CEMAP II: An Architecture and Specifications to Facilitate the Importing of Real-World Data into the CASOS Software Suite

Terrill L. Frantz and Kathleen M. Carley

August 1, 2008
CMU-ISR-08-130

Institute for Software Research
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Center for the Computational Analysis of Social and Organizational Systems
CASOS technical report.

This work is part of the Dynamics Networks project at the center for Computational Analysis of Social and Organizational Systems (CASOS) of the School of Computer Science (SCS) at Carnegie Mellon University (CMU). This work is supported in part by the Office of Naval Research (ONR), United States Navy, N00014-06-1-0104. Additional support was provided by National Science Foundation (NSF) Integrative Graduate Education and Research Traineeship (IGERT) program, NSF 045 2598, NSF 045 2487, the Air Force Office of Scientific Research, FA9550-05-1-0388 under a MURI on Computational Modeling of Cultural Dimensions in Adversary Organizations, the Army Research Institute W91WAW07C0063, and CASOS. The views and proposal contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the Office of Naval Research, the Air Force Office of Scientific Research, the Army Research Institute, the National Science Foundation, or the U.S. government.

Keywords: source data, social network, ORA, AutoMap, software, DyNetML, LOOM, spatial data, dynamic network analysis

Abstract

An often overbearing logistical problem that social network researchers and analysts face is the challenge of carrying out the basic data processing steps necessary to transform real-world source data into a form that is formatted specifically to the requirements of particular social network analysis software. In particular, this paper focuses on resolving the variety of real-world data forms that must be transformed into the format necessities of the software suite developed at the Center for Computational Analysis of Social and Organizational Systems (CASOS) at Carnegie Mellon University. This data processing or programming problem, while rather straightforward, can be daunting to non-I.T. users of these —and any-- software tools, and at best case, laboriously costly for the programming savvy. This report outlines and describes a software architecture and user paradigm to address this ubiquitous problem. This report provides a basis for later detailed design and implementation of the ideas put forth in this report which will ultimately results in tangible features being made available and fully operational for users of the CASOS software suite (ORA, AutoMap, and Construct).

Table of Contents

1 Introduction

An overbearing logistical problem that social network researchers and analysts face is the challenge of carrying out the basic data processing steps necessary to transform real-world source data into a form that is formatted specifically to the requirements of social network analysis software. This omnipresent problem arises for all software used in the field. The complications of real-world data involve basic data formatting, the collection of the data from variety of physical locations, the aggregation of relevant datum from a multitude of separated sources, and often performing detailed field mapping and masking. This general data processing or programming problem, while rather straightforward, can be daunting to users of these —and any-- software tools, and at best case, laboriously costly for even the most programming savvy.

One can easily foresee that 80% of the analyst's time is currently wasted on messaging the data to fit the software, which leaves little precious time for the true value-add of the analysis step. By themselves, the analytic tools developed at the Center for Computational Analysis of Social and Organizational Systems (CASOS) do not differ from this omnipresent data processing requirement. The tools included in the CASOS suite (Carley, Diesner, Reminga, & Tsvetovat, 2004) consist of the Organizational Risk Analyzer (ORA; Carley & Reminga, 2004), AutoMap (Diesner & Carley, 2004), and Construct (Carley, 1991). The technique and architecture described in this document address and provide a solution for this overbearing real-world data processing problem specifically for users of the CASOS suite of tools, by describing a simply, yet powerful, process for bridging the two separate worlds of data and analytic software.

This report outlines and describes a software architecture and user paradigm to address this ubiquitous problem. We organize this report in a manner that first provides the specifics to several example scenarios and problems that are faced, then we describe the CEMAP II architecture which reduces the problem considerably. Next, we provide important terminology used and further describe the details of CEMAP II, then demonstrate the operational aspects of the solution, which we identify as CEMAP II. Finally, we discuss the expandability features of the architecture then some notable workarounds and conclude with a short Forward Section.

1.1 DyNetML Input for ORA

CEMAP II's purpose is to transform real-world data into relational network data that can be used by ORA. ORA operates on data in the DyNetML format. DyNetML (Tsvetovat, Reminga, & Carley, 2003) is the XML-based document markup language that ORA relies upon. It is this format that allows for fully representation of the meta-network data and serves as the native input format of ORA. Producing network data in this DyNetML format is a primary purpose of CEMAP II. An extension to DyNetML for ORA's LOOM feature (Davis, Olsen, & Carley, 2008), call walksets is also an output of CEMAP II. CEMAP II will be expandable so as DyNetML changes, so can CEMAP II.

1.2 Text Files Input for AutoMap

A second purpose of CEMAP II is to transform real-world data into data files that can be used by AutoMap. The AutoMap software expects documents to be input as basic flat, text-only files. CEMAP II provides a mechanism to create these files from real-world source data. CEMAP II will permit the operator to have complete control over the file names of these data files as they often need to consist of important datum that ties the data file to a entity. As a byproduct of this feature, CEMAP II will also be equipped to output flat files in CSV, tab-delimited, or other similar output formats that may be used as input to other software applications.

1.3 Input for Construct

The CEMAP II architecture is fundamentally designed to enable the user to create input network files for the initial set-up of Construct, however this feature will be fully explored and evaluated for use after CEMAP II meets is initial goal of serving ORA and AutoMap.

