AN ELECTRONIC MEDICAL RECORD FOR NEUROLOGY

Jeffrey L. Sponsler, MD, MS
Alaska Brain Center, LLC Wasilla Alaska 99654
Fei Pan, MS
134 W. Rincon Avenue, Apt. J Campbell, CA 95008-2857
fei.p.pan@gmail.com

ABSTRACT

Background: Literature review yields no neurology-specific electronic medical record (EMR) systems. *Hypothesis*: A prototype neurology EMR can be designed and rapidly prototyped using available technologies including database management. *Methods*: A neurology EMR prototype (NEMR) has been designed and developed using conventional tools (relational database, web browser language PHP). *Conclusions:* NEMR is in daily use and serves as the functional core of the clinic. Compared to paper based systems, the EMR allows instantaneous record access, automated report generation, electronic prescribing, and numerous other important clinical functions. Rapid prototyping with standard tools facilitated system development. Designing the NEMR specifically for neurology patients was key as the system if functionally integrated into neurology clinic methodology. Links to an artificial intelligence system for decision support has also been prototyped but exists as a research (non-production) subsystem.

KEY WORDS

Neurology, Electronic Medical Record, Database Management System, MYSQL, PHP, APACHE

1. Introduction

In this document, we describe an electronic medical record (EMR) designed for a neurology clinic practice. The EMR is called the Neurology Electronic Medical Record (NEMR). This system was designed and developed in 2004, has been the key data management facility for a busy clinic, and continues to be improved.

Literature review documents the benefits of electronic medical records. Malaviya reported that with a rheumatology specific EMR, more patients were seen, patient waiting lists were reduced, without compromising the quality of notes [1]. The conclusions included that physicians should work closely with designer/developers. Data retrieval has simplified clinical research with increased numbers of abstracts being presented and research papers being published.

2. Architecture

The database management system for the Neurology Electronic Medical Record (EMR) was created using MySQL [2], Apache [3], PHP [4], Java Scripts [5] with an Internet browser front-end [6]. The selection of these readily available systems made development quite straightforward and cost-efficient. Using an Internet browser user interface is economical insofar as each computer workstation with the Windows XP Professional operating system is equipped with Internet Explorer and therefore no user interface code was required except for PHP code and HTML code.

2.1 MYSQL Database Management System

A shareware relational database management system, **mysql**, is utilized to organize and store patient EMR data [2]. Data flow begins with database login, query for page specific data using conventional relational query language, incorporation of query result data into php code, interpretation of php code by the browser and finally presentation on screen.

2.2 Apache Monitoring Service

A shareware software package, Apache, is employed to function as web server for the database. Software development was not required, just a download and configuration of parameters. Apache links to PHP were set for web page communication and presentation.

2.3 PHP Web Page SQL Interface

Data presentation pages are constructed using PHP code. PHP code is imbedded in hypertext markup language (HTML) code, interpreted dynamically, and the resulting results are integrated into the HTML code. The resulting data are displayed in a web browser, Internet Explorer. Adopting object-oriented

(OO) techniques in the PHP applications allows greater flexibility and easier maintenance for the application. Programs become easier to extend, or debug, and code sharing and reuse becomes simplified. There are quite a few new PHP frameworks developed based on the OO model. *Zend Framework* [6], developed by the creator of PHP, is one of them. Our team developed a light-weight framework that follow the MVC (Model-View-Control) model. It provides a consistent implementation model for all the sub-modules, simplifies the interaction between PHP scripts and the database. Developers only need to focus on the logic of the business module, and complete the coding with less code than traditional PHP coding, with greater flexibility and scalability.

JavaScript. We used Java Script in our HTML page. And we used jQuery framework [7] (an open source Java Script framework, for AJAX functions and some special effects (like calendar pop-up, type auto-completion, etc).

3. Data Organization

The database management organization is described in this section. Requirements for the NDMS were created by the neurologist/software engineer on the project based on standard techniques of office organization and patient record charting.

3.1 Background Information

3.1.1 History

This history data is entered by keyboard into a text box. These data include the medical symptoms that brought the patient to the clinic (for example, headache). Future improvements include using a voice to text system to reduce or eliminate typing. The history is considered important initial data in the patient workup. Often lengthy, the History data is augmented in our system by menu oriented history pages, History By Diagnosis, described below.

3.1.2 Previous Medical History

The previous medical history is a list of common maladies that can be selected by clicking on a checkbox. There is a comment field for extra text and a date field for the date of the medical diagnosis. For example: “Diabetes 1995”.

