

Drahtlose Erfassung GCP-konformer klinisch-epidemiologischer Studien mit mobilen Endgeräten

Michael Wallner, Torsten Illmann, Jan Suchanek, Michael Weber

**Abteilung Medieninformatik
Universität Ulm
Albert-Einstein-Allee 11
89069 Ulm**

michael.wallner@informatik.uni-ulm.de
torsten.illmann@informatik.uni-ulm.de
jan.suchanek@informatik.uni-ulm.de
weber@informatik.uni-ulm.de

Abstract: Im Kompetenznetz "Ambulant erworbene Pneumonie" (CAPNetz) werden epidemiologische Daten von Patienten mit ambulant erworbener Pneumonie (CAP) deutschlandweit erfasst. Die klinisch-epidemiologische Studie wird vollständig übers World Wide Web definiert, erfasst und ausgewertet (Remote Data Entry, RDE). Ziel des Projektes ist es einerseits papierbasierte Erhebungen für derartige Studien abzulösen und Daten nur einmal und direkt beim Patientenkontakt zu erfassen. Andererseits werden verschiedenste Erfassungsgeräte (unterschiedliche Laptops, Webpads und PDAs) eingesetzt und auf Ihre Tauglichkeit – sowohl technisch als auch anwendungsorientiert - evaluiert. Die Erfassung geschieht in acht deutschlandweit verteilten Erfassungszentren von speziell geschulten Ärzten. Diesbezüglich werden Funknetze in acht Krankenhäusern und Universitätskliniken unter Berücksichtigung nötiger und speziell angepasster Sicherheitsvorkehrungen installiert. Zur Gewährleistung des Patientendatenschutzes ist ein umfangreiches Datenschutzkonzept erarbeitet worden, welches mit den Landesdatenschutzämtern abgestimmt ist.

Dieser Artikel hebt die Probleme und Anforderungen an die elektronische Erfassung von klinisch-epidemiologischen Studien und deren Erfassung mit mobilen Endgeräten hervor. Es wird zwischen Online- und Offline-Erfassung unterschieden und sinnvolle Einsatzszenarien dafür genannt. Die entwickelten Konzepte beschränkten sich nicht auf die Erfassung der "Ambulant erworbenen Pneumonie", sondern sind allgemein für klinisch-epidemiologische Studien einsetzbar. Bei allen Konzepten stehen neben der allgemeinen Verwendbarkeit die Datensicherheit und die Gewährleistung des Datenschutzes im Vordergrund. Dies ist jedoch nicht Schwerpunkt dieses Artikels.

Probleme und Anforderungen

Mobile Erfassungsgeräte wie z.B. Laptops, Webpads und PDAs sind in punkto Betriebssystem, Bildschirmauflösung und Speicherverbrauch i.a. sehr unterschiedlich. Aufgrund dessen wird ein adaptives Modell benötigt, das die Struktur, Inhalte, mögliche Interaktions- und Darstellungsarten von elektronisch zu erhebenden Studien logisch trennt. Das zu entwickelnde System muss diese Bestandteile in Form einer Schnittstelle zur Verfügung stellen, um angepasste Darstellungen und Interaktionsmöglichkeiten auf verschiedenen Geräten zu ermöglichen. Um klinische Studien GCP-konform verwalten zu können werden im Folgenden die Anforderungen an elektronische Erhebungsbögen (Case Report Forms, CRFs) analysiert. Ein CRF besteht im Wesentlichen aus einer Menge beliebiger Texte und Interaktionselementen, in welche Dateninhalte eingegeben werden sollen. Diese können beliebig gruppiert und beliebig tief geschachtelt werden. Plausibilitäten werden auf unterschiedlichen Ebenen (z.B. für ein Attribut, ein CRF oder einer Studie) und unterschiedlicher Wichtigkeit benötigt. CRFs können darüber hinaus mehrfache Multiplizitäten verlangen und zwischen ihnen müssen parameterisierte Relationen definiert werden können (z.B. ist-abhängig-von, besteht-aus-n). Durch die Protokollierung einer Historie (Wert, Datum, Benutzer) von Datenänderungen verknüpft mit der Möglichkeit, Kommentare unterschiedlicher Gewichtung mit anzugeben, können Audits durchgeführt und sog. severe Events verarbeitet werden.