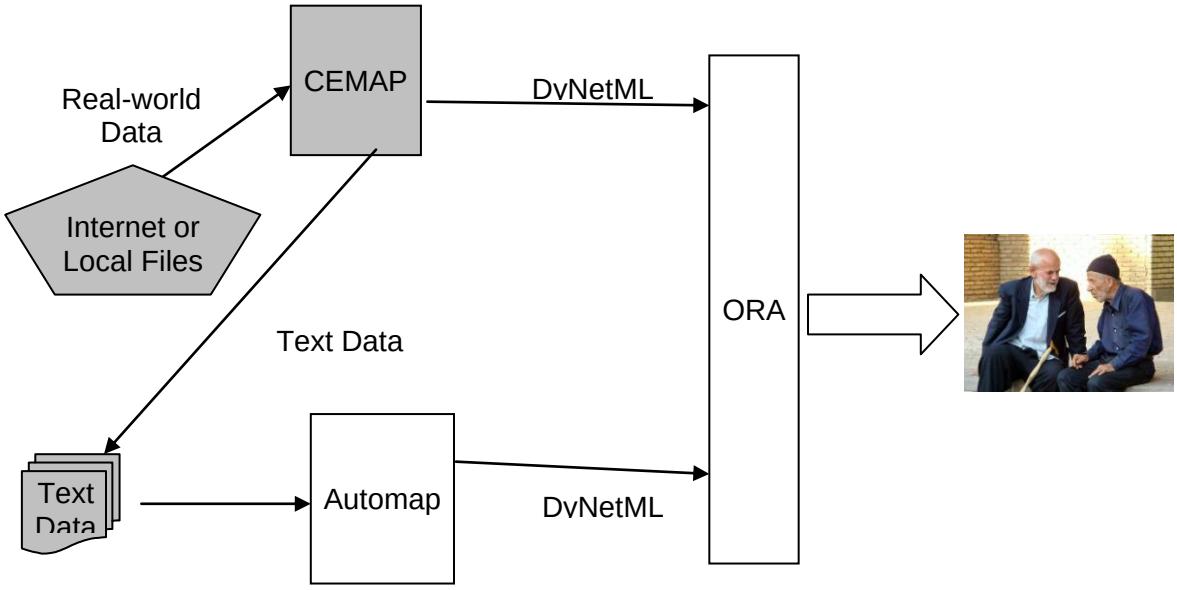

Figure 1. Primary functionality of CEMAP II.

2 Example Real-world Data Source Scenarios

To provide a sense of the scope of the circumstances and the features, functionality and need for a tool like CEMAP II, in this section we present several examples of real-world data sources that are commonly destined as input to ORA or AutoMap. We describe a few of the scenarios in substantial detail, particularly email, so as to illustrate the complexity that can come with such real-world data, and to provide a sense of the power and intended features of CEMAP II. These scenarios provide instances of various data formats, as well as the location of the data physically that is within the scope of the CEMAP II architecture and designed functionality.

2.1 Email Data

With the ubiquitous arrival of email and its inherent use to network analysis, we will use it as an example of the technology issues faced by a social network analyst when attempting to import email data into ORA and AutoMap. (Microsoft Outlook users see Section 8 for an important workaround) Email is a ideal example of a real-world data source that is rich in information of great interest to the social analyst, but must be obtained from a technology-dense setting that is often quite foreign to the analyst, even to the technology-savvy. This massive data repository of email forms a planet-level, intertwined data-set that social computing researchers widely recognize as being rich in social information—if the data can be harvested efficiently and effectively analyzed. The analyst must somehow operationalize real-world email data and prepare it for use in ORA and/or AutoMap.

The email exchange that two people have, is actually an exchange between two computers, often with several other computers serving as intermediaries in the delivery process. Ultimately, a specific message sent by one person is received by the target person, in its electronic form, via a client software program that picks up the email from its host email server. The email message rests on a designated email server until the receiver "picks up" the message from the server; this is practically identical to the process of picking up hardcopy mail from the post, but in electronic form. The electronic email can be delivered to the post office repository for you to physically pick up, or directly to your personal mail box for you to pick up. Once you pick up the message, via your email client software, you can read it, print it, store it, or follow up on its contents, if you desire.

There are several hurdles that must be overcome. First, the physical location of the email data is complicated. In the email architecture, the message can be delivered to the equivalent of the post office lobby-box, to what is called an IMAP server. Alternatively, the email can be delivered to your personal mailbox, what can be called a Post Office Protocol (POP) server. There are many differences between these two email servers, and your email account is most certainly one or the other of these. The primary difference between an IMAP and a POP email server is the storage feature of the server. An IMAP server will allow you (or your email client software) to persist, or store, your email physically on that server. A POP server only serves as a temporary holding station for a message that is removed once it has been retrieved by your email client software. An IMAP server is enabled to store the message even after the email has been initially retrieved. It should be noted that the POP protocol calls for an email to be removed from the incoming mail box once it has been retrieved, however some software extensions allow for a read-only access to the POP inbox, resulting in the message remaining in the inbox when retrieved and is therefore managed by the client software level. While emails are initially managed via a server, often times the user has an option to move or copy the email message from their inbox into a separate folder for storage. These folders can be physically located on either the original email server, or on a disk local to the computer user as an operating-specific data file.

