IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 5 Issue 7, July 2018
ISSN (Online) 2348 – 7968
www.ijiset.com

A Virtual-Community-Centric Architecture to Support Coordination in a Large Scale Distributed Environment: A Case Study of the South African Public Sector

Godwin A. Thomas[1], Reinhardt A. Botha[2] and Darrell van Greunen [3]

[1] Department of Computer Science, University of Jos,
Jos, Plateau State, Nigeria

[2] Institute for ICT Advancement and School of ICT, Nelson Mandela University,
Port Elizabeth, Eastern Cape, South Africa

[3] Institute for ICT Advancement and School of ICT, Nelson Mandela University,
Port Elizabeth, Eastern Cape, South Africa

Abstract

Coordination in a distributed environment is customarily a difficult endeavour. The work pattern that emerges owing to the complexities associated with size, autonomy, structure and geographical dispersal of the participants provides unique coordination challenges, as evidenced in the South African Public Sector. To take advantage of opportunities and maximise the utilisation of limited resources, entities are required to periodically collaborate. This denotes that the need for effective coordination and cooperative collaboration among role players is crucial, to ensure that the ineffective use of resources does not occur, through incoherence, fragmentation and the duplication of efforts.

This paper presents an overview of Virtual community-centric architecture that has the potential to support and promote sustainable coordination *within the South African public sector*. The architecture looks to support the operations of a context-aware collaboration life cycle model geared to streamline coordination in a distributed environment from a collaborative stand point.

Keywords: *Coordination, Distributed environment, Virtual community, collaboration, Context-aware, South Africa, Public sector.*

1. Introduction

Coordination in a distributed environment is typically challenging, especially when a workforce is physically dispersed, with participants unable to interact directly, making communication and the coordination of activities more difficult. Geographically dispersed workers experience more miscommunications and misunderstandings [1]; have increased difficulty in sharing information [2, 3]; and experience problems associated with receiving feedback [4], than a workforce which is situated in the same location. Despite this, if a workforce is physically separated they can still collaborate towards a common objective; however, there is a requisite for the facilitating mechanisms to ensure they coordinate their work effectively, thereby delivering value to the organisation. To cope with the challenges of coordination, organisations adopt various flexible and decentralised work patterns [5]. Therefore, coordination in a distributed environment subscribes to a loosely coupled, decentralised, network mode of governance, as exemplified in the circumstances of the South African public sector. Acha and Cusmano [6] identify loose-coupling as a form of governance for teams working across organisational boundaries.

Hinds and McGrath [4] and Kiesler and Cummings [7] maintain that dispersed teams will experience fewer coordination problems if there is less interdependence between members at distant sites. Essentially, employing a loosely-coupled work pattern between dispersed workers can provide coordination benefits while maintain a level of autonomy. Happel [8] and Olson and Olson [9], for example, note that dispersed teams who modularise their work by site are able to function more proficiently. However, there are risks, which incorporate isolation, incoherence and the duplication of efforts. This indicates the requisite for maintaining awareness of team efforts. Cooperative work commonly consists of many interdependent activities. For this reason, coordination among stakeholders is critical to avoid conflicting or repetitive actions. The South African public service, as a typical distributed environment, is no different, in that proper coordination among the various organisations charged with delivering public policy/services will prevent both redundancy and gaps in service delivery. This suggests that ensuring proper service delivery requires effectively coordinated activities from cooperating organisations or stakeholders.

However this is a challenging endeavour, South Africa has a complex governmental structure involving a number of provinces, local governments and municipalities with different authorities and responsibilities [10]. The three spheres of government are distinct, but are also interdependent, as they must work together towards a common governmental goal. Thus, all the sectors of government are for instance required to observe the principles of cooperative government, as set out in Chapter Three of the Constitution, which includes the coordination of activities to avoid duplication and waste among other things [11]. The challenges associated with geographic dispersal of documents; their manual integration and the inadequate application support available, all impact on coordination, as there is limited overview regarding activities and resources. This results in, *inter-alia*, conflict bookings duplication, incoherence and the overextension of staff. More so, the potential opportunities to work together is obscured from various stakeholders with similar interests, as well as myriad of other issues.