Zusätzlich zur Online-Eingabemöglichkeit ist die Möglichkeit einer Offline-Erfassung sinnvoll, um Daten auch außerhalb der funkvernetzten Bereiche erfassen zu können und diese zu einem späteren Zeitpunkt mit den bereits gespeicherten Daten zu synchronisieren. Neben einer bidirektionalen Synchronisation von Inhaltsdaten von CRFs ist mindestens eine unidirektionale Synchronisation der Strukturinformation von CRFs inklusive Relationen wünschenswert, um auch strukturelle Änderungen automatisch zu propagieren. Bei gleichzeitiger Erfassung eines Falls von zwei Personen, können dabei Konsistenzprobleme auftreten. Diese können automatisch als severe Events oder manuell durch den letzten Erfasser aufgelöst werden.

Die Datenein- und -ausgabe stellt an normale Personalcomputer leicht zu erfüllende technische Anforderungen. Der Rechner benötigt lediglich einen Zugang zum Internet und einen Web-Browser, der JavaScript und die Verschlüsselung über SSL ermöglicht. Laptops und Webpads [FP02][HW02], die über eine grafische Auflösung von mindestens 800x600 Bildpunkten verfügen, können direkt über web-basierte Schnittstellen an der Datenerfassung teilnehmen; anders stellt sich jedoch die Situation dar, wenn Subnotebooks oder gar PDAs eingesetzt werden sollen. Hier reicht die grafische Auflösung i.d.R. nicht aus, um die elektronischen Erhebungsbögen sinnvoll in Webbrowsers darzustellen. Es wird notwendig, die dargestellten Informationen geeignet auf das jeweilige Eingabegerät anzupassen. Die Erzeugung des Layouts und die Benutzerführung der Erfassungsoberfläche muss im Falle von Kleinstgeräten einer separaten Clientapplikation obliegen, um das zu übertragene Datenvolumen und die interne Verarbeitung möglichst gering halten zu können. Für größere Erfassungsgeräte kann diese Information z.B. in Form eines XSL-Stylesheets übermittelt werden.

Da in klinischen Studien mit Patientendaten umgegangen wird, ist Sicherheit der Daten ein wichtiges Thema. Um die Persönlichkeitsrechte der Patienten nicht zu verletzen muss in Absprachen mit den Datenschutzämtern in Deutschland ein Datenschutzkonzept erarbeitet und verabschiedet werden. Auch auf der Seite der Informationstechnik gibt es zahlreiche Aspekte, die zu berücksichtigen sind. Diese müssen in das Datenschutzkonzept einfließen.

Bisherige Entwicklungen und Ergebnisse

Als Grundlage für die Datenerhebung dient serverseitig ein eigens in PHP entwickeltes, komponentenbasiertes Web Content Management System (WCMS), welches speziell für obig beschriebene Bedürfnisse erstellt ist. Es können hierin sehr einfach einzelne PHP-basierte Software-Komponenten mit folgenden Eigenschaften definiert werden:

- Eine Komponente ist ein kleiner sichtbarer oder unsichtbarer Bestandteil einer Webseite
- Komponenten haben eigene Read/Write-Zugriffemodi, realisiert durch ACLs (Access Control Lists).
- Komponenten können mehrere Ansichten haben (inhaltlich, strukturell, präsentativ, etc).
- Komponenten können über sog. Eigenschaftseditoren webbasiert bearbeitet und zusammengesetzt werden.

Das WCMS bildet eine zentrale Serverplattform und kann unter Nutzung dieser Komponentenstruktur dazu verwendet werden, sowohl den Internetauftritt des Projektes zu erstellen und zu verwalten, als auch die Dateneingabe und Speicherung der Studiendaten abzuwickeln. Es stellt eine detaillierte Benutzer- und Gruppenverwaltung zur Verfügung, welche die Grundlage für die Zugriffskontrolllisten (ACLs) bildet, mit denen die Zugriffskontrolle geregelt wird. Eine ACL ist realisiert als eine Menge von Benutzern, die Mitglieder der ACL sind. Diese Menge setzt sich zusammen aus explizit eingeschlossenen Benutzern, Benutzern, die Mitglieder in eingeschlossenen Gruppen sind und explizit ausgeschlossenen einzelnen Benutzern. Da alle Komponenten durch diese ACLs geschützt werden, ist es möglich, für jeden Benutzer einen individuellen Funktionsumfang zu definieren.