Secondly, the format of the individual email message can be in one of several detailed data formats. The most widely used format is MBOX format, which is a world-wide standard that allows for different email client software programs to access the email from the server, IMAP or POP, without confusion. The MBOX format specifies that

email is represented as a text file where each email begins with a “From ”-labeled field and ends with a blank line. Between these delimiters the email is formed into two sections: the Header Section and the Body Section. The Header Section holds the envelop-level data such as the To:, From:, Subject:, Date:, et cetera. The Body Section holds the message text and file attachments, if present. Figure 2 shows an example email message in MBOX format. The header section consists of four header fields: “From”, “To”, “Subject” and “Date.” And the Body Section is below the Header Section, delimited by a blank line.

```
From - Wed Feb 27 09:10:44 2008
From: "tom Thumb" <tthumb@msn.com>
To: "terrill frantz" <terrill@org-sim.com>
Subject: Journal Article – For review
Date: Fri, 22 Feb 2008 22:01:26 -0600

Terrill,
Please submit the article review by next
weekend.
Cheers,
Tom.
```

Figure 2. Example email message in MBOX Email Format.

Finally, the data making up an email which is rich in information about the particular instance of the communication network. The Header Section contains transactional data such as who emailed who, when, and about what (the subject line) that can be represented as a network. The email’s Body Section contains text that can be processed into a network by applying a natural language processor to it to also construct a network configuration .

Focusing on the header data, there are several bi-modal networks that can be constructed: From-task, task-to, task-cc, task-bcc. From these four principal networks, various secondary networks can be created by folding them together in various ways. For example, a From-To network can be constructed by folding the From-Task and the Task-To networks together. Most often, an analyst will construct a person-to-person social network, which can be constructed by folding the From-task, task-to, task-cc, and task-bcc networks into a single network representation that captures the person-person network. Several other socio-technical networks can be constructed from the header data, but we limit ourselves to a discussion on the actor to actor networks in this paper.

2.2 Excel (& CSV) Data Files

For a variety of reasons, the most common format that an analysts starts with before using ORA is data stored in an Excel spreadsheet. This form is by and large the format that most non-technical analysts are comfortable with and accustomed to. Synonymous with the Excel spreadsheet format, is the CSV file. The CSV (Comma Separated Values) file has its data columns delimited by a comma character and the rows delimited by a linefeed or similar other special character. Often times utilized quotations for quoting character strings that may contain the delimiter character(s). Certainly other delimiter characters can be used in this type of file format, such as the tab character.

2.3 SQL Database

Many institutional users, those who work in a work-group or within a large organization, and power users have their data stored in an SQL database such as mySQL, Oracle, etc.. Reading directly from these databases is an important capability of CEMAP II. The data is often reachable over IP or on the users local machine and is inherently available in tabular format -- as a characteristic of the relational database paradigm and SQL platform. (SQL users should see the Workarounds Section (#8) for an important note).

2.4 Web Scraping

A rich real-world data source for analysis data is the World Wide Web (WWW). The WWW consists of a web of hyper-links connecting separate HTML pages. Each HTML page can have useful data for analysis and frequently the linking relationship among the pages is in itself the phenomenon of study. CEMAP II will facilitate the automated crawling of web pages and the subsequent scrapping of the HTML that, combined, can be represented as meta-networks for ORA and unstructured text files for AutoMap. CEMAP II enables the user to automatically troll the WWW and scrap the HTML content according to various parameters set by the user. From this process both network and AutoMap data files can be constructed.

2.5 Web Services

An exciting real-world data source for analysis data are web-based social networking sites, retail sales vendors, and other web-based tools and sites, even the CASOS SOAP APIs. CEMAP II recognizes that web services are the future of distributed computing as evidenced by the early and rapid growth of public SOAP and RPC APIs. CEMAP II has a web service client built into its architecture so that extensions to the core features can be made readily available for expansion. For example, the LinkedIn networking site is rich with data that can be harvested by CEMAP II, as does Amazon.com. By making these entry points available in CEMAP II, ORA and AutoMap users can, with procedural ease, explore these real-world data sources using the features of ORA and AutoMap without having to involve programmers or writing your own customized data parser.

2.6 Hybrid-Combined Data Sources

Occasionally, the network analyst is faced with the task of having to meld two or more real-world data sources together to generate the meta-network or text data for ORA or AutoMap, respectively. An immensely powerful feature of CEMAP II is to make this process simple to configure and trivial to execute. For example an analyst may use the company database data as their primary data source, but also have a personal spreadsheet with local information about the entities that needs to be combined into one output (network or text files). CEMAP II accommodates this important requirement, which is particularly important for the production-oriented analyst who replicates the same process routinely every day, for example.

3 Application Architecture

CEMAP II is available as a menu item in both ORA and AutoMap, but can also be executed independently as a stand-alone interactive program, or as a stand-alone, scripted batch program. The primary purpose of CEMAP II is to prepare and format raw source-data for use as relational network input data files for ORA, or unstructured text input files for AutoMap; although CEMAP II can also produce formatted output data files for other purposes, e.g. input to Excel, etc. CEMAP II's interactive interface provides a simplistic process for analysts to prepare their source data in either, a stable and routine manner, or in a frequently changing, exploratory manner. With ease, the user can repeatedly process the source data while tweaking the resulting output to their specifications. Once the CEMAP II process is producing the desired results, the profile for the process can be saved for easy re-use at a later time, by an individual or a specific group of users.

