CoccoExpress: Database of Expressed Sequence Tags of Coccolithophorids

M. Ranji, and A. R. Hadaegh
Department of Computer Science at California State University San Marcos
{ranji001, ahadaegh}@csusm.edu

Abstract: **Coccolithophorids contribute about 15 percent of the average oceanic phytoplankton biomass to the oceans. They produce elaborate, minute calcite platelets (Coccoliths), covering the cell to form a coccosphere and supplying up to 60 percent of the bulk pelagic calcite deposited on the sea floors. In-depth profile and detailed understanding of Coccolithophorids genome will significantly contribute to science of regulating the atmosphere. Increasing amount of research is being conducted on effect of Coccolithophorids in fight against global warming and production of greenhouse CO2. With growing need for a genomic database of all genome sequences for Coccolithophorids, CoccoExpress was planed and built to provide a solid database and search engine of Expressed Sequence Tags (EST) of Coccolithophorids Marine Alga.**

This paper describes the role of the several web components used in CoccoExpress that facilitate navigation, search, security, and maintenance of this database. Components are selected based on a detailed look into research involving the data deposited into CoccoExpress for gene analysis of these species.

Index Terms--**Coccolithophorids, Dynamic Programming, Emiliania Huxleyi, Expressed Sequence Tags, Marine alga**

I. Introduction

Bioinformatics, the science of using computational techniques to solve biological problems, has been an influential research for the past decade. Using the fast growing advancements in computers and mathematics, new discoveries are made every year that can significantly contribute to life matters. To this day, several hundreds of organisms have been examined and their DNA sequences were decoded. These DNA sequences are stored in databases and are being used to solve many of the biological mysteries such as determining genes that code for description of proteins, discovery of diseases, or gene predictions and their functions. There are several tools developed every year to contribute to these efforts and make break through where not possible before due to limitation of technology.

Developing CoccoExpress is a project dedicated to serve as a database to perform efficient search on genetic characteristics of Coccolithophorids: a family of species known for their unique ability to produce calcium carbonate platelets [6], [8]. Through extensive research, increasing number of sequences of this family is extracted. This project was initiated to maintain this data and present a meaningful representation of it by building a solid relational framework to study the relation and interaction of data. This paper centers around presenting our effort made to create CoccoExpress.

The main contribution of CoccooExpress in this bioinformatics research of analyses of Expressed Sequence tags of Coccolithophorids include:

- a dynamic, extendible web site that provides an intuitive customized simple and advanced search engine for the researchers
- integration of the database with a set of built-in backend application tools that helps the users and administrators of the database in many ways. This includes information retrieval, security of the data, system traffic and web statistics, and database administration

The rest of this paper is organized as follows. Section 2 provides a general overview of related work to this project. We discuss the primarily work on Coccolithophorids that led us in creating the CoccoExpress. In addition, we show how CoccoExpress relaxes the constraints involved in the search limitation of other exiting EST databases. Section 3 details the implementation of the web site. It describes the dynamic nature of CoccoExpress and illustrates the flexibility of the search engine that makes this work a novel. Integrated tools that have simplified the usability of CoccoExpress are presented in section 4. Finally, we conclude this paper and point out the open problems.

II. Relates Work

Currently, CoccoExpress contains over 120,000 ESTs from *Emiliania huxleyi.* Emiliania Huxleyi (Ehux) is the most commonly known species of Coccolithophorids and was selected for whole genome sequencing by the Joint Genome Institute (JGI). The sequences are obtained through a number of experiments and collected by the researchers at California State University San Marcos. As explained in [13], [14], each EST sequence is a single pass read from a randomly selected cDNA clones. ESTs can be assembled into either overlapping (contigs) or non-overlapping (singletons or singlets). Cluster analysis reveals the most likely arrangement of these fragments. The products of cluster analysis are known as consensus sequences, representing overlapping stretches of cDNA in which each string position is filled by the most likely

nucleic acid for that position. By comparing each fragment to large, publicly available databases of known genes, the identification of the consensus sequences within the genome can be partially established. The consensus sequences are blasted against the NCBI non-redundant sequence database, and the top matches are stored for later analysis [1], [11], [13], [14]. This is still an ongoing research at California State University San Marcos. The primary goal of this exercise is to reduce the EST dataset into a biological meaningful set of sequences which can be readily maintained, manipulated and queried in a database.

