To: Computer Science Graduate Students
From: Prof. Jeff Wiegley
Subject: Open Graduate Projects
Date: 10/20/2006

---

1 Selection of a Thesis Topic

The production of a thesis is a requirement for the degree of Masters of Science in computer science at CSU Northridge. Student's frequently find it difficult to select a suitable topic.

Here are some tips:

- Pick a narrow topic. (Yes, **narrow**.) Wide topics may appear to be more desirable because of the misconception that they generate more information to fill 60–100 pages of material. But wide topics are impossible to tackle properly and are very difficult to narrow down later. Narrow topics provide a wealth of details readily and are easily expanded if the problem turns out to be too easy.
- Do something unique. JSP/ASL/.NET/SOAP shopping cart projects have been done to death. Prof. Wiegley will not chair or committee yet another such project.
- Consider typesetting your document using LaTeX 2_ε. It will produce a much more professional document than Word will ever hope to produce and Prof. Wiegley has provided a `CSUNthesis.cls` class file for use with LaTeX 2_ε that will automatically make your thesis properly conform to the CSU Northridge graduate evaluator's thesis guidelines. If you select either Prof. Wiegley or Prof. Noga as your thesis chair then using LaTeX 2_ε is a requirement.

2 Availability of Topic Ideas

If you don't have your own topic idea and you are open to strange thesis topic ideas then Prof. Wiegley might have an answer for you.

Prof. Wiegley is an inquisitive sort of soul and gravitates to a wide array of topics and interests. This results in lots of project ideas but he does not have the time to tackle all of them himself. So they are well thought out, narrow ideas that make excellent thesis topics and you can adopt one of these ideas as your own.

Assuming that Prof. Wiegley is not swamped with graduate students all ready he will agree to chair any of the thesis topics described below.

Here are the advantages of picking one of these projects:

- The problem is usually relevant and useful to industry.
- The problem is sufficiently narrow yet rich enough to easily generate enough detailed information to fill a 100 page thesis.
- The problem is considerably unique and has not been tackled by a CSU Northridge student before. Possibly not by anyone before.
- Prof. Wiegley knows enough about the topic to keep you on track and manage the scope of the project should areas of it prove too difficult.

- A significantly successful solution to most of these projects can result in a published research paper as well as a successful defense.

Here are the disadvantages:

- Prof. Wiegley is a facist when it comes to typesetting. You must use LaTeX 2_ε to publish your document and you will be expected to do it right.
- The projects are significant. You cannot expect to do two weeks of implementation followed by a month of writing and defend. You can expect that the implmenentation portion of each of these projects alone will take 6 months of dedicated effort.
- The projects are challenging. If they weren't, Prof. Wiegley would crank out a solution in a month of his own time. These projects are put up for adoption because Prof. Wiegley has realized they need more than one man-semester to tackle.
- The projects generally utilize and require technical topics that are not taught in any of the CS classes. You are a graduate student and the thesis is designed to show that you can make your academic contribution, not just regurgitate what you have learned.
- They tend to be projects that could be turned into Ph.D. dissertations rather than thesis. I.e. Prof. Wiegley has seen doctoral theses which are weaker than some of these topics successfully defended.

3 Topic List

3.1 Marine Navigation and Performance

Strangely enough, Prof. Wiegley has a dream of single handedly sailing across the Pacific Ocean. He plans to have a "smart boat" and acquire lots of useless data during his voyage.

Here is a list of topics that have been identified as related to this project and suitable for theses topics:

3.1.1 Ethernet Network and Protocol for Maritime Communications

The National Marine Electronics Association (NMEA) is a standards organization that publishes expensive, closed, proprietary communication standards for communication between electronic maritime equipment. NMEA-0183 is an RS-232 based communication standard. It is commonly available on almost all maritime electronic devices. It has many limitations, mildy expensive, slow and a generally antiquated technology. NMEA-2000 is a CAN based standard designed to replace NMEA-0183. It is a horrible design. It is ten times as expensive, closed and proprietary and solves almost none of the problems associated with NMEA-0183.

Prof. Wiegley proposes leveraging the advantages of Ethernet networks and TCP/IP protocols to provide a truly open standard capable of higher reliability, lower deployment cost, higer bandwidth, all the things NMEA-2000 should have been, only truly free.

The project will include the development of TCP communication protocols and UDP datagram packet protocols for effectively deliving the range of communication needs required by electronic maritime equipment such as autopilot, GPS, Speed, depth and temperature sensors as well as

the possibility of high bandwidth transducers such as sonar and radar.

3.1.2 Electronic Navigation Chart (ENC) library and parser

The Internation Hydrographic Organization (IHO) has produced a standard for encoding hydrographic survey information for use in electronic maritime navigational and charting applications. This standard is known as S-57 and is used to provide modern Electronic Navigational Charts (ENC).

The National Oceanic and Atmospheric Administration (NOAA) Maintains and releases ENC charts of United States coastal regions. The United States Army Corps of Engineers also provides Inland Electronic Navigation Charts (IENC) of most navigatable inland waterways in the United States. Other international governments produce ENC charts though often only for purchase.

This project will provide a C++ object oriented library and parser for reading, storaing and manipulating such ENC encoded files.

3.1.3 Vector Based Digital Library of S-52 Navigational Symbols

The Internation Hydrographic Organization (IHO) has also produced a standard for how navigational information is to be presented visually. This standard includes the visual description and coloring of how all navigational items will be displayed.

This project will provide a vector based digital library of such information for use in graphical navigiational applications in conjunction with a C++ ENC library.

3.1.4 Maritime Sensor Data Acuisition, Storage and Retrieval System for Linux.

A modern boat has a weath of data sensors and relies on the use of this data for crew safety, performance and successful navigation.