The key to CEMAP II's effectiveness is the method in which it separates the physical characteristics of the source data, which is often the domain of computer programmers, from the characteristics of the stylized, reformatted meta-network and unstructured data necessary for analysis, either in ORA or AutoMap. These two very different perspectives on data are joined together, as is necessary, by a simple drag-and-drop mapping process that conjoins the source data which is viewed as straightforward, row-column structured tables, and the output data format – usually DyNetML or text files.

There are three essential steps the user ultimately must perform: (a) identify the input tablesets, (b) identify the output templates, and (c) set up mapping between the columns of the tables in the tablesets and the requirements of the various output template fields. However, these steps can be combined and hidden from a user such that a simplistic two-click process may be situated.

3.1 Architecture Blueprint

The entire design of CEMAP II takes advantage of the notion that data stored in a tabular form has an inherent characteristic that each data item in a row is related in some comprehendible manner to any other data item in that same row. Since network data is essentially all about relational data, therefore, any pair of data items in the same row can form a link in a network. In short, given a table of data, each column can be the source for a DyNetML nodeclass, and each pair of data items in a row (or column by column) can be the source for a DyNetML network. It is this inherent characteristic of a data table that CEMAP II takes advantage of. CEMAP II provides features to convert real-world data into a tabular form and to describe DyNetML and AutoMap formats in a simplistic manner. CEMAP II then facilitates the mapping of the tabular data into the output format according to the desires of the analyst. Essentially, CEMAP II conveniently separates the technical task of formatting real-world data into a generic, consistent and easy to understand table from the network-centric vantage point that an analyst holds, then provides a simple mapping mechanism to join the two representations.

3.2 Design Principles

The purpose of CEMAP II is to provide a pathway from the complexity of real-world data to the CASOS Software Suite. It can only accomplish this goal if the process of using CEMAP II is less complicated than the alternatives and if it does indeed save time, other resources, and eliminate the frustration inherent with having to deal with real-world data at the technology level. With this in mind, the CEMAP II GUI is a critical component of the tool and it is essential to get the GUI right; that is to say that the user interface is of utmost importance to the viability and the usefulness of CEMAP II. Another critical aspect of CEMAP II is its extendibility feature, which is described in later section. The future will most certainly bring about new and different real-world data sources that cannot be dealt with today, and this includes both public and private data sources. It is utmost importance that CEMAP II software have the hooks built in for others to extend the features set. This expandability will be solely focused on two aspects of CEMAP II: (a) the ability to develop and use custom tables and tablesets, and (b) the ability for the user to develop and use custom functions for field-level data manipulation – for example, removing the ".com" off a URL.

4 Primary Components

There are four primary component parts to the CEMAP II architecture and user-design that even than the most basic user (a user that only is concerned with completed profiles) of CEMAP II should be somewhat familiar with. The four components are described in this section and show in the Fig. 3 graphic below.

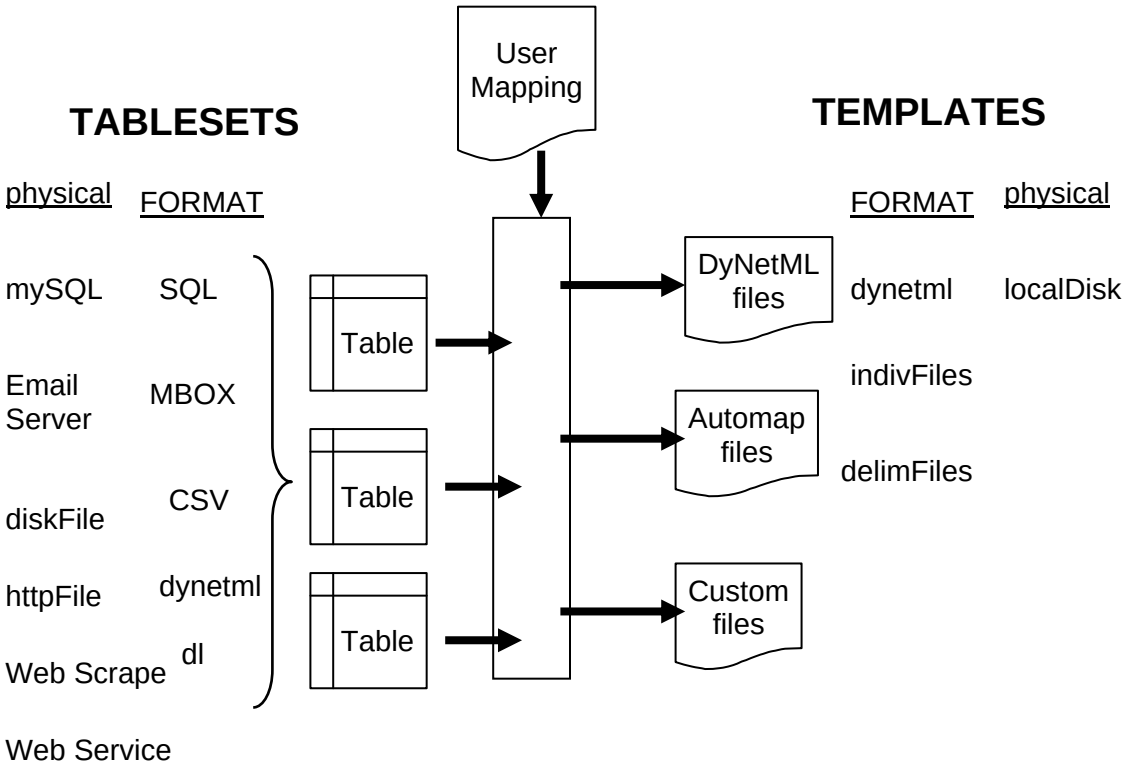

Figure 3. High-level architecture of CEMAP II; All combined is a "Profile".