3.1.3 Previous Surgical History

The previous surgical history is a list of surgical procedures presented in a menu with check-boxes and data fields. For example, “Appendectomy 1975”.

3.1.4 Allergies

The Allergies page includes a list of medicine adverse reactions. The medicine names are typed in as text. The adverse reaction is selected from a menu and a text field is provided for date of allergic reaction. For example, “Penicillin Rash 2001”.

3.1.5 Family History

A simple table that is filled out for the patient, the Family History table includes class of relative (“Sister”), age (“45”), status (“Living”), and disorders that are selected from a menu (“Stroke”).

3.1.6 Personal History

This page contains mechanisms for marital status, smoking status, alcohol use, education level, profession, number of children.

3.1.7 Radiology Page

The Radiology Page contains multiple entries. Each entry contains these elements: date of study, study type (“MRI”), study anatomy (“cervical spine”), results text, and impressions text. Additional fields include notes and print flag (for report writers). Text from radiology reports are available as characters to be copied and pasted or via optical character recognition (OCR).

3.1.8 Laboratory Data

Laboratory data enters the clinic is various formats including faxes, PDF files, and Health Level 7 (HL7, ref) formatted files. Database tables have been constructed to adapt to the various formats. Once added to the database, the information can be presented on-screen or be included into clinic reports.

3.1.8.1 Cerebrospinal Fluid Data

The neurology clinic will frequently study patient cerebrospinal fluid. A data table has been constructed to store the attributes for this lab set. The table includes date, fluid color, fluid clarity, anesthesia (“lidocaine”), fluid volume, white blood cell count, red blood cell count, Gram stain, myelin basic protein, oligoclonal bands, IgG index, IgG synthesis rate, herpes simplex virus, cytology, and other fields.

3.1.8.2 Standard Lab Data

A table called Standard Labs is included in the database. This table includes laboratory names with reference ranges (high, low values), and lab units. An interface page allows dynamic alterations to this table. For example, a user can add the lab "SODIUM", with units "mg/dL", low "130", high "140".

A second table Standard Lab Groups has been created in the database. An interface page allows dynamical alterations to this table. For example, a user can add the standard lab group "COMPREHENSIVE METABOLIC PANEL" and then add entries selected from Standard Labs (for example "SODIUM" into that new group. The relationship of group to lab is *many-to-many*.

A third table Patient Standard Labs includes the actual laboratory data for a specific patient. The lab data entry process begins with selecting a Standard Lab Group (such as COMPREHENSIVE METABOLIC PANEL). All lab components of that group (such as SODIUM) will be displayed in a table and the actual data value can be entered (e.g. 135). A Submit button commands that the data are stored into the table Patient Standard Labs. Presentation of data on screen or in a report is organized by group (only if data is stored for that patient).

3.1.8.3 HL7 Lab File Interface

The Health Level 7 (HL7) specification has been incorporated into our database system. The data is organized according to that specification and is stored in tables. Please refer to HL7 references [8] for details of this interface. Patient report generators print HL7 data in an organized manner; lab results are incorporated into reports.

3.2 Visit Information

The clinic is structured around individual patient visits. A given patient may visit the clinic once, twice, or many times. The first visit is labor-intensive due to the cost in staff-time of entering data into the tables for the patient. This cost is relatively high and if patients were seen in clinic only once, this cost might be prohibitive. The residual visit 1 data for a patient facilitates subsequent visits because most of the data stored in the first visit can be accessed and in some cases copied from previous visits. Some global data (address, telephone, family history, radiology entries, drug allergies) is not stored with visit keys. Visits are stored in a Visits database table with keys of patient ID and visit ID (an integer). Visit date is stored in each entry.

3.2.1 Subjective Data

The subjective data can be entered into a text box. This is usually an updated free text narrative. For example, "the drug that was prescribed caused double vision." A voice to text system can be employed to generate this text. A system such as this is not currently integrated.

3.2.2 History By Diagnosis

A NEMR page called "History By Diagnosis" contains a menu of specific diagnoses (e.g. Seizure). Generally, a neurology clinic patient has one major diagnostic issue that brings the patient to the office. An epilepsy patient has a chronic neurological illness and will visit the clinic multiple times for follow-up. Organizing the history data by diagnosis therefore improves history gathering efficiency. This organizational style is rigid and a brief free text narrative history must usually be obtained also.

