Privacy-Aware Agent-Oriented Architecture for Distributed eHealth Systems

Adel Taweel[1,2], Samhar Mahmoud[1], and A. Rahman Tawil[3]

[1] King's College London, UK
adel.taweel@kcl.ac.uk
[2] Birzeit University, WB
ataweel@birzeit.edu
[3] University of East London, UK
a.r.tawil@uel.ac.uk

Abstract. Distributed Integrated ehealth systems are becoming a key need for achieving improved healthcare, in which healthcare processes across organisations must work in tandem to achieve this goal. However, at its core, enabling data-sharing safely while maintaining privacy and confidentiality is a critical requirement. The wide spread of electronic health record systems to manage health data and healthcare processes may provide the needed infrastructure to facilitate data-sharing. However, most of these systems are often designed to work within localised settings and rarely across organisations. In a health service, organisations and individuals are autonomous and often obey different data governance policies and would require different levels of data-sharing needs, depending on their roles and goals within the service. This would make agent-oriented architecture a strong candidate to enable privacy-aware seamless data-sharing between participating organisations. The paper presents an approach for privacy-preserving agent-oriented architecture that enables organisations to work together overcoming sharing sensitive data and evaluates its use within a real-life project.

Keywords: eHealth, Agent Architectures, privacy, security, System of Systems.

1 Introduction

The benefits of technology supported integrated health is increasingly recognised as one of the key needs of a modern health service. It is crucial not only to reduce costs, improve healthcare and patient safety but also due to the increase expectation of patients to receiving care at points of care irrespective of location or time [15][16]. However, to achieve they require health organisations to share and exchange clinical-sensitive information within a robust privacy-preserving environment. Organisations, within a single health service let alone across different ones, often operate autonomously governed by their individual data governance policies. The widespread of the use of electronic health record systems, in health organisations, and recent advances in networking and information systems provide the needed infrastructure to enable such paradigm [15][16]. However, transferring highly-sensitive data is not without

R. Meersman et al. (Eds.): OTM 2014 Workshops, LNCS 8842, pp. 418–427, 2014.

risks and poses several security concerns [1][9][10][11] especially when different orgnisations require sharing such data to achieve their function, in which demands for different levels of data-sharing needs vary, and where privacy preservation and control of usage is required [2][3][6][7].

On the other hand, multi-agent systems provide potential solutions to address some of these issues. In their intrinsic design, they have shown to meet needs in several applications including high-speed, mission-critical, content-rich, distributed information systems where mutual interdependencies, dynamic environments, uncertainty, and sophisticated control play a notable role [5][12][13][19]. eHealth applications can utilize these intrinsic characteristics of multi-agent systems given the notable features that these applications expose. They are often composed of autonomous (complex) systems, realised by heterogeneous components and legacy systems; designed to dynamically manage distributed data and resources within inherent regulatory frameworks, and built with regulated interactions for collaboration [17][18]. The key requirement however here is to provide an approach to enable sharing of data between different organisations and/or institutions while preserving the privacy of patients and users of the systems, and only allow access at the right points of access by the right people in the system. This would require providing adequate data processing security aligning with privacy legislation. Several approaches have been proposed, which often focus on control [3][6][7] and/or enforcement mechanisms [2][4][8]. These approaches, often however take heavy-handed approach with management of policy configurations, verification and validation steps that may make them unnecessarily expensive and potentially not scalable for some applications.

This paper proposes a different approach to enable data-sharing while preserving the privacy and confidentiality of data between participating organisations. It considers a particular application, in which different types of organisations have different goals and actors carry out different roles, yet they are required to collaborate and share information to meet the system requirements. It proposes an agent-oriented architecture that uses one-directional hashing for the sharing of sensitive data, where agents integrate with information systems (e.g. EHRs) and handles data transfers and the levels of encryption of data where only destination organisations can re-identify. This architecture is evaluated in the context of a real-life project environment, which demonstrates such architecture can meet the system set goals.

