Prajna: a Novel Architecture for Developing Applications for Knowledge Representation

Edward Swing*

Vision Systems & Technology, Inc.

Abstract

Applications and systems can represent knowledge in a variety of ways. A graphic display might allow a knowledge analyst to infer new information through interactive visualizations. Knowledge can be represented as a collection of facts, which can then be used for automatic inference. Knowledge can also be represented, or stored, in various archives, such as databases or formatted files. Developers who are challenged with creating applications for knowledge representation frequently have to contend not only with data challenges, but with challenges caused by a wide variety of software toolkits, architectures, and standards for knowledge representation

In order to meet these challenges, we have developed the Prajna Project. 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.

Keywords: Information Visualization, Semantics, Knowledge Representation, Software Toolkit

Index Terms: D.2.11 [Software Engineering]: Software Architectures - Domain-specific architectures; I.2.4 [Artificial Intelligence]: Knowledge Representation Formalisms and Methods; I.3.6 [Computer Graphics]: Methodology and Techniques - Interaction Techniques.

1 Introduction

Over the past decade, the Information Visualization community has developed a plethora of techniques, tools, and guidelines for presenting information. Through the course of software evolution, these tools gained more robust techniques and compelling interfaces.

Similarly, the Semantic Technology community has developed toolkits and standards for representing knowledge and inferring additional information implied by available ontologies. As part of this development effort, commercial companies and the open source community has developed a myriad of new standards, concepts and development efforts.

In addition, developers must apply specialized representations of data. Geographic or temporal data representations are quite common. The Geographic Information Systems community has developed a wide variety of tools and formats, as well as interaction techniques specific to the geographic domain. Similarly, reasoning and interacting with temporal data provides additional challenges, particularly when considering animations or streaming media.

To further complicate this challenge, data can be stored in a variety of ways, including unstructured text documents, XML-based formats, SQL databases, or even more specialized storehouses of data such as an RDF triple-store, a Faceted Search Engine, or a even a proprietary data store. In order to access this data, a developer needs to either develop a data-specific access for the application, or create a separate process to convert the data into a usable format.

The vast variety of tools, techniques, standards, data stores, and guidelines provide a monumental challenge for data analysts. On any particular problem, an analyst may need to interact with data from a geographic perspective, or display the data as a network. A corpus of data may have an implied tree structure, which is only evident when the analyst applies an inference engine and ontology. Frequently, knowledge analysts may be unaware of their needs until they have started working on a particular problem.

A developer tasked with supporting such an analyst may be asked to add new capabilities. For instance, an analyst may need semantic reasoning added to a geographic tracking system. However, in order to provide this capability, a developer would need to select an appropriate semantic toolkit, develop a mechanism to represent the geographic information in a format that the semantic toolkit can process, and integrate any user interface components required. All of these tasks are highly specific to the data, the geographic system, and the semantic toolkit. Furthermore, it requires the developer of a geographic system to become an expert in semantic technologies.

2 Prajna Architecture

The Prajna Project was created to provide capabilities from a variety of technologies, selectively blending toolkits and techniques where appropriate. In this fashion, Prajna allows a developer to select visualization techniques appropriate for a data set, rather than trying to coerce the data to conform to a particular visual tool. Similarly, a developer can include a semantic reasoning toolkit, or access data stored in a particular format.

For visualization, Prajna provides lightweight but highly extensible visualization capabilities that are able to display graphs, trees, and other structures of data. Prajna does not tie these visualization techniques to any particular representation of data. Instead, a layer of abstraction allows custom renderers to provide specific displays of data.

For semantic reasoning, the Prajna Project adopts a similar architecture. It provides lightweight capabilities for ontological inference and ontology creation. A developer can use this

Email: deswing@vsticorp.com

lightweight capability, or again use software bridges to access the data.

For data storage, Prajna offers data accessors which can retrieve data from a variety of formats and storage systems. Again, Prajna capabilities can be easily extended for custom data stores. Data format descriptor files enable data to be mapped from one data storage system to another. These files also enable Prajna to identify fields within a data store which can be used to construct more significant knowledge topologies, such as networks or trees of data.

While Prajna provides lightweight, extensible capabilities in each of these technology areas, it also provides software bridges to other software toolkits where appropriate. For visualizations, Prajna provides a software bridge to the Prefuse visualization toolkit, which can provide sophisticated visual displays. Prajna also provides a software bridge to the JFreeChart charting library for displays of statistical data.

Prajna also provides bridges to various data storage mechanisms. A JSQL bridge provides access to standard databases. Prajna also provides a data accessor for faceted navigation. Prajna also can access other formats for data, supporting common XML formats, ESRI shape files and KML for geographic data, and even unstructured text.

3 Prajna Applications

Prajna was used for VSTI's entry for the 2008 VAST Contest. Its flexibility is demonstrated by the diverse challenges which the toolkit was able to meet. In these challenges, Prajna was able to read data in a variety of formats, and provide a number of different visual representations. The types of analysis included social networks, geo-temporal data, tracking data, and semi-structured text analysis.

Developers at VSTI have also used Prajna to provide visualization utilities for faceted navigation. VSTI designed a user interface for a faceted navigation engine which allowed sophisticated queries and searches. However, using Prajna, the developers identified fields within the faceted navigation data which could be used to derive a social network. A user could then use the rich web-based interface to query and interactively display this information.

Prajna also has been used to provide semantic reasoning to a faceted search engine. To demonstrate this capability, VSTI used the Prajna semantic reasoner to apply rules from an OWL ontology against data retrieved from the faceted navigation system. The sample system used wine and food ontologies provided by W3C, and a faceted data store containing wine data. Whenever a user queried the system for a particular wine, the interface could provide a list of dishes which would compliment the wine.

Other tools have applied this same reasoning to the Wine and Food ontologies. However, none of these other tools attempts to apply the reasoning to data contained within an entirely different system. Prajna provides this capability by allowing a common data representation that applications may use for semantic reasoning or visualization.

Prajna capabilities also have been used for geographic event analysis, tracing events and matching them to known geographic areas of interest. Developers are continually investigating how to apply the Prajna framework to diverse knowledge representation applications.

4 Future Development

Prajna continues to expand its capabilities. Prajna already provides data accessors for various file formats, faceted search engines and standard SQL databases. Future capabilities include enhancing these capabilities, as well as adding accessors for other data storage techniques such as XQuery.

Prajna will also continue to enhance its visualization capabilities by providing software bridges to other robust visualization systems. While Prajna's own visualization utilities are quite flexible, and the bridge to Prefuse enables additional sophisticated visualizations, other visualization tools are evolving. Future techniques will offer new visual paradigms. By providing software bridges to these new applications, a visualization developer can incorporate the power of other technologies such as semantic technologies or data fusion.

Similarly, Prajna will provide software bridges into semantic toolsets such as Jena, and explore the capabilities of entity extraction. While Prajna provides some capabilities for semantic reasoning, incorporating Jena and other semantic utilities will considerably enhance the semantic capabilities.

5 Conclusion

Researchers in academia and industry will continue to create new tools and techniques in their fields. The Open Source community will continue its creative explosion in a rich set of technologies. Developers who need to apply techniques from different fields should have tools that they can adopt without having to become an expert in an entirely new technology. 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] C. Fluit, et al, "Ontology-based Information Visualization: Towards Semantic Web Applications", In .eds V. Geroimenko, C. Chen, Visualizing the Semantic Web – second edition, 2005.

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

[3] 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.

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

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

[6] The Prefuse visualization toolkit (website): http://www.prefuse.org/