Selecting a diagnosis opens a page with a list of menus. Each menu has a header that is the heading part of a sentence (e.g. "The location of the headache pain is"). The menu with checkboxes includes possible items that can complete the sentence (e.g. "left frontal", or "right temporal"). One of the clinic's staff will work with the patient to complete each menu. The menus are colored coded by priority so that high priority questions are asked first. The menus are so detailed that completing a set of menus is rarely possible in a typical clinical setting.

During the report writing stage of the patient process, the information entered in the History By Diagnosis menus will be appended, with commas and other punctuation inserted, thereby producing a complete sentence. For example, "The location of the headache pain is left frontal and right temporal." The constructed sentence is added to the report. The artificial nature of these menus does produce sentences that are at times somewhat awkward but the time efficiency is very high with this instrument insofar as detailed patient information is created as text rapidly with mouse/menu technology. This system increases the correctness of spelling and grammar compared to voice-to-text systems (we propose).

3.2.3 Medication spelling support

Several pages support manual entry of drug names. For these tables, spelling support and drug name completion is supported. For example, entering “DI” yields a menu with Dilantin and Digoxin. The user may continue typing or may click on a menu entry.

3.2.4 Generic Medications

A dynamic table for medications, Generic Medications, has been created. The table relations include Drug Name, Class, Contraindications, Adverse Effects, Dose, Number, Route, Frequency, and PRN. The first step in using the associated user interface is clicking on the command ADD ENTRY which adds a new row to the medications table. Drug Name is a text entry gadget with a dynamic spelling helper. As text is typed a menu appears with possible drug names. For example, entering “DI” yields a menu with Dilantin and Digoxin. The user may continue typing or may click on a menu entry. Dose is a free text gadget and an example would be “25 mg”. Number relation is free text and is an integer number of doses. Route is selected from a popup menu containing elements such as “PO”, “IV”, “TOPICAL” etc. Frequency is selected from a popup menu containing elements such as “DAILY”, “Q2 HOURS”, “TID”, etc (the list contains standard pharmacy shorthand codes). The PRN (“as needed”) field is populated by a popup menu containing standard symptoms for “as needed” medications such as “PAIN”, “DYSPNEA”, “HEADACHE”, etc. This table is initially populated by an external file; new entries can be made by an editing page.

3.2.5 Trade Medications

A table Trade Medications has been created. This table contains non-generic drug entries each of which points back (by unique ID) to a parent generic medication. This table is initially populated by an external file; new entries can be made by an editing page.

3.2.6 Formulations Table

The third row of the drug hierarchy is the Formulations Table. Each entry here is semantically designed to represent a prescription. Each entry has a parent Trade Medication and a grandparent Generic Medication. The electronic prescription webpage is generated using this table and selections from this table are the first step in e-prescribing.

3.2.7 Drug Interactions Table

The Drug Interactions Table provides a dynamic extensible instrument to store entries. Each interaction entry contains Drug1 (a trade or generic drug id), Drug2 (another id), Combination Effect (such as increased sedation, or hypertension), Metabolic Effect (of Drug1 on Drug2), Notes. The purpose of each entry is to encode drug interactions. If the Metabolic Effect is present then any changes that Drug1 will make on Drug2. If there is a combined effect then that field has a value and examples include increase levels, decreased levels, sedation, serotonin syndrome, and hypertension.

3.2.8 Electronic Prescriptions

A database table ERX consists of a number of fields that support e-prescribing. The fields are Drug ID, Drug Dose, Route, Frequency, PRN, Count, Refills. The algorithm for the user is as follows: From the Formulations page, the user selects medications/formulations to prescribe. A command COPY TO ERX moves the selections to the ERX table. The user then can move to the ERX Page (which is date stamped). A command WRITE RX will cause the medications listed to be written to the prescription which can be printed, signed, and faxed. Alternatively, the prescription (which has an authenticated signature) file can be sent by other electronic means to the pharmacies (that have secure email for example).

Drug Name is a text entry gadget with a dynamic spelling helper. As text is typed a menu appears with possible drug names. For example, entering “DI” yields a menu with Dilantin and Digoxin. The user may continue typing or may click on a menu entry.