Thus, "Cocco Express" was designed to assist in the gene annotation of Coccolithophorids, and is the first known database built to store and organize data for Coccolithophorids. Using the search engines in CoccoExpress, researchers can profile expression patterns under specific conditions to determine the portion of the Coccolithophorids genome that is transcriptionally active. CoccoExpress serves both as a repository for ongoing sequencing efforts and facilitates the public dissemination of sequence information of Coccolithophorids.

Some other similar EST tools and databases developed for retrieval and gene analysis of different species include [2], [3], [5], [9], [10]. Compared to these existing similar databases, CoccoExpress has several new and unique features. First, its dynamic website is developed based on the most current tools and programs available today in the market. Thus CoccoExpress has relatively better manipulation speed and less overhead. Second, many of the exiting databases have very limited search capabilities. Built-in rigid queries limit the user's choices in retrieval of the information. Occasionally, researches may have to send special request to the administrators of the database and ask for specific data that could not be easily queried directly from the provided web interface. CoccoExpress relaxes this constraint by providing a very simple customized search. Researchers can set any valid criteria that can be acceptable by the database server on any field without having the knowledge of structural query programming.

In addition, many of the exiting databases have developed their own packages to secure their data, provide help, or administrate their information. In general, nowadays, developing these tools for databases are waste of time. CoccooExpress takes advantage of the open source tools such as phpAdmin, VikiPedia, and integrate them with the database.

III. Planning

The overall structure of CoccoExpress can be divided to three main sections: Front-end, Backend, and Database. Each section was carefully planned to accommodate our objectives of building an efficient system.

The main objective of CoccoExpress engine was to provide a simple platform to be used by scientists especially by biologists. For platform's operating system, Linux was picked because of its exceptional features and security that it provides for multi-user projects. Further, Unix-based operating system (OS) is the primary OS for bioinformatics tools and research.

To create a dynamic front-end, we employed different client side techniques such as Cascading Style Sheets (CSS), JavaScript and Asynchronous JavaScript and XML also known as AJAX.

PHP programming language was adopted for constructing the backend to provide a bridge between the database and front-end. This decision was made after evaluating several criteria such as:

- Naturally optimized for Linux servers
- Database access layer support
- Web development friendliness
- Development speed and
- Runtime performance

Meanwhile, we decided to make use of Perl and C++ for some of the back-end routine functionality such as backup rolling and data parsing. Generally speaking PHP was picked for web development and Perl/C++ was reserved for batch processes.

Back-end database was indeed the core of this project. Our goal was to select a database that could be capable of running stable on Linux platforms. Based on the structure and interrelationships of our data, MySQL, Oracle and PostgreSQL were nominated to perform as our backend database.

Although PostgreSQL was proven very stable with large datasets, different benchmarked suggested it may not be qualified for heavy queries of CoccoExpress. Oracle and MySQL both were equally well qualified for housing the CoccoExpress data; however, MySQL was picked primarily for lower cost of ownership, better compatibility with PHP, and ease of scalability.

IV. Implementation

A. Back-end and Database

It is expected that the CoccoExpress front-end search engine is subject to be used by scholars from all around the globe. With that in mind, some of the main challenges during planning of front-end were:

- Search engine friendly so scholars could find CoccoExpress by searching the main web search engines.
- Dynamically maintained and redesign
- Several methods of searching the database from simple to advance searches.
- Scalability to dynamically adopt the new species added to the database
- User friendliness to provide robust navigation
- Help and documentation to match the front-end and be accessible through out the entire site for given elements

Fig(s). 1 and 2 demonstrate the front-page of CoccoExpress and the home page of Emiliania Huxleyi, respectively. Sections 1 and 2 in fig. 1 and sections marked as 1-4 in fig. 2 are dynamically built using data pulled from database. Section 1 of fig. 2 demonstrates search engine friendly

naming conventions of CoccoExpress pages. This well-formatted URL is picked by search engines and is easy to parse by search engine's back-end parser. It is important to remember that "/Coccolithophorids/Emiliania-Huxleyi/" is not an existing folder on web server; instead, it is a dynamically built URL for better structure and search engine optimization.