Section 2 describes the case study of a distributed ehealth system and summarizes its main attribute and challenges as an ehealth system and its particular requirements for privacy and confidentiality; Section 3 describes the privacy-aware architecture, its components and usage through a scenario; section 4 describes the evaluation and lesson learned from deploying the architecture in a real-life environment; finally, Section 5 presents some conclusions and future work.

2 Case Study

One of the major obstacles in clinical research is finding enough eligible participants to recruit in clinical trials [13][14][15]. Eligible participants can potentially be found automatically, but not without accessing their clinical information. Patients' clinical data, however, is stored into EHRs, located in their local clinics, which are only accessible by their own clinicians. In the UK, healthcare organisations are

autonomously isolated and function within a separate framework from that of clinical academic or clinical research institutions, which makes accessing patient information by clinical researchers extremely difficult and not without going through an enormous governance and regulatory process. The IDEA project aims to locate, identify and invite patient to participate in trials. However, to do so, it requires enabling data sharing between these different types of organisations, potentially on a large scale.

Fig. 1 illustrates the overall conceptual architecture of IDEA. It notifies practitioners in real-time whenever an eligible patient is in consultation. When a patient visits a practice, IDEA compares their details against a registry of actively recruiting trials; if the patient is found eligible for one or more, the practitioner is prompted to help recruit the patient if they are interested. The IDEA project is described in more details here [13] [5].

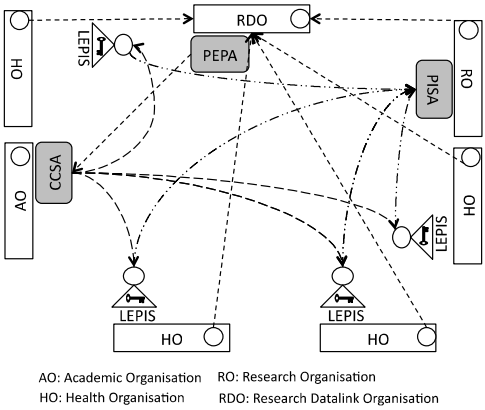

Fig. 1. Conceptual Architecture

However, to enable a more open system in which these organisations can share different levels of data, a flexible agent-oriented architecture was developed. As shown in Fig. 1 there are four types of organisation in this case study:

- *Healthcare Organisations* (HO): *HO* organisations have and must follow strict clinical and governance regulations concerning the privacy and security of their patients. Each adhere to national regulations but also implements their own set of data access policies. These organisations, in our case study, are represented by the General Practice clinics, which each has a pool of patients that only a designated set of local staff can access their data.

- *Research Datalink organisations* (RDO): *RDOs* represent a trusted third party organisations that also have access to a pseudnomyised (i.e. have access to clinical structured data but no access to personal identifiable information). *RDO*s are often national organisations created to enable clinical research through ethically controlled access to clinical data by clinical researchers.

- *(Clinical) Research Organisations* (RO): *ROs* represent research institutes or clinical researchers who run clinical trials. They do not have access to clinical data but require identifying eligible participants for their trials. Only once participants agree to participate in a respective trial, they may obtain their clinical.

-*Academic Organisations* (AO): *AOs* are organisations that undertake clinical or technical research within academic settings. They do not have nor allowed access to clinical data.

The case study presents several challenges that need to be addressed to achieve its objectives; some of the relevant challenges are identified below to gain a better understanding of how the architecture may need to meet the privacy objectives of the system.

*Privacy***.** The architecture must provide organisations with mechanisms to safely share data meeting governance and regularity requirements, especially *HOs*, *RDOs* and *RO*s while ensuring only 'right' organisations can access and re-identify patients. For example, *HOs* can only access their own patients' data, *RDOs* can only access their own data and *ROs* can only access consented participants data and so forth.

System interactions. To be able to recruit eligible patients, it is necessary for researchers, practitioners, patients, databases and practices to interact. This means that several independent institutions, which are completely autonomous and have their own independent goals, must cooperate to achieve a common objective. However, the integration of multiple heterogeneous and autonomous systems can be a complicated and resource-consuming task. There are several issues to address here, including interoperability, distributed data etc., which are beyond the scope of this paper. The focus here is on privacy-preserving between organisations, which is key for the function of the system. The developed privacy-preserving mechanism, however, should not constraint these critical interactions in meeting the system function.