3.2.9 Standard Medications

A dynamic table for medications, Medications, has been created. The table relations include Drug Name, Dose, Number, Route, Frequency, and PRN. The first step in using the associated user interface is clicking on the command ADD ENTRY which adds a new row to the medications table. Drug Name is a text entry gadget with a dynamic spelling helper. As text is typed a menu appears with possible drug names. For example, entering “DI” yields a menu with Dilantin and Digoxin. The user may continue typing or may click on a menu entry. Dose is a free text gadget and an example would be “25 mg”. Number relation is free text and is an integer number of doses. Route is selected from a popup menu containing elements such as “PO”, “IV”, “TOPICAL” etc. Frequency is selected from a popup menu containing elements such as “DAILY”, “Q2 HOURS”, “TID”, etc (the list contains standard pharmacy shorthand codes). The PRN field is populated by a popup menu containing standard symptoms for “as needed” medications such as “PAIN”, “DYSPNEA”, “HEADACHE”, etc.

3.2.10 Examination Table

The examination table includes relations Visit (a key), Patient ID (a key), and examination codes. An array of exam groups is stored in a PHP file. Each group defines elements of an examination for that group. An example group is “BICEPS STRENGTH” and group elements include GRADE 1, GRADE 2, GRADE 3, GRADE 4, and GRADE 5. Each group element is associated with a unique code (e.g. “BICEPS_GRADE_1”) and a text string “biceps strength is grade 1”). The user interface presents this group with the heading BICEPS STRENGTH and the elements as a list of radio check boxes. A user selected a menu item and the associated code is stored. When the examination data is presented later on screen, each code maps into an array of examination elements (thereby mapping from code to English text). This same logic maps from stored codes to English text in the report writers.

The examination menus include the standard physical and neurological examination components [9]. Cranial nerve exam, mental status, strength, reflexes, language, speech, sensory, and gait are included.

There is a set of “normal” neurologic examination menus that can be marked with the mouse. For example there is a menu item “Reflexes were symmetrical and grade 2+ in the four extremities.”

3.2.11 Diagnosis Code Management Page

A separate page, DX Code Management presents the coding groups (such as Epilepsy) at top level. Selecting a group causes page refresh with the actual diagnosis codes under that group (Complex Partial Seizures, Seizures Not Otherwise Specified, Partial Seizures). The DX Code Management page allows users rapid access to all codes. The codes in this page can either be added from a text definition file or via editing commands in this page. The codes in this page are stored into an inverted index table, DX Code Index. This index is used by the diagnosis code lookup subsystem (described below).

3.2.12 Assessment Page

The NCDS includes a database table Assessment that stores entries detailing the interpretation of the history, physical, lab, and radiology data. The entries represent data synthesis and are the diagnostic conclusions for the patient. The relations include Diagnosis (e.g., Stroke), Notes (“Patient has evidence suggesting a right middle cerebral artery CVA”), and Plan (“We plan a cerebral angiogram.”)

The assessment page incorporates a diagnostic code retrieval command. The Diagnosis text (which is typed, “Seizure”) is used as a key into the diagnosis code index table. A popup gadget presents a menu of diagnosis matches (“340.50 Partial Seizure”, “345.40 Complex Partial Seizure”, “345.41 Complex Partial Seizure Intractable”) and the user selects by mouse the appropriate code and this selected code is then stored in the DX_CODE attribute for this Assessment entry in the database. The stored code is then available for reports.

3.3 Standard Prescriptions Page

To support electronic prescribing, a table called Standard Prescriptions is defined. The relations include Identifier (“Neurontin Titration”), Medicine Name (“Neurontin”), Generic Id (“gabapentin”), Dosage (“300 mg”), Sig (“One Tab PO BID”), Dispense (“Dispense 60”), and Refills (“Two Refills”). This table can be modified by users to construct a library of commonly used prescriptions. For a given patient the Standard Prescriptions page is displayed with the drug library listed alphabetically. To select a standard prescription, the user puts a check with the mouse on the entry. A SUBMIT button stores the selection in the patient’s standard prescription table. A VIEW PRESCRIPTION button prints the prescription. Future software revisions may require the library to have an alphabetic index due to the growing list of prescriptions in the database.

3.3.1 Treatment Plan Page

The table called “Treatment” is structured to support the treatment plans for the patient. Typically treatments include starting medications, stopping medications, obtaining external medical records, physical therapy, occupational therapy, and so on. The relations for this table include Treatment and Notes. The Treatment field is population via a menu containing a list that includes “Start Rx”, “Stop Rx”, “Continue Rx”, “Continue Aspirin”, “Continue Plavix”, and other common plan statements. The list of plan statements is defined in a include file and coded using PHP.