4.1 Templates

A template is a description of an output format from CEMAP II. These outputs most often consist of DyNetML templates, but there are numerous output templates that describe other output formats. Most of the output templates can be customized in some manner, according to the designer of the template. For example a DyNetML template allows the user to add, delete or change the characteristics, of a nodeclass, or network, among other several features of ample flexibility. The user has a great amount of flexibility in the flat file template as they are not constrained by the strict requirements of DyNetML. For example, a template can be customized to allow for the user to create a CSV file to import the output data into an Excel spreadsheet. This is merely by-product of the CEMAP II feature set, which is designed expressly for the CASOS software suite.

4.2 Tablesets

A tableset is a collection of tables that logically belong together either in the manner in which the original data is stored, or from the perspective of the user. For example, a collection of SQL statements involving a single database host and password, etc, can situation for a tableset, with each table corresponding to an SQL statement, but with the same characteristics as the other tables within the same tableset. A tableset translates the physical data source into a row-column oriented table that can be mapped with one or more other tables or template. A tableset takes much., most, or all of the complexity out of a user reaching an original data source for their subsequent ORA/AutoMap analysis.

The tableset construct is a form that situates CEMAP II for an unending amount of flexibility in interfacing with real-world data. The underlying code for each tableset deals with all of the peculiarities of the real-world data and transforms that data into a table, or set of tables. This is all the tableset designer, or developer, needs to be concerned with. There is no notion of a network or DyNetML to the tableset view.

4.3 Profiles

A profile is a file whose main role is to keep the completed mappings of a tableset and template(s). There are usually the three logical aspects of a process fully defined within it. This is a CEMAP II file that has a set of template(s), a set of tableset(s), and the mapping between the two. A mapping without a tableset is impossible. A mapping is always associated with a template. The profile can have any, all, or none of the fields within the template, tableset completed, or have the matching completed. A profile might correspond with a routine task that the user has to perform often or periodically. The profile is meant to "remember" all of the details pertinent to the CEMAP II task, also some details can be purposely left unsatisfied by the creator of the profile (the actual name-location of the output file, perhaps, among other things like high-confidential passwords and such).

4.4 Resources

A resource is a mapping of a physical file, data or CEMAP II specific file (profile, tableset, template, or other resource file), that the user has access to. This resource can reside anywhere reachable to the user's computer, e.g. local disks, the Internet, etc. The

function of a resource is to simplify the reaching of the file to the user. That is it saves the user from remembering long URL's or file paths, etc. There are two main types of resources, a resourceDirectory and a resourceFile. Locations for either can be on a local disk, or on the Internet. The resourceDirectory indicates the location of a group of resourceFiles, therefore can be a local disk directory, a local or Internet-based java jar file, a local or Internet-based zip file, or the CEMAP II system files embedded within the CEMAP II software. A resourceFile can be a file on the local disk, the Internet or within the java jar or zip file specified by a resourceDirectory. A resourceFile is an xml document file that has "xml" as its file name extension and contains the root xml element, <CasosCemap>.

5 Mapping Process

The cornerstone of CEMAP II is the process by which the user indicates the source of the data that is transformed into the format specified by a template.

5.1 Description

This data is made available to CEMAP II via a table defined in a tableset. The process by which the output template is associated with the tableset table is call "mapping." The mapping process is simplistic with very few, but strict, rules. The user will have a flexible drag and drop mechanism in the CEMAP II GUI to carry out the mapping process. To set the mappings for a profile, the user will identify the template field that they are working on – the output field in the DyNetML or unstructured text file. In the case of DyNetML file, this output field could be a node, a node attribute, a link, or a LOOM walkset, among other common possibilities. Next the user indicates the specific table that contains the data that should be mapped to the output field. Once the table is identified, a specific column is selected and it is this exact column, table, and tableset combination that is associated with the output field. Once mapped, a complete profile has been created, which can be stored for later use, or executed immediately.

6 Operations

The core process that defines CEMAP II is actually a batch process, however, most users for CEMAP II will never see this aspect of CEMAP II. Like an iceberg, the real engine is below the surface, but the majority of attention to CEMAP II will be just the surface of the over all tool. The Graphical User Interface (GUI) is the actually larger part of CEMAP II, the most difficult to program, and the most difficult to learn. In the ORA and AutoMap implementations of CEMAP II, the user spends most of their effort, and likely hidden from them, on simply manipulating a xml document that contains all of the information for CEMAP II to process and execute a task. The CEMAP II work task is described in the batch process sub-section below, followed by some more discussion on the GUI aspect of CEMAP II.