*Scalability and performance***.** To maximise chances of finding more eligible participants, the system must be able to get the information from as many practices as possible (more than 10K GP practices in the UK) and manage a huge number of clinical trials requirements. However, due to the number of potential active trials (potentially several hundreds) with the size of each trial description and eligibility criteria makes it impossible for GPs to know all active trials to assess patient eligibility during, the often short, consultation period.

3 Privacy-Aware Architecture

This section details the design and components of the architecture. Fig. 2 depicts the main structure of the architecture mainly illustrating the components that handles privacy-perserving mechanism. The architecture adopts a decentralised approach, where a decision of access is decided at the point of access and controlled by individual organisations. Since each organisation is assumed autonomous, each employs an agent that defines its roles, goals, and policies. These determine the behaviour of agents when sending or receiving data. Each agent has the full authority to determine its action with respect to data. When data originates from another organisation, the source-organisation sends data in a data-bucket (DB), which includes four elements:

DB=<PE, DE, OIK, LoE>

-Protocol_Element (PE): defines the characteristics of eligible participants (often named eligibility criteria), which include a Protocal_Id_Key (PIK);

-*Data_Element* set (DE): include specific data of potentially eligible patients, which contains fields of data and Data_Id_Key (DIK) that defines a set of keys of the Data_Element IDs, which specifies the Id of each element in the data set;

-*Organisation_Id_Key* (OIK): key of ID of the organisation where the data originated;

-*Level_of_Encryption* (LoE): specifies how many times the data has been encrypted;

DE (including DIK), PIK, OIK are encrypted, however PE (eligibility criteria) and LoE are not. These are encrypted using a standard one-direction hashing algorithm (e.g. MD5). As mentioned above with for the four types of organisations, four types of agents are created to represent each function and role. Although these are generic types, individual agent mirror individual organisations, and thus their exact behaviour depends on the policies and exact role of each organisation. As mentioned above, *HOs* have identified version of the data of their patients, whereas *ROs* do not have access to data, only except after a participant has consent and provided their data. *AOs* do not and should not have access to data. *RDOs* may have patient clinical data but not necessarily personal information. The function and role of each these agents are described below in more details.

PISA: *Protocol Information Service Agent* is a software agent that represents a *RO*. It represents a *Protocol Manager Role*, which defines and creates protocol eligibility criteria and defines the protocol characteristics. It includes the *Researcher Role*, which is responsible for defining the specific features of each trial under its jurisdiction. Researchers are also responsible for ensuring consistency of their protocol, determine which type of trial this protocol belongs to (see below) and activating the trial in CCSA (see below). They are not allowed to directly contact patients unless they have agreed and consented to participate in one of their own clinical trials. For obvious reasons, each researcher should be part of a specific research institution and follow its specific restrictions. There are two types of trial protocols: *T1*: trials that define participants characteristics only (e.g. age>40 & gender=male etc.), and *T2*: trials that specifies complex characteristics which require pre-specifying potentially eligible patients and their authorised clinicians. The latter (i.e. *T2*) requires further epidemiological pre-processing to identify potential participants that meet particular risk assessment. For the former (i.e. *T1*), *ROs* (*Protocol Manager*) can create these, however for the latter (i.e. *T2*), *ROs* require to pre-consult with RDOs (PEPA) to create these trials before they are sent to the CCSA.