Although the idea of utilising technology to support coordination is not recent, especially in Computer-Supported cooperative Work (CSCW) research, the variety of solutions offered are frequently limited in scope [12]. Customarily, they are quite specific to the unique requirements of a situation. For example, solutions could be targeted towards a unique coordination mechanism, for instance, communication, data sharing, activity conditioning or sequencing of workflow. The complexities associated with the distributed nature of the environment, heterogeneity, autonomy and the increasing need to collaborate engenders the need to develop a novel coordination support system for the segmented and distributed South African public sector. Thus, this study proposes an architecture to help manage and promote the sustainable coordination of collaborators in a heterogeneous and distributed environment. Tellioglu [13] stresses the need for a collaborative work environment that considers socio-technical factors together with a guide to support collaboration.

It has been argued that virtual communities present a suitable environment that can fill collaborative gaps in traditional, hierarchical organizations in an increasingly networked society [14, 15, 16]. Considering the distributed nature of the South African public sector, as well as the number of decentralized and dispersed players, virtual community has been argued to be a suitable platform for sustainable coordination in a distributed environment [15].Thus, the premise of the proposed architecture is that by leveraging the promise that virtual community properties offer seamless and sustainable coordination may be realized. The architecture looks to define the core features and functionality from which an implementation can be developed. Four fundamental functions of the supporting architecture are introduced and discussed as they would support the operation and activities of the proposed collaboration lifecycle model geared towards streamlining coordination in a distributed environment. It advocates an adaptive environment that needs to exist to host a particular behavioral style which is represented as the context-aware collaboration life-cycle process model (CCLCM) which deals with the dynamic aspect of collaboration often subject to several needs contexts as detailed in [17]. The proposed architecture represents a technical virtual community-centric, context-aware environment that can host and support the process-based operations, which is the focus of this paper. However, a review of the life-cycle meta-process is presented to aid discussion. The subsequent sections begins with a discussion on the methodology employed this is followed by a discourse on virtual communities' promise on coordination. Thereafter, the design requirements are revealed, which is followed by a brief on the context–aware collaboration life-cycle model. This is succeeded with a discussion pertaining to, the architecture, followed by a rudimentary evaluation of the architecture components in terms of their services to CCLM modules, where after, the paper culminates with a conclusion and suggestions on future work.

2. Research Methodology

Guided by the requirements as identified based on a case study in Thomas et, al, [18] an architecture towards the support of sustainable coordination in a distributed environment is proposed in this study. The architecture build process employed general design principles, which suggest grouping common things together to effect easier understanding and management. The design approach employs the classic hierarchical design and business process design methodologies [19, 20]. The hierarchical design pattern advocates multilevel abstractions and designs from which system parts can be considered through decomposition and assembly. The bottom-up and top-down approach employed, referred to as Meet-in-the-middle Methodology [21] is guided by the set objective specifications to define the abstraction levels of the model components. Essentially, it employs the modularization principle, which advocates the separation of concerns where functions are separated into distinct parts enabling a complex whole to be divided into smaller and simpler parts, in order to help manage the complexity of a system. These self-contained parts can stand alone or can be intuitively and logically joined together to address a problem [22]. The interaction between logical components is premised on causality relations, a concept that reflects cause and effect. The technique employed identifies input and output relations to create a graphical representation.

Guided by the case study requirements identified in Thomas et al., [18] the composition and fit of the Architecture components was established and synthesized through argumentation [23, 24] and continuous discussions with experts to determine the level of abstraction, granularity and fit between the proposed architecture components that will adequately support the functions of the virtual community centric collaboration life cycle discussed in section. Thus the architecture is founded on assertions arising from reasoned discourse. Discussion with experts determined architecture component granularity and mapping justification to which component it addressed in the lifecycle model in the form of an evaluation table indicating the components contribution to the model services rendered. Thus, an explication demonstrating how the architectural functions support the collaboration life-cycle model activities comes to fore as indicated in section c and appendix A. The experts engaged are proficient in Information Technology service management, Workflow, end to end service design, and Human Computer interaction (HCI) respectively.

