analytical challenges that our customers must face.

2 DEVELOPING THE SOLUTION

2.1 Analysis of the Problem

This challenge clearly required a geographic display for proper analysis. At the same time, the interest in landing rates and interdiction rates over time clearly required a statistical display with charting capabilties.

Furthermore, the data, in XML format, contained the passenger rosters in a single XML element. This would require some additional parsing specific to this data set. Finally, the geographic coordinates did not characterize the landing areas, so we had to create arbitrary areas to characterize the landing areas.

Email: deswing@vsticorp.com

2.2 Building with Prajna

The Prajna Project includes significant capabilities in a variety of technology areas. It also provides various software bridges to enable application developers to use specialized toolkits. Therefore, part of the design of the solution included selecting the appropriate elements of Prajna to include.

The first task for any analytical challenge is parsing the data. Prajna provides a data accessor for generic XML records. This data accessor interprets the data according to a data description file, which identifies fields and their data types within each XML record. This utility significantly simplified the data ingestion process.

In order to augment the data ingestion process, and derive additional information from the data provided, I created several Field Handlers. The first field handler divided the coordinates into the geographic areas we defined. The second field handler separated out the passenger list, and stored the passengers in a multi-valued field for each record. At the same time, this second handler kept track of the passengers, and quickly identified any passengers who occurred more than once.

The second task involved deciding how to represent the data. Since we needed a geographic capability, and the image of the island was already provided in compressed KML format, we chose to use Google Earth for the geographic display. Prajna provides a KML exporter, so it was a simple matter to create a KML file to represent the data. Once we discovered that GoogleEarth also provided a time slider, we augmented the KML exporter to include the temporal component of the records.

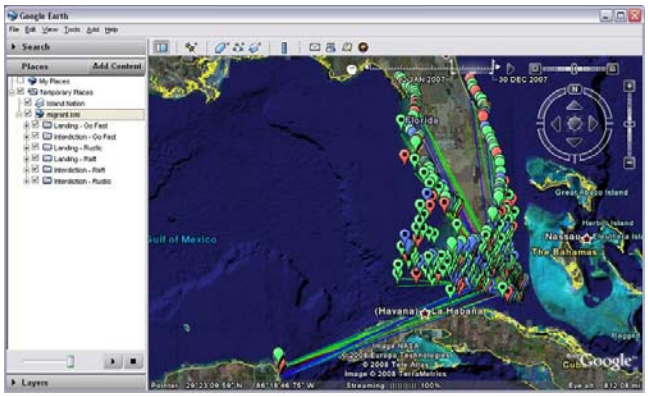
For the statistical analysis, we applied the Prajna bridge to JFreeChart. JFreeChart is an open source project that provides a Java toolkit for displaying graphical charts. The JFreeChart software provides a wide variety of charting styles and capabilities. During the development for this challenge, we augmented the Prajna bridge to include a flexible GUI component that allows dynamic selection of different chart displays, based on the fields contained within the data.

3 PERFORMING THE ANALYSIS

Once we completed the development of the application, we applied the capabilities of the tools to the challenge itself.

In order to characterize the choice of landing sites, we displayed the KML file we created in GoogleEarth. Using the time slider, we were easily able to trace the evolution of landing sites over time.

In addition to the qualitative information we could interpret from the GoogleEarth display, we used the dynamic charting capability to identify the statistical trends in the data. This process allowed us to validate the design of our GUI extension to the JFreeChart bridge.



Google Earth display: Distribution of landing attempts, 2007

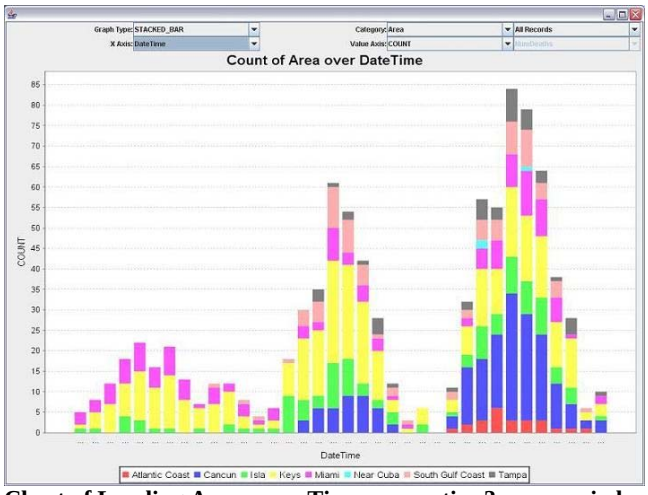


Chart of Landing Areas over Time over entire 3-year period

4 RESULTS

The analysis of landing areas showed that the Paraiso refugees diversified their choice of landing area over the three year time span. Initially confined to the Miami and Keys area, the migrants spread to wider areas in 2006 and 2007.

Initially, we believed that the Coast Guard was failing significantly in its efforts to intercept the vessels. The landing rates increased significantly over the three-year period. The Coast Guard certainly failed to adapt to the wider landing areas chosen by the migrants. However, we noticed that a significant number of migrants chose the Cancŭn area of Mexico in 2007. Once we filtered those records out, we realized that the interdiction rates near the US increased only slightly over the three-year period.

We decided to investigate the migrants who tried more than once to enter the US. A number of names appeared in our list of repeat migrants, appearing twice. However, two names appeared three times – Eduardo Catalano and Jesus Vidro. Furthermore, they attempted to migrate together all three times. We checked the landing records, and discovered they successfully landed near Cancŭn in early 2007 after two failed attempts to enter Florida. We assume their success may have spurred additional migrations into the Cancŭn area in 2007

5 CONCLUSION

The tools developed for this challenge provided us the answers we sought. Furthermore, we were able to apply the principles and design of Prajna to this challenge, demonstrating its utility. The Prajna project attempts to provide a robust toolset, leaving the development of sophisticated visualization tools for other toolkits. In this fashion, Prajna may adopt the best visualization techniques by providing a software bridge to innovative toolkits.

VSTI has already applied the techniques developed for this project to other development efforts within VSTI. Projects have already incorporated the enhanced KML overlays and dynamic charting components.

By providing an innovative architecture, which extends with software bridges to a variety of toolkits, Prajna avoids competing with the rapid pace of development across the spectrum of technology. Instead, Prajna offers developers the utilities to integrate new technology for knowledge representation in an intelligently-designed architecture.

REFERENCES

[1] E. Gamma, et al (the Gang-Of-Four), "Design Patterns: Elements of Reusable Object-Oriented Software", Addison-Wesley Professional, January 1995.

[2] D. Gotz, M. Zhou and V. Aggarwal, "Interactive Visual Synthesis of Analytic Knowledge", Proceedings of the IEEE Symposium on Visual Analytics Science and Technology, 2006.

[3] J. Heer and M. Agrawala, "Software Design Patterns for Information Visualization", IEEE Transactions on Visualization and Computer Graphics (Proceedings), 2006.

[4] JFreeChart Project (website): http://www.jfree.org/jfreechart/