PEPA: *Potential Eligible Patient Agent* is a software agent that represents the *RDO Manager Role*, which is responsible for updating and controlling access to the *RDO* database. It offers a service to pre-compute potential eligible patients for individual trials that have complex search criteria (*CreateEligibleList* service). The role also offers a service to identify whether a GP (and their own practice) is authorized to perform recruitment for each trial (*AuthorizedGP* service), while adhering to local and good clinical practice regulations. This agent (*PEPA*) provides *ROs* with a list of potentially eligible patients, their practices and authorised GPs. However, since *ROs* are not allowed to have access to this data (yet), data is encrypted as part of the *DE* data-bucket. Since this encryption is irreversible, only *HOs* that have respective patient data

are able to re-identify it. When an *HO* receives this data, using *DIK*, they can match it to their local data and re-identify the patient using a simple encrypt-search algorithm (see below). *PEPA*, for some cases, could choose to double-encrypt the *DE*, depending on their policy, if so, the agent then increments *LoE*, so other receiving agents can determine the number of encryptions they need to perform in their matching algorithm.

CCSA: *Central Control Service Agent*, represents *AOs*. *AOs* provide the patient recruitment service along with their network of recruiting agents (*LEPIS*) that are connected to respective *HOs*. It represents the *CCSA Manager Role,* a software application responsible for controlling the *CCSA* database, which stores data about active clinical trials. It offers three services to the other members of the system: (i) a *Register New or update existing Trial* service that allows researchers *(PISA*) to register new clinical trials in *CCSA*; which also verifies that trials follows the specified standards and regulations; (ii) an *Update LEPIS Status* service that ensures consistency with *LEPIS* agents to optimize *LEPIS* updates; and (iii) a *Patients Response* service that, in communication with *LEPIS* Agents, records the response of each consulted patient (and/or their GP) (whether they have agreed or refused to participate in a trial). Since *AOs* are not allowed to have access to clinical data, thus *CCSA* stores data-buckets (*DB*), as received, encrypted.

LEPIS: *Local Eligible Patient Identification Service* agents represents *HOs* within this role of recruitment. It defines the *LEPIS Manager Role* as a software application that resides in each practice and investigates the eligibility of patients. *LEPIS* agent plays this role for each practitioner in each GP practice participating in the recruitment system. *LEPIS* agents continually communicate with the *CCSA* to acquire information about trials related to the type of patients and speciality of GPs and practices and other updates on each trial status. They also provide the GP with a simple GUI interface to notify them of a patient's eligibility. As *LEPIS* agents communicate with GPs through their GUI interface, they collect information about patient and GP responses. However, since *LEPIS* obtains the data encrypted from CCSA, it is kept encrypted in it local data store. To match the eligibility of patients for *T2* type trials, it uses a simple encrypt-search algorithm, in which when a patient is in consultation with a GP, their *DIK* is only encrypted to match with *LEPIS*'s stored encrypted key. If it matches, the rest of the criteria are matched against the patient record to identify eligibility. Since *LEPIS* represents *HOs*, it is authorised by its role to access the data, but no identifiable data is transferred by *LEPIS* outside the *HO* to other agents. *LEPIS* only communicates patients/GP responses back to *CCSA* using their encrypted *DIKs*. If the patient agreed to participate in a trial, *LEPIS* identifies and communicate *PIK* back to the respective *RO* (*PISA*), which consequently can invoke the appropriate case report form to complete the recruitment of the patient into that trial. In this case, to preserve privacy *LEPIS*, sends the encrypted *DB* to *PISA*. Depending on the *LoE*, *PISA* could easily re-identify *PIK* using a simple encrypt-search algorithm.

SEC: *Search-Encrypt Component* is a software application that includes a search-encrypt algorithm. The algorithm uses a sequential search mechanism to locate *DIK* with the local database, encrypt and provide to the agent to match it the encrypted *DIK* in the data-bucket. It's not always used by respective agent, or by all agents, depending on its functionality and role. *CCSA*, for example does not invoke this component, since it has no access to the source data and its role does not require it to re-identify users.

NCC: *Node Connector Component* is a software application that allows agents to connect to local software in each organisation. Its detailed description is beyond the scope of the paper and covered in more details elsewhere [14][15], but its main function is to overcome interoperability issues with local systems including databases. This has been mainly designed to provide uninform programmable interface to communicate with agents, thus providing transparency over local software or databases.