Fig. 1: Home page of CoccoExpress

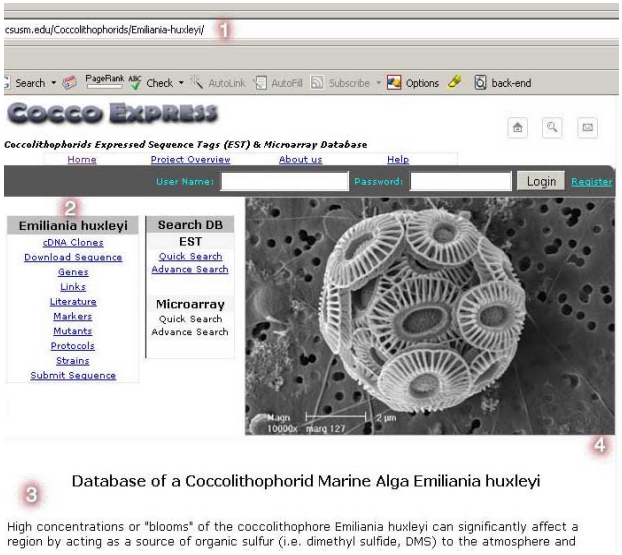

Fig. 2: Home Page of Ehux

A. *Quick Search:*

Quick Search was implemented to provide a fast method of searching the database for different records of specie. Fig. 3 demonstrates the quick search tool for Emiliania. Scholars can search by different criteria as shown in the figure. Help button is also provided to show help documentation on data type of any given criteria.

Fig. 4 demonstrates the results of a quick search submitted with "F2", a library name where some of the EST clones were generated, as criteria. Once this result is sent to client's browser, client can easily sort the result by desired column in desired order. As it is demonstrated in the figure, users can click on the record to see more details about any given column. This request for more details is sent to backend system using Ajax, once the answer to request is available, results page is updated accordingly. This will allow fast drilldown to details of any given record to find the desired record without leaving the page.

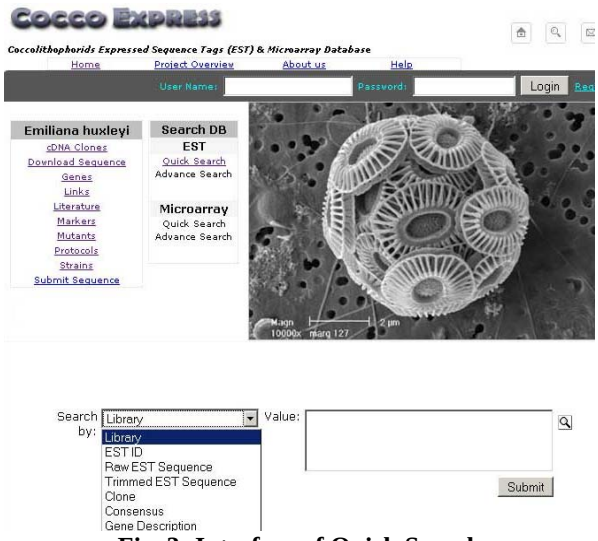

Fig. 3: Interface of Quick Search

Fig. 4: Results generated from Quick Search

B. *Advance Search:*

Quick search was implemented with a very straight forward sequence flow in mind. While it can provide a fast and easy access to records, it has several limitations by design. These limitations (i.e. lack of ability to select other fields or criteria versus predefined options) led to implementation of a more sophisticated search system. Once again, the challenge was to build a system that could execute advanced queries with an easy to use interface that could be used not just by computer scientists but also by scholars of other fields, mainly biologists.

Fig. 5 demonstrates the interface for advance search. As shown in section 1 of this figure, fields for selected tables are pulled dynamically from database to construct series of windows with fields of each table. If a field is added/removed from a table in database, this form will update automatically to reflect such changes. By using advance search, user is capable of selecting desired fields to be pulled from database (section 2 in the figure). In addition, users can specify desired criteria for any given field. (section 3 and 4 in the figure)

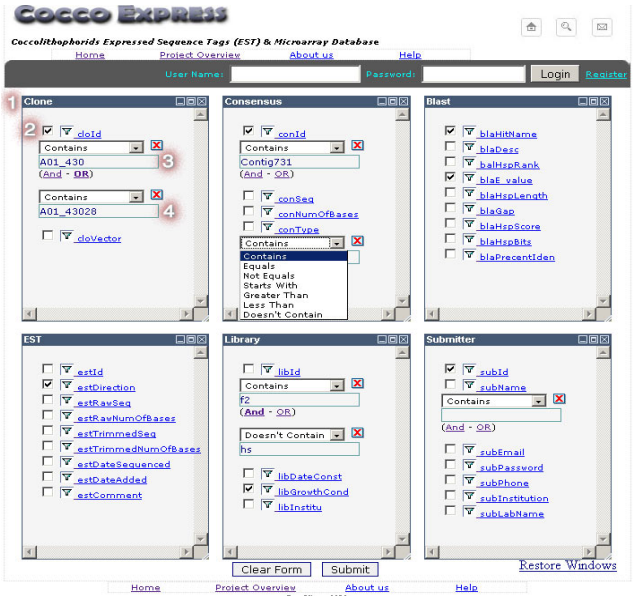

Fig. 5: Interface of Advance Search

Each window in advance search represents a table of CoccoExpress database. These windows are constructed with ability to shrink or grow in size based on available screen size and total number of tables and fields. For user comfort, these windows can be moved around and resized manually to achieve desired length and width just like any other window based desktop application.