Furthermore, a scenario based approach premised on the cooperative work practices derived from the case study analyses [18] is utilized to externalize the services resulting from the functions of the architecture components from an awareness perspective. The model to be supported by the proposed architecture is overviewed in the next section.

3. The Context-aware Collaboration Life-Cycle Model (CCLCM)

The CCLCM represents a behavioral model in the form of a context-aware lifecycle model aimed to streamline coordination of collaborative activities in a heterogeneous distributed environment [17]. The proposed model latches on to the principle of articulation work, project management and the design science research process respectively as the overarching process to help coordinate collaborative efforts in distributed environment. The processes and procedures which occur are reflected in phases where outputs are produced to serve as input to the next, ensuring a consistent approach. Furthermore, there may be simultaneous existences of a collaborative project.

The CCLCM model approach as shown in figure 1 suggests a continuous process that starts with the identification of needs from the environment, developing and implementing plans, monitoring and reviewing implementations. The needs awareness that eventually result into opportunities for collaboration must be realised prior to initiation. Essentially, the environmental requirement in the centre of the model focuses on understanding and documenting needs and interests to a certain degree of detail, to allow for analysis and opportunity identification.

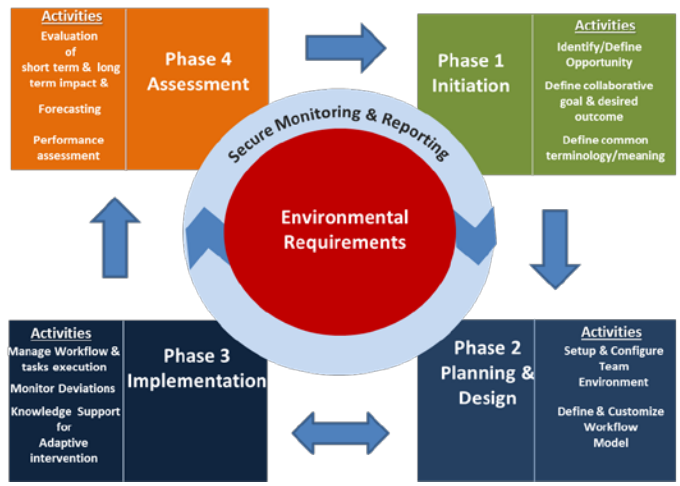

Fig.1 The Context-aware Collaboration Life-Cycle Model [17]

The initiation phase continues the analysis and interpretation of requirement to define an opportunity to collaborate, set goals and parameters. The planning and design phase, provides specifications and configurations of people and artefacts that characterises the project transformation plan. This indicates that it includes all the activities necessary to acquire and establish the resources needed to carry out the project. Subsequently, the implementation phase is activated, executing plans, processes, or procedures in accordance to the specifications defined in the master plan or reference template to produce outputs, while managing changes that may occur. The evaluation phase determines whether collaboration requirement has been satisfied or is some redress action still required. The monitoring and reporting phase occurs prior to and during phases of the project to, *inter alia*, monitor environmental changes; requirements status; resources; schedule; quality; risks; exceptions; and overall project status. This clarifies and demonstrates the need for a supporting architecture to support the operation of the life-cycle. Before presenting how the architecture supports the process based operations/activities the following section reviews the functional scope of the proposed architecture.

The description in the next section assumes a basic understanding of IT technologies. However, in the case of less common technologies, some cross references will be provided for further elucidation should the reader require additional background. The subsequent section relates to the Proposed virtual community centric discourse.

4. Virtual-communities as a mechanisms for coordination support in a distributed environment

This section sets to highlight from literature the properties that should characterize a virtual community centric solution to adequately provide coordination support in a distributed environment. Computerized support for collaboration has been recognized as necessary when the collaborating users are physically distributed [16]. The support technologies aid in facilitating the working together of teams over geographic distances, through the provision of tools that assist communication, coordination and problem solving processes. The following subsections highlight the suitability of virtual communities to support coordination in a distributed environment followed by a revelation of the underlying service subscriptions necessary to ensure its realization. The next section begins by explicating the promise virtual communities hold for coordination.