Sensors include, Radar, Sonar, depth, GPS, velocity through water, velocity through air, water and air temperature, humidty, power monitoring, compass heading, roll, pitch, yaw, acceleration data and autopilot control.

This project will develop a system for capturing NMEA-0183 data and efficiently and conveniently storing such data into a suitable database for realtime use.

3.1.5 GUI integration and display of Maritime Navigation and Sensor Information

This project will design and build a Qt/C++ based graphical user interface to provide an interface terminal to a boat's crew to present aquired sensor information in useful and intelligent manners.

The GUI will utilize a C++ library to display ENC charts with information overlays to augment navigiational capabilities. Additional features will display and monitor various sensors in an intuitive and flexible manner.

3.1.6 Wireless Mobile Interface for Maritime Control

Modern boats still rely on a fixed array of guages to provide information to a ships crew. This is an inflexible and antiquated model given the high availability of 802.11a/b/g and bluetooth devices capable of This project will provide a mobile platform for monitoring and control of a smart ship.

This project will explore programming interfaces for handheld devices to provide convenient and portable information and control systems aboard modern sailing vessles.

3.2 Robotics

3.2.1 Kinematics Library

This project will design and implement a robust C++ Object library for use in robotic control programs. Library will include kinematic transforms for positioning and trajectory, PID controller functions for control and Jacobian matrix functions for velocity, torque and force control.

3.2.2 Industrial Manipulator Simulator

Develop a graphical environment for the specification and simulation of robotic manipulators described using standard robotic techniques. Simulator should detect obstacle collision and trajectory simulation. Simulator will be extended to parse and interpret various robotic control languages as used in industrial equipment.

3.2.3 GCode Servlets

GCode is an antiquated programming language for decribing Computer Numerical Control (CNC) actions. It is still used in modern milling and lathe equipment. It's programming abilities compared to modern languages such as Java, C++ or C# is terrible.

This project research the possibility of using a JSP like servlet interpretation to provide modern language techniques and capabilities while maintaining backwards compatibility with existing industry machines.

3.2.4 DC servo control using Rabbit Semiconductor microcontroller

Rabbit Semiconductor manufactures microcontrollers that have on-board quadrature decoding circuits and PWM digital outputs. This project will implement a PID controller algorithm to produce servo control of a DC brushed motor. Extension to DC brushless control may be considered as well or as a second thesis on the subject.

3.2.5 A Modular Scalable Mail System

Current Mail Transport Agents (MTA) include Sendmail, Qmail and Postfix; all of which were designed prior to the problem of spam mail. As a result none are particularly well suited to detecting or handling such mail, nor are they easily integrated with spam and virus detection applications such as amavis or spamassassin.

This project will consider a redesign and reimplmentation of a RFC-2822 compliant Mail Transport Agent that includes a highly modular design that allows for intelligent and scalable mail processing. The goal will be ease of configuration, security, modular and extensible capabilities and ease of integration with other mail-aware services.

3.3 Software Engineering

3.3.1 Abstract engineering of Subversion Backend

Subversion (SVN) is a revision control system. It is intended to be a more modern replacement for CVS. It is open source and quite robust. One draw back to it's design has been the reliance on Berkeley Database (BDB) to provide the file, information and data storage for repositories. The original design of SVN did not take into account the presence and flexibility of other storage databases such as Oracle, flat files or MySQL.

This project will examine the current architecture of SVN and redesign the functions, methods and API to abstract the backend interface so that repository administrators can choose the most suitable storage backend for their purpose. Such a system could allow SVN repository functionality to be more easily integrated with a wide variety of products. For instance Moodle or Sakai could benefit from having student submissions hosted in a revision controlled manner.

3.3.2 Extending SVN to Support Sets within Repositories

Subversion (SVN) is a revision control system. It is intended to be a more modern replacement for CVS. It is open source and quite robust.

It is sometimes desirable to limit the files checked out in a working copy to a particular subset of all the files present in the repository. For instance to enforce least privileges.

It may also be desirable to maintain a single directory tree of related items but have that tree's directories and files be populated from multiple repositories. For instance `/WorkingCopies/Courses/MATH101/` could be revision controlled in the "CoursesPublic" repository while `/WorkingCopies/Courses/MATH101/Ros` could be revision controlled in the "CoursesPrivate" repository.

It makes sense so that one could for instance pull a working copy of the "CoursesPublic" repository to present as the instructor's web pages but you would not want personal information as stored in a roster to be posted to a web page.

Conversely, it is possible to maintain separate `/WorkingCopies/Public/...` and `/WOrkingCopies/Private/` repositories but then related course materials are not maintained within the same directory hiearchy.

Modifying SVN to provide the ability to overlay repositories or to limit checkouts to a particular subset Solves the problem.

3.3.3 Qt/swing xfig replacement.

XFig is the standard application in Unix for making figures (like Corel Draw). It is simple and supports LaTeX 2_ε. It was built on the original X-windows widget set. Millions of people use it regularly. Unfortunately, the GUI world has come a long way since xfig was designed and xfig's user interface is a disaster compared to modern applications.

The world needs an intuitive, easy to use replacement for xfig that is platform independent. Qt (from TrollTech) or Java swing (Sun Microsystem) are suitable widget libraries for basing the next generation of figure drawing program on.[1]

xfig is strictly 2D but there is no reason that a well thought out replacement could not be extended to also provide 3D facilities. Such features would be a great benefit to the research communities where there is a strong need to depict three dimensional constructions and concepts in textbooks or research papers.

[1] Prof. Wiegley leans towards Qt as it is compatible with C++.