6.1 Batch Processing

CEMAP II's workhorse feature is the batch processing. When a ORA or AutoMap user press the execute button, it is this batch process that actually executes for the user. This batch process can also operate as a stand-alone utility for routine production or user-less

processes. The batch process is entirely driven by an script file; this is consistent with the model used for the CASOS Construct software. This script file is xml-based and must have the starting high-level element name of casosCemap. The xml children elements of a casosCemap xml document are: <fields> <tablesets>, <templates>, and <resourceLocators> (see Fig. 4). A casosCemap document can have all or these, or a subset of these three high-level elements. Most likely, it makes sense that there is at least one of these high-level elements in a document. An empty casosCemap document could be utilized as a placeholder file for later completion; this will ensure that the resource file exists, is well-formed xml and is a valid casosCemap document, albeit empty, as expected. These child elements become apparent as to their meaning and use according to the discussion in the components section previous. Essentially, the entire casosCemap document is, and defines, a profile.

```
<?xml version="1.0"?>
<casosCemap version="200808">
      <fields>
      </fields>
      <tablesets>
      </tablesets>
      <templates>
      </templates>
      <resourceLocators>
      </resourceLocators >
</casosCemap>
```

Figure 4. The basic structure of a CEMAP II script document.

6.2 Graphical User Interface (GUI)

As mentioned above, the sole purpose of the GUI interface to CEMAP II is to create, change and manage the casosCemap script document for the batch process. Importantly the GUI does provide the execute button which performs the execution of the profile in computer memory of the GUI. The GUI is the most critical aspect of CEMAP II as it makes the complex task of creating profiles a much simpler and easier to understand operation.

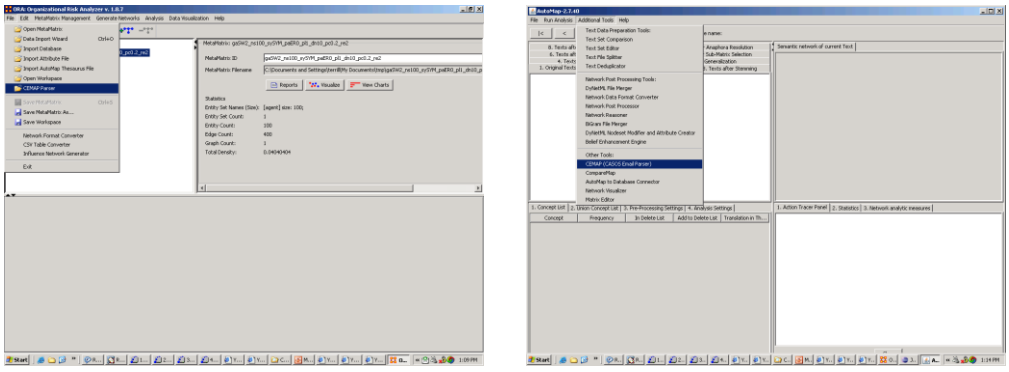

Figure 5. CEMAP II user interface in ORA and AutoMap is identical in all aspects.

The GUI allows for complete maintenance of profiles, tablesets, templates and resources. The GUI makes the mapping process between template and tables an easy and uncomplicated exercise for the user to perform. By design, the CEMAP II GUI in ORA is identical with that in AutoMap, which allows for a simple user transition across the ORA software suite, which requires no extra training or instruction (see Fig. 5). There is however, one very minor, but logical, difference in operation between the two hosts. In ORA, when the CEMAP II user creates a DyNetML network file, the file is automatically loaded into ORA, whether an output file is identified or not. This is not the case in AutoMap as it is obvious that AutoMap does not operate on network files so therefore the automatic load feature of CEMAP II does not occur.

7 Important Features

There are several important features that are characteristic of CEMAP II that necessitate some articulation in this report as they are important to the character of CEMAP II and the user experience in CEMAP II, and overall in ORA and AutoMap. The features highlighted in this section are not a complete set, but are discussed here because they are the most critical or perhaps somewhat novel and not necessarily obvious to CEMAP II stakeholders.

7.1 GUI user start up

When the GUI users starts their CEMAP II session in ORA or AutoMap, CEMAP II automatically loads a special resource file. For a first-time user, a default resource is loaded that comes installed with ORA or AutoMap. This resources profile will have references to the default profile as provided by CEMAP II. These are not user created profiles, but basic email and standard other profiles that the beginning user might be interested in using straight off. This resource file is not changeable by the CEMAP II user, however, the user can create a custom resource file that does allow for customization of the CEMAP II start-up. Whenever this user starts CEMAP II, it will be this customized resources profile that is loaded at the start. This user-custom start-up resources profile is stored on the user's local hard drive disk.

7.2 Component Notes

Rather quickly, when using CEMAP II is becomes apparent for many users that there will be many profiles, tablesets, templates, and resources to understand, maintain and keep track of. The ability to organization these documents is greatly enhances by the tree structure designed into the GUI interface, but there is a great deal of information that cannot be contained in a filename and a tree structure. For example, notes on the details what data a table precisely contains is critical to have at the quick dispose of the user. These notes can be rather lengthy at times. To address this, CEMAP II has a notes feature for all of the documents that readily display the notes when the user is exploring the documents when in the process of making a document selection. These notes will, of course, need to be included in the document by the person creating or maintaining the documents. Optionally the user can append or make changes to the notes as well.

7.3 Component Fields

CEMAP II documents, profiles, tablesets, templates, and resources, often will have customization requirements within, or in the case of tablesets, templates and profiles will have a user-input requirement. The fields feature in these documents is an xml element that is uniquely special in that it can be present at any level of the xml document structure. The field element can be, and often is at least at the top level of the document. (The component notes feature described in the prior sub-section, is in fact operationalized as a component field.) At the top level, the fields feature serves the users best to always (but not required) to have a Notes field. The field contains from zero to many field elements. The field elements are very specific to their host parent element and can be used to contain information such as execution parameters, file locations, etc. When at the top level of the document, a Notes fields is used to describe the contents of the document in terms that a user will understand and appreciate. The value of the Notes field will be presented to the users in the interactive usage of CEMAP II when they are looking through the resourceLocator for a resource file. This can be an extremely helpful field to make use of.