Based on the assigned/selected criteria, CoccoExpress advance search automatically generates queries for users. Note that manually writing a query can have several disadvantages. First, users are required to have an in-depth knowledge of our backend database. Second, user written queries are usually written and tweaked several times before desired results is achieved, wasting both user's time and server's resources. Third, user created queries are often not well-optimized and do not use proper indices. Therefore, they can be slow and take up a lot of resources to run. Finally, to write such queries, knowledge of database and SQL is required.

C. Back-end and Database

Back-end system is written primarily in PHP. In most cases PHP is used to generate the HTML and Apache is responsible for running PHP and streaming HTML over to user's browser. Earlier, we mentioned that MySQL is picked as our back-end database system. This allows us to take advantage of compatibility between PHP and MySQL. MySQL connection is established at the beginning of each page by including a centralized header. Later, this connection is used within the PHP script to access the database for various reasons such as dynamically building the navigation menu section. We deployed two primary databases: one is used by CoccoExpress and the other serves as data warehouse for CoccoPedia, the help database of CoccoExpress. CoccoExpress database structure consists of eleven tables, out of which nine are used primarily to house genomic data and two are used to control CoccoExpress front-end interface.

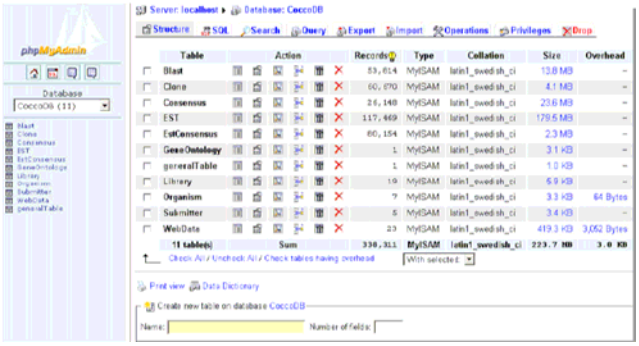

Fig. 6: Interface of phpMyAdmin

D. Database Maintenance

To easily access and work with the database, phpMyAdmin is deployed on the server. phpMyAdmin is an open source software that creates a web-based interface for MySQL database (see Fig. 6).

With phpMyAdmin, administrators of the CoccoExpress can simply maintain and modify the database with minimum knowledge of database administration.

E. CoccoPedia: The centralized help system

A centralized help system was an absolute necessity since CoccoExpress consists of several technical terms and has a unique structure. After some research, we decided to adopt an already developed platform rather than reinventing the wheel.

This help system is created by taking advantage of open-source software also used by WikiPedia. The help system is linked to different places of CoccoExpress to provide definition and help on different technical terms. Fig. 7 shows a snapshot of CoccoPedia entry for Emiliania Huxleyi.

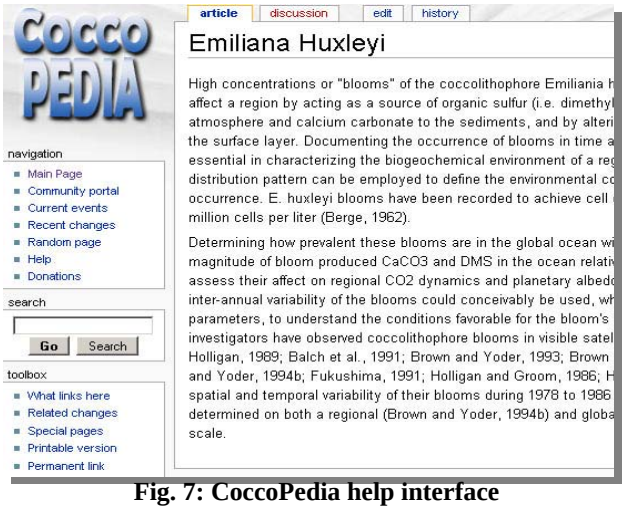

Fig. 7: CoccoPedia help interface

While we could manually link to CoccoPedia's articles and definitions from anywhere in front-end website of CoccoExpress, it seemed too much of a hassle to check CoccoPedia for existence of a page and manually linking all

occurrences of that term within the entire site. It was even more hassle to manually go back and update all occurrences of a term every time a page was added, deleted or modified in CoccoPedia. CoccoPediaLinker was employed to overcome this problem.

