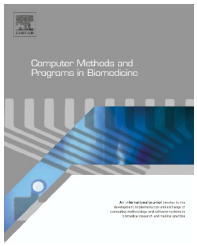
COMPUTER METHODS AND PROGRAMS IN BIOMEDICINE 110 (2013) 369–379

journal homepage: www.intl.elsevierhealth.com/journals/cmpb



A mobile and asynchronous electronic data capture system for epidemiologic studies

Jens Meyer, Daniel Fredrich, Jens Piegsa, Mohamad Habes, Neeltje van den Berg, Wolfgang Hoffmann*

Institute of Community Medicine, Section Epidemiology of Health Care and Community Health, Ellernholzstr. 1-2, 17487 Greifswald, Germany

ARTICLE INFO

Article history:
Received 2 January 2012
Received in revised form
4 September 2012
Accepted 30 October 2012

Keywords:
Mobile asynchronous data capture
Electronic data capture (EDC)
Electronic case report form (eCRF)
Data dictionary

ABSTRACT

A Central Data Management (CDM) system based on electronic data capture (EDC) software and study specific databases is an essential component for assessment and management of large data volumes in epidemiologic studies. Conventional CDM systems using web applications for data capture depend on permanent access to a network. However, in many study settings permanent network access cannot be guaranteed, e.g. when participants/patients are visited in their homes. In such cases a different concept for data capture is needed. The utilized EDC software must be able to ensure data capture as stand-alone instance and to synchronize captured data with the server at a later point in time. This article describes the design of the mobile information capture (MInCa) system an EDC software meeting these requirements. In particular, we focus on client and server design, data synchronization, and data privacy as well as data security measures.

The MInCa software has already proven its efficiency in epidemiologic studies revealing strengths and weaknesses concerning both concept and practical application which will be addressed in this article.

1. Introduction

The Institute of Community Medicine (University of Greifswald) has developed Central Data Management (CDM) systems for a variety of epidemiologic studies, including the Study of Health in Pomerania (SHIP) [1,2], the project "Personalized Medicine" [3,4], GP-supporting, community-based e-health-assisted, systemic intervention (in German: "Arztentlastende, Gemeindenahe, E-Healthgestützte, Systemische Intervention", AGnES) [5], and Greifswald Approach to Individualized Medicine (GANI_MED) [3,6]. These comprise both clinical and population based epidemiologic studies which all have in common (1) a high rate of data acquisition performed, (2) synchronously at different locations. The total sizes of data captured and entered into the system in the different studies ranges from a few gigabytes up to approximately 10 terabytes. To manage the data capture, plausibility checking, data integration, data storage, data refinement, and data transfer in these complex studies an adequate CDM is of utmost importance [4]. In many cases a conventional CDM system composed of web applications for data capture and central databases provides a suitable solution, but in epidemiology those pure central systems often reach their limits. The primary limit is the accessibility of network applications as data input tools. If the tools are made available in an internal, secured network, data capture can only be carried out from personal computers within this network. Even if the accessibility is not restricted

* *Corresponding author*. Tel.: +49 3834 867544; fax: +49 3834 867752.
E-mail address: jens.meyer@uni-greifswald.de (J. Meyer).
0169-2607/$ – see front matter © 2012 Elsevier Ireland Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cmpb.2012.10.015

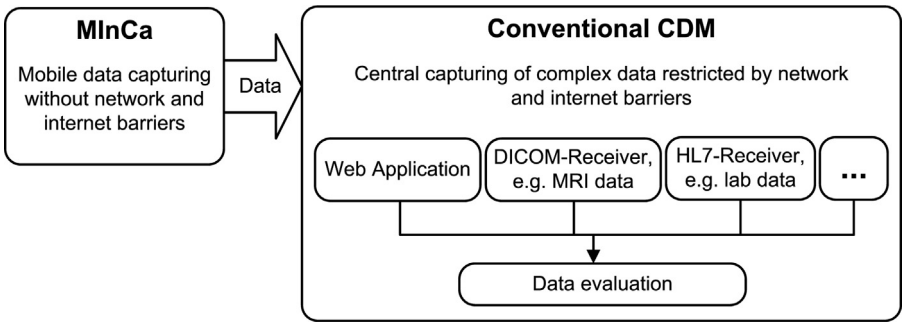


Fig. 1 – MInCa as extension of the conventional CDM system described in [4].