4.1 Virtual communities for coordination

Distributed teams have often turned to software systems, which incorporate groupware, project management, business process modelling tools, and Workflow Management Systems (WFMS), in an endeavour to fulfil their requirements and to satisfy their coordination needs. These coordination support technologies range from the customarily strictly defined and asynchronously executed business processes, as with most WFMS, extending to those that provide communication and cooperation support for groups dealing with more fluid and ill-structured processes, as in most groupware (e-mail, shared workspaces) or combinations thereof. Virtual communities can account for a larger scale of collaborating partners, contrary to groupware, in which coordination functions and relationships become cumbersome when situations involve a larger number of users, or workflow where coordination is often focused on individually driven activities.

However, virtual communities can be leveraged to account for groupware and workflow functions, commonly at different levels of granularity. Thus, virtual communities can maintain a hybrid state, with the possibility of a mixed composition of both groupware and workflow properties. This scenario has become more common, with the advances in networking technologies and the increasing popularity of the Internet and the World Wide Web [25] which have since formed the basis for sophisticated virtual communities. De Moor and Weigand [14] deem that in an increasingly networked society, with a need for global and flexible ways of professional or social interaction, virtual communities are the natural candidates to fill collaborative gaps in traditional, hierarchical organisations [16]. To support collaboration and to optimise the effective use of limited resources, organisations adopt new organisational structures, which facilitate flexible and decentralised work management. This has led to flatter, coordination-centred organisational forms, such as networked organisations [26] and virtual organisations [27] to account for the business processes that extend across organisational boundaries [28]. This leads to virtual communities being capable of representing flexible networks consisting of independent, globally distributed entities (individuals or institutions) that share knowledge and resources and work towards a common goal.

Holt [29] advocates that coordination technologies should express tasks, their diverse relationships and connexions to each other and the people responsible for them, in a flexible and well-integrated manner, while accounting for unpredictability. Thus, the need for context awareness cannot be over emphasized. Virtual communities exhibit dynamic properties, which include: flexibility, adaptability, scalability, and robustness [15]. They can form, disband, and re-form to meet fluctuating, spontaneous and emerging situations. In addition, they transcend geographic locations and time constraints, thereby enabling anywhere, anytime access. The need to pursue approaches that allow collaborative systems to evolve over time has been suggested [30, 31]. This underscores the need to develop collaborative systems that are capable of co-evolving to meet requirements that cannot be predicted in advance and this issue is explored with virtual communities.

The assumption is, that by leveraging the virtual community properties, seamless and sustainable coordination may be achieved [15]. As an inherently socio-technical environment, a virtual community extends beyond the traditional environment to support distributed interaction; fusing the virtual community with the real world community, as the interaction is still contextualised in reality. Thus, the sub-systems interact and support each other and consistently striving for balance. In order to understand some of the sophistication necessary to adequately account for coordination in a distributed environment, then next section highlights some service criteria necessary to ensure a virtual-community centric realization.

4.2 The underlying Virtual community characteristics

Virtual communities facilitate the rapid exchange of information and knowledge, utilising electronic media to communicate within a shared semantic space [32]. Differences in knowledge can encourage people to communicate, with Sumi and Mase [33] positing that *knowledge-sharing* may facilitate the formation of a virtual community. Essentially, information sharing denotes an awareness of that knowledge. *Awareness information* is considered crucial for the success of cooperative work [34, 12]. For instance, Workspace awareness encompasses the knowledge associated with being present at the physical workspace. By being aware groups are equipped to collaborate effectively. However, as Gutwin and Greenberg [35] observed, too much awareness information can result in an overload. Users thus have trouble *separating useful and unimportant information*, when large amounts of information are presented [36, 37].

It is vital, however, to consider what information users should be made aware of and how they should be made aware of it [16]. This denotes, to prevent excess distribution of data, it is critical only to notify users regarding events deemed relevant, as aligned with the notification means. This results in the principal requirement in dealing with unusual information overload, which may generate requirements for extra effort or mask important information. There is a vital need to *personalise recommendations or feedback*.