Within the architecture, each organisation respective agent can push or send that information, on request, to other agents, or act based on their workflow for respective actions. Receiving agents once they receive the data bucket, depending on their role, they decide whether to store it as is or need to do further processing on it. For agents that require further processing, since the data is irreversible, they can use *SEC* to allow them find to whom patient/user this data belongs. Once found, they can do further processing based on their role.

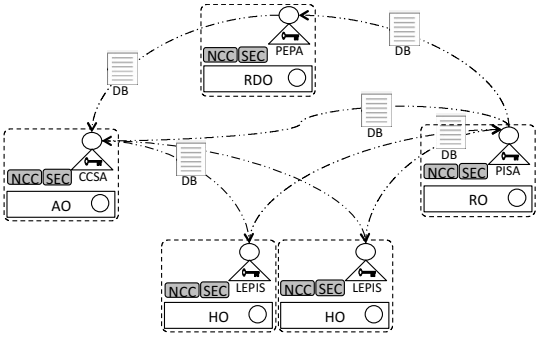

Fig. 2. Agent-oriented Architecture: agents, services and organisations

4 Evaluation and Lessons Learned

This section describes the evaluation of the architecture and lessons learned. The architecture was employed in the IDEA project, described in section 2 above. The architecture was realised using service oriented web services with well-defined interfaces, given its well-established standards. The services were deployed to enable four different types of organisations to achieve patient recruitment into clinical trials. The main challenge was to overcome the regulatory and their consequent networking barriers. It requires that involved organisations obey their individual regulatory constraints, where only allowed staff in respective organisation accesses safely their patient data, while staff in other organisations must not. The *AOs* and *ROs* are the two organisations that do not and are not allowed to have access to clinical data, yet they want to use the data to find, identify and recruit patients. *RDO*s have often access to the data aggregated from participating *HOs* but only pseudonymised. The case study's *RDO* aggregate clinical data from 640 primary care clinics (*HOs*) on a monthly, and from some on a daily, basis keeping data periodically updated. In the IDEA project 60 *HOs* participated in which *LEPIS* agents were installed on 134 GP machines within these clinics. These GP clinics are geographically distributed across the UK and possess limited Internet connectivity. Three key elements were evaluated in these settings: agents, the privacy-preservation and functionality in meeting the objective of the

study. The main objective and functionality of the architecture is to share data between these organisations to enable recruiting patients while preserving their privacy at all time and only organisations that already have patients data are only able to identify them. These three elements are reflected upon below:

Functionality: the architecture was able to successfully recruit patients from the participating *HOs* while protecting patient privacy. Three clinical trials were deployed in the system, and all recruited successfully meeting their targets – a more details on recruitment is reported elsewhere [12].

Privacy-preservation: as described in the scenario above, data-origin-organisations when releasing patients in the architecture, data is encrypted using a simple but effective irreversible hash encryption algorithm. The well-known MD5 algorithm was used to encrypt the data. To ensure that the encrypted data matching algorithm works, the same version is used in all agents and organisations. As per the follow described above, while *RDO* and *HOs* have access to clinical data, only *RDO* (and in cases *ROs*) was originating encrypting data and *HOs* were only performing data matching to re-identify their own patients. That meant, with any data request, *AOs* and *ROs* would only have access to encrypted data within the platform, which is not identifiable. For *RDO* and *HOs*, to identify if they have access to the encrypted data, they use a search-encrypt algorithm, named as *SEC* above. This algorithm used by *SEC*, to match *DIKs*. Since it does not have access to actual local keys it performs a linear search on the local database, where it sequentially encrypts each local Data ID and then matches to the encrypted *DIK* in the data-bucket. In the participating *HOs*, sizes of their patient pool is between 5k-10k patients, which the sequential execution of the *SEC* algorithm did not show as a bottleneck or slowed the performance., however, we recognise this may prove inefficient for large databases. In the case study above, since patients are only invited for recruitment while in consultation, this was not an issue, since only that patient is used for matching by the *SEC* algorithm.