Für den Internetauftritt des Projektes stehen mehrere Komponenten zur Verfügung, etwa Layout-Strukturierungselemente, Text-, Bild- und andere multimediale Objekte oder inhaltliche Strukturierungselemente wie Menüs. Auf den allgemeinen Internetauftritt soll jedoch an dieser Stelle nicht weiter eingegangen werden.

```

<DATE key='birth'
      descr='Geburtsdatum' />

<NUMBER key='height'
        descr='Größe'
        format='3.0'
        unit='cm' />

```

Abbildung 1: Schematische XML Ansicht

```

<VALUE key='birth'>
19730420
</VALUE>

<VALUE key='height'>
182
</VALUE>

```

Abbildung 2: Wertansicht in XML

Geburtsdatum:	<input type="text" value="20"/>	<input type="text" value="04"/>	<input type="text" value="1973"/>
Größe:	<input type="text" value="182"/>	cm	
Gewicht:	<input type="text"/>	kg	

Abbildung 3: „HTML“ Repräsentation

Geburtsdatum:	<u>20</u> . <u>04</u> . <u>1973</u>
Größe:	<u>182</u> cm

Abbildung 4: „Text“ Repräsentation

Für die Datenerfassung wird als clientseitige Erfassungssoftware im webbasierten Fall ein gewöhnlicher Webbrowser und im Offline-Fall eine Clientapplikation verwendet, welche speziell auf die mobilen Erfassungsgeräte zugeschnitten ist. Die Clientapplikation selbst ist, wie das serverseitige WCMS, unabhängig von der jeweiligen Studie. Die Anforderungen an GCP-konforme Studien sind über einzelne WCMS-Komponenten umgesetzt. So wurde beispielsweise für jede mögliche logische Eingabemöglichkeit eine eigene Interaktionskomponente entwickelt. Diese stellt sich in einer gewöhnlichen HTML-Ansicht als HTML-Formularelement dar. Zusätzlich ist diese versehen mit etwaigen Plausibilitäten, welche wenn möglich bereits clientseitig über Javascript abgefragt werden. Die restlichen Plausibilitäten werden serverseitig ausgewertet. Weiterhin sind WCMS-Komponenten für die Kombination von Interaktionselementen, einen vollständigen CRF und letztlich die komplette Studie definiert. Die Eingabekomponenten tragen auch dafür Sorge, Erfasser, Eingabezeitpunkt und alle Änderungen an bereits eingegebenen Daten serverseitig zu protokollieren. Daher kann jederzeit nachvollzogen werden, welcher Wert eines CRFs wann durch wen eingetragen oder geändert wurde.

Als Darstellungsformen existieren momentan neben der HTML-Ansicht, eine Editieransicht zum Verändern der CRF-Struktur, eine XML-Ansicht als Austauschformat für Daten und Struktur der CRFs und eine proprietäre Textversion, welche im Gegensatz zu XML wesentlich einfacher zu parsen und weiterzuverarbeiten ist. Daher ist diese besonders für Clientapplikationen auf Kleinstgeräten wie PDAs zur Offline-Eingabe geeignet. Die Darstellung meist erheblich langer Erhebungsbögen (i.d.R. über 50 Elemente) auf kleinen Displays erfordert eine automatisierte, bedienerfreundliche und übersichtliche Unterteilung der CRFs in schlanke, einfach navigierbare Unterformulare. Hierzu werden Gruppierungen und Abhängigkeiten (Plausibilitäten) ausgenutzt, um die Aufteilung und Sichtbarkeit einzuschränken.

Besondere Aufmerksamkeit wurde auf die strikte Trennung nicht nur von (wie üblich) Struktur und Layout, sondern von Struktur, Layout und Inhalt gelegt. Dieses soll anhand der Abbildungen eins bis vier verdeutlicht werden. Die Abbildungen zeigen (in einer stark vereinfachten Form) immer die gleichen beiden Komponenten, allerdings in unterschiedlichen Repräsentationen bzw. Ansichten. Abgebildet sind beispielhaft zwei Eingabefelder eines CRFs, Geburtsdatum und Größe. In der Abbildung eins sind diese Eingabefelder in der Strukturansicht dargestellt, die aber keine Werte (also keinen Inhalt) enthält. Diese Ansicht dient der Clientapplikation dazu, das Layout des CRFs den technischen Möglichkeiten des Eingabegerätes entsprechend bestmöglich zu generieren. Auch können Änderungen am Schema des CRFs serverseitig durchgeführt werden, welche die Clientapplikation getrennt von ihren Daten mit dem Server synchronisieren kann. Die Abbildung zwei hingegen zeigt eine reine Werteansicht der Eingabefelder, aus der nichts über deren Struktur hervorgeht. Die Clientapplikation benutzt diese Ansicht, um die erfassten Daten von und zum Server zu transportieren. Die Kommunikation mit dem Server geschieht sowohl in der strukturellen als auch in der wertebasierten Ansicht über (verschlüsseltes) HTTP-Protokoll. Die Abbildungen drei und vier zeigen nun zwei Präsentationsansichten dieser Eingabefelder, die Abbildung drei zeigt einen Ausschnitt aus einem Screenshot einer Sitzung mit einem Webbrowser.