CoccoPediaLinker is a parser system that is enabled by default for all the pages in CoccoExpress website. It can be turned off manually per page. It is responsible for parsing the page content and examining the content against available articles of CoccoPedia. If a term is found in CoccoPedia engine, CoccoPediaLinker will link the term to CoccoPedia entry automatically. This will provide a well documented front-end for CoccoExpress. More importantly, documentation of CoccoExpress pages will get richer as more articles are added to CoccoPedia.

Fig. 8 demonstrates the term "Coccolithophorids" that is picked up by CoccoPediaLinker and is linked to proper CoccoPedia article page.

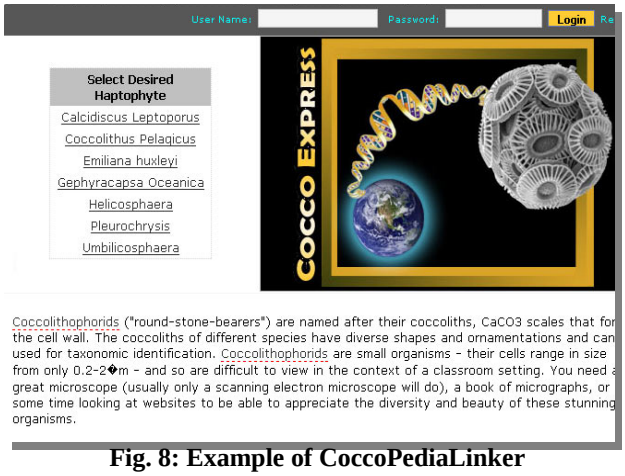

Fig. 8: Example of CoccoPediaLinker

We previously pointed out that most pages on CoccoExpress were dynamically built using data in back-end database. While planning and writing CoccoPediaLinker, we had to overcome the challenge of going through all contents of CoccoExpress every time a change was made in CoccoPedia and update them accordingly whether they were static or dynamic. This could be a time consuming process using a lot of server resources as CoccoExpress scales over time. Instead we took a different approach. Basically, CoccoPediaLinker is designed to start the parser engine for each page after rendering of the page is completed in user's browser. Once pages is fully rendered, CoccoPediaLinker will go through all available content of the page no matter how they were collected and will link each term to appropriate CoccoPedia article. This will eliminate the need to examine dynamic content and static content separately every time a change is made. The links built by CoccoPediaLinker are underlined with dashed line to be distinguished from normal links of the site. CoccoPediaLinker is built with some intelligence to avoid unwanted link creations such as modifying an already linked keyword or modifying keywords within html tags and attributes.

F. Statistic Software

In order to measure traffic of CoccoExpress and enhance search engine optimization, we needed a traffic analyzer system. Once again, after proper investigation, we adopted an open-source system instead of reinventing the wheel. Apache was configured to produce combined log of all requests made to http://bioinfo.csusm.edu. Awstats (http://awstats.sourceforge.net/) was deployed to parse these apache logs, analyze and create series of web-based reports.

Fig. 9: Example of Awstats traffic report

Fig. 9 demonstrates a sample Awstats traffic report. It is scheduled to update the reports once a day. Traffic stats are available through http://bioinfo.csusm.edu/stats/.

V. Future Work and Conclusion

Flexibility, simplicity, and speed of information retrieval are the main features of the CoccoExpress. However, as we mentioned earlier, currently, CoccoExpress includes the Ehux data only and will eventually be loaded with the EST of other species of Coccolithophorids. Therefore, CoccoExpress overall performance will have to be tested and modification may be required. Further, there is also an extensive amount of active research and planning being conducted in creating microarray database [11]. Once microarray database is designed, it can be linked to EST database and some possible cross-database functionalities could be added.

Basically, CoccoExpress was designed to serve as the primary tool to access and manage EST database of Coccolithophorid. This project was approached with two primary objectives of advance usability and easy maintenance/usage. It was originally designed as EhuxExpress [4]. Due to requirements for several fundamental enhancements, it was redesigned and built from ground up to better accommodate the requirements of the fast growing data set. As further research is conducted on the genome analysis of Coccolithophorids and more data are added to the CoccoExpress, we will benefit more from dynamic nature of CoccoExpress and the flexibility of its search engine.