Roles and Agents: as mentioned above four types of agents were developed in the architecture: *LEPIS* (for *HOs*), *PISA* (for *ROs*), *PEPA* (for *RDO*) and *CCSA* (for *AOs*). One instance of each of the latter three agents were deployed in respective organisations, while 134 *LEPIS* agents were deployed in participating *HOs*. Although these agents fall within four types, nevertheless the exact behaviour of individual agents and the decisions they make regarding their role and data processing depends on the local policy configuration of individual organisations and their own workflow configuration that specifies their communication with other agents. Four types were deemed sufficient for the purpose of the IDEA project, but the architecture allows expanding and adding other types. The most complex of these agents was *LEPIS*, given the amount of decisions and interactions it has to perform with local GPs. *PISA* and *PEPA* have some complexity but since interactions were limited to only limited users, their GUI design was minimal. As the IDEA project is being scaled up, the design of these agents is being revised.

A number of key lessons have been learnt from the deployment of this architecture, these include:

Scalability: no scalability issue was observed. However, although there are more than 134 agents deployed in the system to test scalability of the system, scalability may pose a challenge when the size substantially increases. The amount of communication

between *LEPIS* agents and *CCSA* is limited to three times a day, but given the potential scale, *CCSA* may prove to be a bottleneck. To overcome, *LEPIS* agents have been designed to communicate with each other, within a common domain or clinic. Scalability of this designed has been tested and detailed somewhere else [20], which was found to achieve greater efficiency opposed to the initial workflow design.

Security barriers: in the medical domain, in *HOs*, network security is critical but it can be cumbersome. Often, these domains are tightly network secured with stringent firewalls in place. However, these often only allow communication to be initiated from within the domain. Thus this posed some challenge deploying the platform to enable system agents to communicate at will. To facilitate the process, agents have been designed to initiate outgoing communications with other agents, according to a configurable communication flow that defines each agent communication pattern.

SEC algorithm: because the SEC algorithm uses sequential search mechanism to match and re-identify local participants, this may pose another scalability challenge. Although in the current set-up, the data size is limited and thus this was not an issue, future designs may need to consider other mechanisms to improve efficiency for larger size data.

5 Conclusions and Future Work

The paper presents a simple yet effective privacy-ware agent-oriented architecture that enables data sharing between organisations while preserving privacy. The architecture employed the use of role-driven agents that each represented the goals and policies of each organisation, with enabled communication to meet their specific functionality. The agents were designed with privacy-awareness so that their behaviour is controlled within their local settings, including governance and communication policies, taking decisions in regards of how to process and handle data. The architecture was evaluated within a real-life ehealth project and has been shown to maintain privacy of patients, while enabling data sharing which was employed for the identification of eligible patients for clinical trials in real-time. The results obtained show that the architecture has successfully met the sought functionality of the system to recruit patient for three trials, while enable safe sharing of data.

The evaluation has also shown a number of lessons learned from the deployment of the architecture, and potential future development to address these lessons. In particular, lessons related to ehealth environments and their potential scalability needs, posing a potential limitation of the approach.

Acknowledgment. The author would like to thank all colleagues from the IDEA project who contributed to this work, through discussions or providing insights.

References

1. Leon, M., Hipolito, J., Garcıa, J.: A Security and Privacy Survey for WSN in e-Health Applications. In: Electronics, Robotics and Automotive Mechanics Conference, CERMA 2009, pp. 125–130 (2009)
2. Rath, A.T., Colin, J.: Towards purpose enforcement model for privacy-aware usage control policy in distributed healthcare. IJSN 8(2), 94–105 (2013)
3. Rath, A.T., Colin, J.: Modeling and expressing purpose validation policy for privacy-aware usage control in distributed environment. In: ICUIMC 2014, vol. 14 (2013)