In Zusammenarbeit mit den Landesdatenschutzbeauftragten der betroffenen Bundesländer wurde ein Datenschutzkonzept erstellt. Die Daten des CAPNetz-Projektes werden zentral auf einem Server gespeichert, der durch eine Firewall vor unbefugten Zugriffen von außen geschützt ist. Die Clients haben auf die Studiendaten nur über das HTTPS-Protokoll Zugriff, welches die übertragenen Daten in einer geeigneten Form verschlüsselt. Durch die Nutzung eines sicheren Übertragungsprotokolls auf der Anwendungsebene kann, unabhängig von den auf unteren Ebenen benutzten Übertragungsprotokollen, sichergestellt werden, dass keine vertraulichen Daten das Netz verlassen. Dies ist bei der Benutzung von mobilen Eingabegeräten von besonderer Bedeutung, da Funknetze sehr leicht abgehört werden können, ohne dass dabei ein physikalischer Zugang zu dem Netzwerk notwendig ist: Versuche der Hersteller die Netze gegen die ungefugte Benutzung zu sichern haben bislang keine zufriedenstellende Ergebnisse ergeben (die zunächst genutzte Methode, WEP (Wired Equivalent Privacy) [WECA02][BGW01], konnte schon sehr bald durch eine Schwäche des Verschlüsselungsalgorithmus umgangen werden und auch andere Mechanismen zeigen ähnliche Schwächen). Daher war es unumgänglich, die übertragenen Daten bereits auf der Anwendungsschicht zu sichern. Umso mehr, als dass auch fest installierte Netze nicht wirkungsvoll vor Angriffen geschützt werden können. Eine gesicherte Verbindung über SSL zwischen Client und Server ist somit eine geeignete Form, schützenswürdige Daten zu übertragen. Nahezu alle Webbrowser unterstützen diese Art der Verschlüsselung und eigene Clientapplikationen können durch die Einbindung freier Verschlüsselungsbibliotheken [OSSSL] diese Sicherheitsmaßnahme verwenden.

Bei der Offline-Eingabe der Daten muss neben der Synchronisation und Übertragungssicherheit ein weiteres Problem berücksichtigt werden. Das Eingabegerät selbst kann verloren gehen und damit sind die auf ihm gespeicherten Daten nicht länger sicher. Dieses Problem wird dadurch angegangen, dass die Daten auf dem mobilen Gerät bis zur nächsten Synchronisation mit dem Server verschlüsselt gespeichert werden. So ist gewährleistet, dass eingegebene Daten für einen eventuellen Angreifer unbrauchbar sind.

Andere Arbeiten

Der Markt für Content-Management Systeme ist stark zersplittert, doch die meisten CM-Systeme sind überwiegend zur Bearbeitung von Texten redaktionellen Inhalts gedacht und verwalten daher hauptsächlich Dokumente und Artikel. Vertreter dieser Gattung sind beispielsweise NetObjects Fusion [Fusion], OpenCMS [OCMS] und Zope [Zope]. Zope ist in sofern besonders interessant, als es explizit die Möglichkeit vorsieht, Erweiterungen (sog. „Zope Applications“) einzubinden, allerdings ist die für den medizinischen Einsatz vorgesehene Erweiterung ZopeMed [ZMed] in einem (bestenfalls) rudimentären Stadium. Diese Systeme sind nicht auf die entfernte Erfassung hoher Datenmengen hin entwickelt.