ACKNOWLEDGMENT

We thank members of bioinformatics research group specially Dr. Betsy Read and Dr. Thomas Wahlund for their effort in preparing the data for us to be placed in CoccoExpress. Special thank also goes to Dr. Zhang and Computer Services at California State University San Marcos for setting up and maintaining the database servers.

REFERENCES

[1] B. Nguyen, R. M. Bowers, T. Wahlund, and B. A. Read, "Suppressive subtractive hybridization of and differences in gene expression content of calcifying and noncalcifying cultures of Emiliania huxleyi strain 1516," *Applied and Environmental Microbiology*, May 2005, pp. 2564-2575.

[2] A. Caprera, B. Lazzari, A. Stella, I. Merelli, A. Caetano, and P. Mariani, "GoSh: A Web-based database for goat and sheep EST sequences," *Bioinformatics*, vol 23, no. 8, 2007, pp. 1043-1045.

[3] N. D'Agostino, M. Aversano, L. Frusciabte, and L. Chiusano, "TomatEST database: In silico exploitation of EST data to explore expression patterns in tomato species, " *Nuclectic Acids Research*, 2007, vol 35, Database issue, pp. D901-D905.

[4] M. Ebert, "A EhuxExpress: Novel genetic database for the marine alga Emiliania Huxleyi," Master project, California State University San Marcos, 2004.

[5] M. Hiller, S. Nikolajewa, K. Huse, K. Szafranski, P. Rosentiel, S. Schuster, R. Backofen, and M. Platzer M, "TassDB: A database of alternative tandem splice sites," *Nuclectic Acids Research*, 2007, vol 35, Database issue, pp. D188-D192.

[6] M. D Iglesias-Rodriguez, I. Probert, and J. Batley, "Microsatellite cross-amplification in coccolithophores: application in population diversity studies," *Hereditas,* 2006 Vol 143, p. 99-102.

[7] Maheswari, U., et al., The Diatom EST Database. *Nucleic Acids Research*, 2005, vol 33, Database issue, p. D344-D347.

[8] N. Ozaki, S. Sakuda, and H. Nagasawa, "A novel highly acidic polysaccharide with inhibitory activity on calcification from the calcified scale "coccolith" of a coccolithophorid alga Pleurochrysis haptonemofera," *Biochem Biophys Res Commun*, 2007, Jun 15, vol 357, no. 4, pp. 1172-1176.

[9] E. A. O'Brien, L. B. Koski, Y. Zhang, L. Yang, E. Wang, M. W. Gray, G. Burger, and B. F Lang, "TBestDB: a taxonomically broad database of expressed sequence tags (ESTs)," *Nucleic Acids Research*, 2007, vol 35, Database issue, pp. D445-D451.

[10] N. Pavy, J. Johnson, J. Crow, C. Paule, T. Junau, J. Mackay, and E. Retzel, "ForestTreeDB: A Database Dedicated to the Mining of Tree Transcriptomes," *Nuclectic Acids Research*, 2007, vol 35, Database issue, pp. D888-D894.

[11] P. Quinn, R. M. Bowers, X. Zhang, T. M. Wahlund, M. A. Fanelli, D. Olszova, and B. A. Read, "cDNA microarrays as a tool for identification of biomineralization proteins in the coccolithophorid Emiliania huxleyi (Haptophyta)," *Appl Environ Microbiol*, Aug 2006, vol 72, no. 8, pp. 5512-5526.

[12] A. Soto, H. Zheng, D. Shoemaker, J. Rodriguez, B. A. Read, and T. M. Wahlund , "Identification and Preliminary Characterization of two cDNAs Encoding Unique Carbonic Anhydrases from the marine alga Emiliania Huxleyi, " *Applied and Environmental Microbiology,* Aug 2006, vol 72, no. 8, pp. 5500-5511.

[13] T. M. Wahlund, A. R Hadaegh, R. Clark, M. Fanelli, and B. A. Read, "Analysis of Expressed Sequence Tags in Emiliana Huxleyi," *Nature*, 2003, vol 377, pp. 320-323.

[14] T. M. Wahlund, X. Zhang, and B.A. Read, "EST Expression Profiles from Calcifying and Non-Calcifying Cultures of Emiliana Huxleyi," *Micropaleontology*, Dec. 2004; vol. 50; no. Suppl_1, pp. 145-155.