3.3.2 Follow-Up Page

The data table RTC stores entries that describe the clinic follow-up plan. The relations include RTC Id and Notes. RTC ID is populated by a popup menu of common follow-up plan statements. Examples include “RETURN TO CLINIC AS NEEDED”, “RETURN TO CLINIC AFTER TESTS”, “RETURN TO CLINIC IN 2 MONTHS”.

3.3.3 Procedure Code Page

The Procedure Code Page supports selection of the procedure code. Example codes include 99205 (for new patient with high complexity) or 99215 (return patient is high complexity). Other codes pertinent to the clinic includes 95819 (EEG Technical and Professional) or 95860 (EMG One Limb). The command COPY VISIT does not copy the procedure code selections to the new visit; this is because a new visit to the clinic will involve different procedures, generally.

3.3.4 Abnormals Processing

A database page to support processing of abnormal patient lab data has been created and is actively used. When lab data enters the clinic (by fax or by HL7 electronic interface), a staff member will review and, if abnormal, will copy the lab data to a new *abnormal* entry for the patient. This abnormal result is tagged with date and patient identifier. A report generator, **Abnormal Lab Report**, lists for the clinician all abnormal lab results (over a date range) for all patients.

```
Patient: John Doe
Last Visit: 2011-03-31
Lab Value: Vitamin D 22.2L 30.0-100.0,
Interpretation: Low vitamin level.
Action: START VIT D 1000 UNITS DAILY
Appt Date: 2011-11-30
```

The clinician can make notes and create new test orders for the abnormal entry. These notes and orders are processed by staff periodically. This ensures that abnormal lab (or imaging) results are processed, patients are informed, and further testing is done as needed.

3.3.5 References Page

A text only page, References Page, is supported. This page supports multiple editable sub-pages. Each sub-page contains attributes: Title (the title of the reference), Author (the author of the reference), and Text (the text of the reference material). This page has been used to store pertinent research articles or abstracts for the patient which are often obtained during the clinic visit. The data in the References Page are included in the automatically generative History Report.

3.4 Report Generators

The database contains a set of report generators that incorporate data into specific standard reports.

3.4.1 History Report

The History Report generator pulls out text and data from the database and generates a formatted conventional history and physical report (reference). The generator creates a new file in HTML format for the report. The authenticated signature for the report is the login of the current user. For hard copy of the report, the HTML file can be pasted into a word processor program for printing.

3.4.2 Progress Report

The Progress Report generator pulls out text and data from the database and generates a formatted conventional progress report (reference). The generator creates a new file in HTML format for the report. The authenticated signature for the report is the login of the current user. For hard copy of the report, the HTML file can be pasted into a word processor program for printing.

3.4.3 Prescription Writer

The Prescription Report generator pulls out text and data from the database and generates a formatted prescription. The generator creates a new file for the report and for historical record. The prescription is printed, signed, and then faxed to pharmacy.

3.4.4 Patient Plan Report

A brief report, the Patient Plan Report, is generated at the close of each patient visit. This report includes vital signs, medication list, flagged radiology reports, flagged lab reports, diagnosis list, test plan, lab orders, treatment plan, and follow-up plan. This report provides the patient with a snapshot of their visit and can later be reviewed by the patient and can be stored in home records.

3.4.5 Lisp Report

A comprehensive data file, the Lisp Report, is generated by command. This report includes virtually all data for all visits in a lisp function format (*function arg1 arg2 arg3*). These reports are loaded into the research environment for expert system experimentation [13]. Important data that is transferred includes patient profile, patient examination data, and imaging data. Strict patient confidentiality is maintained in the research environment.

4. External Patient Data Entry

A second database management system has been created to support patient data entry outside of the clinic. This database contains a subset of the database and this subset includes chief complaint, medication list, previous medical history, previous surgical history, medical allergies, demographic data, and insurance data. The method of use includes the following: First, the patient is schedule for an appointment. Next, the patient is given a username and password for the external database. The patient can then enter data via menus and text gadgets. When the patient comes to the appointment, the external database information is downloaded to the clinical database. The external records are then deleted to minimize violation of confidentially.

5. Benefits of electronic medical record

Benefits in the clinic of an EMR include nearly instantaneous access to patient data, electronic prescribing, automated download of lab data via HL7 (with incorporation into reports), automated report generation, automated diagnosis and procedure code lookup.