to an internal network, internet access is indispensable for entering data. Furthermore, questionings of study participants often take place in rooms where a designated personal computer is not accessible. In these cases interviewers resort to paper-based Case Report Forms (CRFs), instead of electronic data capture (EDC) systems. Because of these limitations the request for mobile devices in epidemiologic studies is increasing steadily [7]. A prime example is the AGnES project. In this project qualified members of GP practice teams visit patients at home to collect data, e.g. in standardized computer-assisted personal interviews (CAPI). Many patients have no Internet access. Thus, a flexible and independent solution for mobile data capture is necessary. Another example is the GANI_MED project, where medical specialists collect clinical data from in-patients of the Greifswald University Medical Center and from cooperating specialist surgeries. Data capture needs to be mobile inside and outside the clinical structures and cannot rely on the availability of the hospital's wireless LAN. Medical specialists must be able to enter data at any location and at any time. In order to meet the resulting requirements we designed the mobile information capture (MInCa) software as extension of a conventional CDM system (see Fig. 1).

2. Methods

2.1. *Requirements*

A conventional CDM system with web-based EDC interfaces and central databases has to satisfy a variety of requirements, especially concerning the data privacy and data security [4]. Extending a CDM system by mobile EDC interfaces will address additional requirements primarily concerning data privacy/security, data synchronization between client and server, and maintenance of clients. The key requirements of the MInCa software are shown in Table 1.

Secondary requirements can be derived from the key requirements in Table 1. Due to the independency of the MInCa client from the network, data have to be stored temporarily in a local database. Thus, during offline periods the database on the client stores sensible patient data which need to be appropriately encrypted. The next time the client is online, stored data is transferred to the central MInCa server. This transfer requires an adequate encryption algorithm and techniques to prevent any data manipulation.

Robustness of the MInCa clients is a further issue resulting from the independency. User activities like killing processes, shutting down the system unexpectedly etc. should under no circumstances result in data loss, in data inconsistencies or in critical client failures. Furthermore, users must be able to enter data on mobile devices very quickly and ergonomically with help of available tools. Thus, electronic Case Report Forms (eCRFs) have to be very user-friendly and optimally geared to mobile data capture. Additionally, a high client performance is necessary in order to prevent interruption during data entry caused by transmission of large data volume.

2.2. *Design*

2.2.1. *Client*

The MInCa client serves as a user interface for data capture. The first challenge is the achievement of compatibility to a multitude of mobile devices. The data capture should not be restricted to one kind of mobile devices. In order to achieve an accurate compatibility, the software is based on Java. An installed Java Runtime Environment (JRE) is the only prerequisite to execute Java programs. Consequently, the MInCa client can be used on each mobile as well as resident device for which a JRE is available.

MInCa clients are designed as rich clients. Thus, they can operate without any connection to the server. This essential requirement comprises two main aspects: (1) the client has to run as stand-alone application with implementation of all application logic concerning data capture and -management, and (2) a robust and encrypted mobile database to store captured data.

The distribution of MInCa clients is based on the Java Web Start technology [8]. Java Web Start is an efficient way to facilitate the distribution and the maintenance of standalone Java programs with a minimum of manual interaction. Only a web server acting as distributor for the client application is needed. Users can follow a web link to contact the web server and ask for the application. Again, the only prerequisite is an installed JRE. After the first installation the client can be started via a desktop icon. Each time the client is started Java Web Start tries to connect to the web server in order to compare the client version. If the server is accessible and a software update is found, the new version will be installed immediately (see Fig. 2).