7.4 Thesaurus Mapping

A common task for many situations is to want to change the value of a data item according to a mapping. For example, a node labeled, "Tom Thumb" and a node labeled "Thomas Thumb" in the data source, should be melded together as a single node, "Tom Thumb". This process of mapping words is a very common ask in AutoMap that is made available in CEMAP for users who desire to perform this matching before importing the source data into a network for ORA. The same format file(s) used in AutoMap can be utilized for this mapping process in CEMAP II.

7.5 Groupware

An important design feature of CEMAP II is the ability for analysts who work in groups to be able to share profiles, tablesets, and templates. This is accomplished by situating CEMAP II to be able to obtain any of its files from the Internet. This means that a profile for a work team, or entire organization, can be stored on a web server and the CEMAP II user, if permissioned, can see that profile in there resources profile and then load that profile into their CEMAP II session. This internet-enabled capability can be extended from http services to ftp, https, and even SOAP and RPC web services.

Along this same line of groupware, CEMAP II does allow for a soft link to a profile, tableset, or template that is stored on a web server. This means that a profile, for example, can link to a tableset file that is stored on a web server. Then should the tableset file be modified, the profile file may not need to change as the most recent changed tableset file will be loaded when the user profile document is loaded. This is opposed to embedding the tableset configuration in the profile document, which makes tablesets that are used by a group of users, or used many time by a single user, difficult to maintain across multiple profile documents.

7.6 Extendibility

The ability for CEMAP II to have unforeseen features added later is an important design goal and feature of the CEMAP II architecture. In means that in essence, any real-world data source is accessible to CEMAP II, and thus ORA and AutoMap, if the user can find any way to obtain the data electronically. If this primary condition is met, then a tableset can be created to integrate the source data into CEMAP.

A completely customized tableset can be written in java and called by CEMAP II: The class file is identified in the template element by: (A) identifying the physical location of the jar file in the httpJarFile attribute of the tableset element (this can be stored on the Internet or on a local disk, and(B) identifying the class and its java path in the classType attribute of the tableset element.

```
class testDummy {
      int count = 0;
      public void open() {
                  }
      public String [] getNext() {
            String [] row = {"a", "b", "c", "d", "e", "f", "g"};
            ++count;
            if (count > 10) return null;
            return row;
      }
      public void close() {
      }
}
```

Figure 6. testDummy.java code for a minimal customized tableset.

Figure 6 shows an example of the minimal amount of java code necessary to implement a customized tableset; CEMAP II does the rest. The second aspect of implementing this customization is the standard tableset xml which describes the actual table being created.

7.7 Migration Path from CEMAP I

The design of CEMAP II accommodates the original CEMAP (Frantz & Carley 2008) users by maintaining for a period of time an interface identical to that of the original CEMAP email parser. However, the processing underneath and hidden from view of the user utilizes the CEMAP II architecture as described in this document. A screen shot of the primary parameter input window (Fig. 7) is show below and is maintained in both ORA and AutoMap until such time that users have fully migrated to the CEMAP II tool. All of the features in the original CEMAP are available in the enhances CEMAP II, however, as current CEMAP users are migrating to the more powerful CEMAP II, it is eventual that the legacy CEMAP email parser will be removed from service in both ORA and AutoMap at a time to minimize any disruption to ORA and AutoMap users.

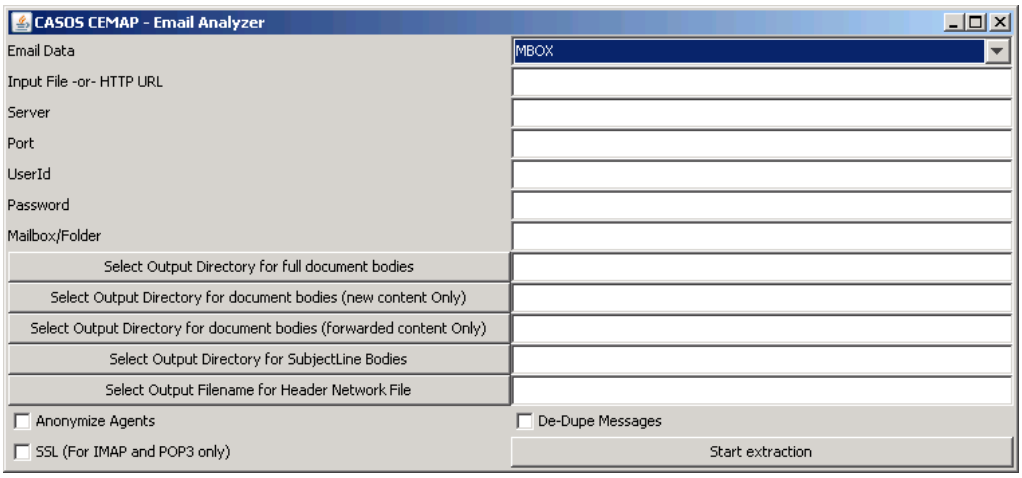

Figure 7. CEMAP I email parser user parameter-input window.