6. Decision Support Subsystems: StrokeDx and ENCAS

In a separate effort, a program in our clinic that has been developed (as a prototype) for expert decision support. This system, StrokeDx [13], processes patient data and produces diagnostics for ischemic stroke. This prototype is not in production but is an on-going research effort. StrokeDx was recently augmented to include knowledge of a rare genetic stroke disorder, CADASIL, and this has reported [4]. Another program, the EMG and Nerve Conduction Analysis System (ENCAS) has been developed using Lisp and is daily use for automated expert interpretation of EMG data [15].

7. HIPPA Compliance

Our clinic and EMR users follow the guidelines of the Health Insurance Portability and Accountability Act (HIPPA). The NDMBS is situated on a secure server computer (Windows XP Professional) with firewall protection. Users are required to use passwords to access the database. All clinic staff have completed HIPPA training as required.

8. Conclusions

The Neurology Clinic Database/Knowledge Base System has been created in a dynamic fashion. We make the following conclusions:

1. Fielding an evolving prototype has advantages. Rapid changes to the software allow important changes to be made in a short time. Freeware systems support cost efficiency.
2. Fielding an evolving prototype has disadvantages. There is an inherent danger to fielding a prototype in a production environment; a non-functioning EMR negatively impacts the clinic.
3. Random access to patient data via web browser is efficient. Web browsers are ubiquitous and obviate the need to build a dedicated front-end to the EMR. Data access

via an EMR is hundreds (perhaps thousands) of times faster with an EMR compared to paper records.

4. Automatic electronic prescription writing and standard prescriptions have increased accuracy and time efficiency and this has also been described (CMS 2011). In the US, e-prescribing is encouraged by the Centers for Medicare and Medicaid (CMS 2011).
5. Automatic report generation produces a major efficiency improvement (compared to dictation). Patient reports are often nearly complete immediately after the clinic visit.
6. Integrated diagnostic code lookup facilitates coding process.
7. Engineering an EMR that is specific to the field of neurology is beneficial for this specialty clinic by supporting neurology focus on history, examination, and test orders.
8. Integrated diagnostic code lookup facilitates coding process.
9. Engineering an EMR that is specific to the field of neurology is beneficial for this specialty clinic by supporting neurology focus on history, examination, and test orders.

9. References

[1] A. Malaviya, S. Gogia, Development, implementation and benefits of a rheumatology-specific electronic medical record application with automated display of outcome measures. *Int J Rheum Dis.* October 13, 2010 (4), 347-60.

[2] MySQL 5.6 Reference Manual http://dev.mysql.com/doc/refman/5.6/en/.

[3] APACHE Reference http://httpd.apache.org/docs/2.3/.

[4] PHP Documentation Group, 2011. http://www.php.net/manual/en/copyright.php.

[5] JAVASCRIPT Reference http://www.javascripter.net/faq/javascr4.htm.

[6] Zend Framework Reference http://en.wikipedia.org/wiki/Zend_Framework.

[7] Resig, John (2009-01-14). "jQuery 1.3 and the jQuery Foundation". *jQuery Blog*, 2009-05-04.

[8] Laboratory interface specification for HL, 2011. http://www.hl7.org/

[9] W. Bradley, C Daroff, G. Fenichel, C. Marsden, *Neurology in Clinical Practice*, *3rd Edit*ion. Butterwork-Heinemann, Massachusetts 2000; pp 5-8.

[10] S. Keene, *Object-Oriented Programming in Common Lisp: A Programmer's Guide to CLOS*, Addison Wesley, 1989, ISBN: 0201175894.

[11] J. L. Sponsler, F. VanScoy, J. Culberson, 2004. The System for Neurological Analysis of Patient Symptoms: An Integrated Artificial Intelligence Prototype. *Proceedings of the IASTED International Conference on Modelling, Simulation and Optimization (MSO2004)*, August 17 -19, 2004, in Kauai, Hawaii.

[12] Centers for Medicare and Medicaid. http://www.cms.gov/EPrescribing/ "E-Prescribing" 2011.

[13] J. Sponsler. StrokeDx: An expert system to diagnose stroke, *Proceedings of the IASTED International Conference on Telehealth*, Innsbruck Austria, February 2012.

[14] J. Sponsler. Stroke Expert System Extended to Diagnose CADASIL, *Proceedings of the IASTED International Conference on Telehealth*, Innsbruck Austria, February 2012.

[15] J. Sponsler. Automated Analysis of Electromyography Data, *Proceedings of the IASTED International Conference on Telehealth*, Innsbruck Austria, February 2012.