4. Dong, N., Jonker, H., Pang, J.: Challenges in eHealth: From Enabling to Enforcing Privacy. In: Liu, Z., Wassyng, A. (eds.) FHIES 2011. LNCS, vol. 7151, pp. 195–206. Springer, Heidelberg (2012)
5. Bergenti, F., Poggi, A.: Multi-agent systems for e-health: recent projects and initiatives. In: 10th Int. Workshop on Objects and Agents (2009)
6. Solanas, A., Martínez-Ballesté, A., Pérez-Martínez, P.A., Pena, A., Ramos, J.: m-Carer: Privacy-Aware Monitoring for People with Mild Cognitive Impairment and Dementia. IEEE Journal on Selected Areas in Communications 31(9-Suppl), 19–27 (2013)
7. Armellin, G., Betti, D., Casati, F., Chiasera, A., Martinez, G., Stevovic, J.: Privacy preserving event driven integration for interoperating social and health systems. In: Jonker, W., Petković, M. (eds.) SDM 2010. LNCS, vol. 6358, pp. 54–69. Springer, Heidelberg (2010)
8. De Coi, J., Delaunay, G., Albino, A., Muhlenbach, F., Maret, P.: The Comprehensive Health Information System: a Platform for Privacy-Aware and Social Health Monitoring. In: IADIS e-Health 2012, Lisbon, Portugal (2012)
9. Appari, A., Johnson, M.E.: Information security and privacy in healthcare: Current state of research. Int. J. Internet Enterprise Manage 6(4), 279–314 (2010)
10. Karygiannis, A., Antonakakis, E.: Security and privacy issues in agent-based location-aware mobile commerce. Saf. Sec. Multi-agent Syst., 308–329 (2011)
11. Rashvand, H.F., Salah, K., Calero, J.M.A., Harn, L.: Distributed security for multi-agent systems - review and applications. IET Inf. Secur. 4(4), 188–201 (2010)
12. Mahmoud, S., Tyson, G., Miles, S., Taweel, A., Vanstaa, T.: M Luck, B. Delaney. Multi-agent system for recruiting patients for clinical trials. In: Proc. AAMAS 2014, Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems (2014)
13. Tyson, G., Taweel, A., Miles, S., Luck, M., Staa, T.V., Delaney, B.: An agent- based approach to real-time patient identification for clinical trials. In: Proc. 4th Intl. Conference on eHealth 2011 (2011)
14. Tyson, G., Taweel, A., Zschaler, S., Staa, T.V., Delaney, B.: A model-driven approach to interoperability and integration in systems of systems. In: Modelling Foundations and Applications: MBSDI (2011)
15. Taweel, A., Delaney, B., Speedie, S.: Towards achieving semantic interoperability in ehealth services. In: Watfa, M. (ed.) E-Healthcare Systems and Wireless Communications: Current and Future Challenges, pp. 388–401. IGI (2012)
16. Taweel, A., Speedie, S., Tyson, G., Tawil, A.R., Peterson, K., Delaney, B.: Service and model-driven dynamic integration of health data. In: Proceedings of the first international workshop on Managing interoperability and complexity in health systems, MIXHS 2011, pp. 11–17. ACM (2011)
17. De Loach, S.A.: Developing a multiagent conference management system using the o-mase process framework. In: Proc. Int. Conf. on Agent-oriented Software Engineering VIII, pp. 168–181 (2008)
18. Gonzalez-Velez, H., Mier, M., Julia-Sape, M., Arvanitis, T., Garcia-Gomez, M.R.J., Lewis, P., Dasmahapatra, S., Dupplaw, D., Peet, A., Arus, C., Celda, B., Huel, S.V., Lluch-Ariet, M.: Healthagents: distributed multi-agent brain tumor diagnosis and prognosis. Applied Intelligence 30 (2009)
19. Vecht, B., Dignum, F., Meyer, J.-J., Dignum, M.: Handbook of research on multi- agent systems: Semantics and dynamics of organizational models. In: Autonomous Agents Adopting Organizational Rules. IGI Global, pp. 314–333. IGI (2009)
20. Feyisetan, S., Tyson, G., Taweel, A., Vargas-Vera, M., Staa, T., Delaney, B.: ePCRN-IDEA2: An Agent-Based System for Large-Scale Clinical Trial Recruitment. In: Proc. AAMAS Workshop on Agents Applied in Health Care (A2HC), Spain, (2012)