8 Workarounds

8.1 Microsoft Outlook Email Files (pst)

While CEMAP II is designed for extendibility, there are some situations that may be especially difficult to deal with, although there usually is a technical solution, but the solution may be outside the boundaries of the current CEMAP II feature-set. Microsoft Outlook email clients persist emails in a proprietary format (pst format) which makes it a awkward for software to interface with directly with these email files. Based on a search of non-Microsoft tools that apparently are able to access pst files, it seems that Microsoft does indeed make the format available, albeit not publicly, under commercial licensing agreement. There likely are other formats that all into this same category. Because of this political, non-technical, complication, CEMAP II may be unable to interface with data that is persisted in such proprietary formats, without future work.

Currently, to workaround this issue, it is necessary for users to first go into Microsoft Outlook to export their email data to a CSV file. CEMAP II has a tableset available for users to import this special CSV file, so the import process into ORA and AutoMap can continue unabated. While this straightforward process of exporting the email (in pst format) to CSV is not complicated or necessarily time-consuming, empirically users are having a difficult time with this step and certainly goes against the design goal of simplicity and ease-of-use for CEMAP II. Furthermore, this process depends on the user (or Microsoft) not changing the exact format of the exported CSV file in the future. While this workaround works from a technical perspective, it has its usability problems and has format change risk in the foreseeable future.

However, in this case a viable and complete technical solution may indeed be possible, but will require addressing some technical hurdles. The Microsoft Outlook files may indeed be made available to CEMAP II, thus ORA and AutoMap, by using Microsoft's Visual Basic platform to read the pst files and make the conversion to a industry standard format like MBOX (discussed earlier), which is readily readable by CEMAP II. Since Visual Basic and pst files are on the Microsoft platform, this is a

natural process for the two to work in harmony, thus the availability of this workaround. While it may be technical feasible for CEMAP II to spawn a parallel task to execute a Visual Basic program to make this conversion, it would require placing a Windows-only restriction on CEMAP II, thus also on ORA and AutoMap.

8.2 MySQL Java Library

Another technical situation that CEMAP II faces and will require a workaround, is that it is somewhat limited by its use of the mySQL java library. The mySQL library is not licensed to be distributed in the ORA and AutoMap installation packages, thus is also restricted for CEMAP II distribution. This library is a standard library used in Java programs to interface with SQL databases, such as Access, or Oracle. CEMAP II follows the standard practice of using this particular library for SQL interfacing as well. Given this, there is a difficulty that users will face when installing CEMAP II, ORA, or AutoMap, that being that the SQL library cannot be automatically installed with these programs. Instead, the user must install the mySQL library separately, which can cause some confusion to users. CEMAP II will perform the necessary software checking for this library if the mySQL library is necessary to the profile being executed. CEMAP II will utilize the standard function used in ORA for checking for the presence of this library (if needed) and will advise the user that they will need to load the library before performing the function. Possibly CEMAP II may be able to make this loading process a run-time feature so that the user needs not to exit and re-enter ORA or AutoMap to load the library; instead the library may be linkable dynamically.

9 Forward

This report provides an introduction to the architecture and specifications for CEMAP II, which facilitates the importing of real-world data into the CASOS software suite. There is a great deal of important operational detail not provided in this report, which will be provided in later reports that are published as the ideas in this report are actually coded and installed in the CEMAP II software. The features described in this report are considered rather all-encompassing and complete, particularly because the architecture allows for easy extendibility, so at the high level it is not expected that CEMAP II will evolve much more that as described herein.

10 References

Carley, Kathleen. (1991). A Theory of Group Stability. American sociological Review, 56, 331-354

Carley, Kathleen, Diesner, Jana, Reminga, Jeffrey, Tsvetovat, Max. (2004). Interoperability of Dynamic Network Analysis Software.

Carley, Kathleen & Reminga, Jeffrey. (2004). ORA: Organization Risk Analyzer. Carnegie Mellon University, School of Computer Science, Institute for Software Research International, Technical Report CMU-ISRI-04-106,

Davis, George B., Olson, Jamie, & Carley, Kathleen M. (2008). OraGIS and Loom: Spatial and Temporal Extensions to the ORA Analysis Platform. Carnegie Mellon

University, School of Computer Science, Institute for Software Research International, Technical Report CMU-ISR-08-121.

Diesner, Jana & Carley, Kathleen. (2004). AutoMap1.2 - Extract, analyze, represent, and compare mental models from texts. Carnegie Mellon University, School of Computer Science, Institute for Software Research International, Technical Report CMU-ISRI-04-100

Frantz, Terrill L. & Carley, Kathleen M. (2008). Transforming raw-email data into social-network information, In Christopher C. Yang, Hsinchun Chen, Michael Chau, Kuiyu Chang, Sheau-Dong Lang, Patrick S. Chen, Raymond Hsieh, Daniel Zeng, Fei-Yue Wang, Kathleen Carley, Wenji Mao, and Justin Zhan (Eds.) (2008). 'Intelligence and Security Informatics Workshops, PAISI, PACCF and SOCO 2008' Springer, Lecture Notes in Computer Science, No. 5075. Pacific Asia Workshop on Intelligence and Security Informatics (PAISI 2008). Grand Formosa Regent Hotel , Taipei, Taiwan, 17 June 2008.

Tsvetovat, Max & Reminga, Jeffrey & Carley, Kathleen. (2003). DyNetML: Interchange Format for Rich Social Network Data. NAACSOS Conference 2003, Day 2, Electronic Publication, Pittsburgh, PA.