Zur elektronischen Erfassung und Weiterverarbeitung von Daten existieren v.a. im medizinischen Bereich sog. EDC (Electronic Data Capture) oder e-Clinical Trials Systeme [MKG] [EDM]. Um Daten entfernt zu erfassen, finden sich kommerzielle und freie RDE (Remote Data Entry) Systeme. Einige dieser Systeme ermöglichen die Erfassung von Daten synchron übers WWW [Wmed] [ORA] oder andere Protokolle [PhO]. Die Erfassung übers Internet wird auch als IDC (Internet Data Capture) bezeichnet. Andere hingegen erlauben die Erfassung asynchron im Offline-Modus PhOSCo [PhO] ist hierbei ein System, das unseren Anforderungen an RDE am nächsten kommt. Lediglich die Verwendung sehr kleiner mobiler Erfassungsgeräte (z.B. PALM) erscheint uns hierbei problematisch zu sein. Diese Systeme sind nicht für die Verwaltung von textuellen Inhalten (CM) geeignet.

Das im Rahmen des CAPNetz-Projektes realisierte WCMS ist für beide oben genannten Funktionen konzipiert und bietet eine einheitliche Plattform für Web Content Management, Internet Data Capture und Remote Data Entry. Durch die feingranulare Zugriffskontrolle auf Einzelkomponentenbasis kann eine einheitliche Plattform für alle Benutzer (vom Besucher des Internetauftritts bis hin zum Prüfarzt) mit einheitlicher Bedienung und Benutzerführung realisiert werden.

Diskussion und Zusammenfassung

Die Entwicklung eines komponentenbasierten Web Content Management Systems mit den Grundfunktionen der einfachen und beliebig granularen Komposition von Komponenten zu Webseiten, eines feingranularen Zugriffskonzepts und unterschiedliche Ansichtsmöglichkeiten hat sich bereits bewährt. In einem Zeitraum von nur fünf Monaten wurden bereits sieben CRFs online erstellt und über 250 Krankheitsfälle eingegeben. Insgesamt beinhaltet das WCMS bereits 45 unterschiedliche Arten von Komponenten und etwa 900 Instanzen.

Außer der strukturellen Ansicht, wurden bisher zwei HTML-Ansichten zur Eingabe und zur Bearbeitung von Studien entwickelt. Diese sind selbst bei umfangreichen Komponenten auf kleinen, handlichen Webpads mit einer Auflösung von 800x600 noch übersichtlich und gut bedienbar. Weitere Ansichten für Kleinstgeräte müssen noch entwickelt werden. In Arbeit ist eine elegante Definition von Interaktionsmöglichkeiten einer WCMS-Komponente. Da die Kommunikation mit einer Komponente (wie z.B. Darstellungsanforderung, Eingabe und Datensynchronisation) über sicheres HTTP geschieht, ist eine genaue Abbildungsvorschrift von HTTP-Request Parametern auf Methodenaufrufe in den Komponenten erforderlich. Ebenso wird an der detaillierten Definition des Synchronisationsprotokolls und erster Clientapplikationen zur Offline-Eingabe gearbeitet.

Zur Beschreibung der Semantik von klinisch-epidemiologischen Studien wird beabsichtigt, Techniken des Semantik Web anzuwenden. Über eine RDF-Beschreibung der CRFs und seiner Relationen ist es denkbar, dass unterschiedliche RDE-Systeme Studieninformation austauschen können oder webbasierte Auswertungssysteme diese analysieren können.

Referenzen

- [BGW01] Nikita Borisov, Ian Goldberg, und David Wagner: Security of the WEP algorithm, <http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html>
- [FP02] Sonicblue: Fontpath Progear Webpad, <http://www.frontpath.com/>
- [HW02] Höft&Wessel Skeye Webpanel AG: Skeye Webpad, <http://www.skeye.de/>
- [OSSSL] OpenSSL, eine freie Implementierung des SSL-Protokolls, <http://www.openssl.org/>
- [WECA02] WECA über Sicherheit in WANs; <http://www.weca.net/OpenSection/secure.asp>
- [Fusion] NetObjects Fusion, <http://www.netobjects.com>
- [OCMS] OpenCms Project, <http://www.opencms.com>
- [Zope] Zope Application Server, <http://www.zope.org>
- [ZMed] Open Source Web Telemedicine, <http://www.zopemed.org>
- [MKG] Dr. Manfred Köhler GmbH, <http://www.koehler-freiburg.de/>
- [EDM] EDC Management, <http://www.edcmanagement.com/>
- [Wmed] WISEmed; TU Dresden, <http://wiseweb.wiwi.tu-dresden.de/wisemed/>
- [PhO] PhOSCo, Guillemot Design Ltd, <http://www.phosco.com/overview.pdf>
- [ORA] Oracle Corporation, Remote Data Capture in Clinical Trials, White Paper, http://www.oracle.com/industries/life_sciences/rdc_wp.pdf, 2001.