To achieve such capabilities, Rocker [36] and Pedersen and Sokoler [38] posits that three complementary strategies should be employed: *information filtering*, progressive levels of information depth or abstraction, and context based information representation. By *combining user centered information with context information* the most relevant or adequate information and services can be delivered or recommended to a user. Also, components or objects can be reconfigured to behave in certain ways, enabling or inhibiting some their actions to meet a given need context. A recommender system provides personalized shared information to a user based on the specific need context. It can deliver different and unique content to different individuals based on their interests [39]. The value of *shared knowledge* before entering into and complying with commitments provokes the need for a recommender module. In addition, through *personalization*, more relevant and thus, valuable throughput can be achieved. By personalizing, environmental or situational information can be captured and filtered according to a specific user's preferences. A class of systems which facilitates *personalised context-driven* responses is called recommender systems which incorporate Amazon and EBay. Here people rate information items they come across, for instance books. These ratings are used to filter incoming information, relative to the specified interest level. Recommender systems are considered efficient tools for overcoming the information overload problem by providing users with the most relevant content, data or ideas, habitually generated through the selected preferences, partialities and stated predilections of individuals and the ratings supplied to choices made by them [40,41]. In an effort to provide users with personalised content, applications employ various techniques, which comprise: content recommendation, content filtering, and preference-driven queries. These techniques exploit different items of knowledge, organised into profiles and contexts [42].

Considering that Personalisation relates to tailoring products and services to better fit the needs of a user, employing various factors inclusive of their preferences, interests, expertise, workloads and tasks, a scalable and dynamic information service delivery system is required [43]. This type of information service which publishes and subscribes systems, connects information providers and consumers through delivering events from several sources to interested users [44]. Where large distributed networks are concerned, a publish/subscribe paradigm is often suggested for information dissemination from the publishers (data/event producers) to the subscribers (data/event consumers) [45]. These publish/subscribe systems have been employed in a variety of applications, ranging from personalised information dissemination to large-scale and critical monitoring. Linda [46] advocates that by introducing a publish/subscribe system information access can be made more efficient, as the paradigm ensures that information is forwarded to the users according to their preferences. In essence, in order to reduce information overload awareness information should be aggregated and/or abstracted into a number of simpler forms or modules Rocker [36].

This allows the logic required to solve the problem to be decomposed into a collection of smaller, related pieces with each piece addressing a specific concern. According to Yoon [46] the design approach for building distributed solution logic often revolves around the "separation of concerns. The service orientation parading can be viewed as a distinct approach through its abstraction capability in which to realize separation of concerns [46, 47] The value provided by a adopting service oriented paradigm include an increase of flexibility, agility, interoperability, federation and better responsiveness to constantly changing business environments. Thus, the service oriented approach provides a mechanism for integrating existing legacy applications regardless of their platform or language [48]. A European one-stop portal termed life-event portals for example offers service integration of distributed government services for citizens [49]. These Life events are composed automatically on the basis of public service descriptions, supplied as Web service that subscribe to the service-oriented computing paradigm. The model proposed in this research embraces as part of its driving principles the concept of Modularization and separation of concerns as the basis of its service orientation. The approach adopted in the research employs the concepts and technologies of Service-Oriented Computing to cope with the issues of distribution, and heterogeneity associated with the environment.

5. The Architecture Design Considerations

The South African public sector exhibits the complexity associated with distributed coordination, which factors include, *inter alia*, size; the governmental structures; the number of role players; and its loosely-coupled work pattern. The magnitude of this complexity imposes a crucial need to develop a coordination support mechanism which encompasses the dynamic collaborative requirements of the public sector. Figure 2 provides an overview of requirements that served as input towards the development and design of both the process model and the supporting architecture to account for coordination in a distributed environment. The motivation latches on the premise that virtual communities offer an ideal platform for collaboration. This is further substantiated by the requirement element, as identified in [18]. How the architecture model meets the requirement is made explicit in Section 5.

Relative to the architecture, in order to leverage the potential benefits of Virtual Communities, a flexible, context sensitive middleware infrastructure, capable of coping with the needs of diverse collaborative scenarios, is stressed.

Its *flexibility* reflects the capability of the envisioned system to rapidly adapt to any of a variety of emerging and evolving behaviours in collaborative organisations; signifying that the proposed model must be open with regard to the integration of technologies and tools, as it must account for existing tools and concurrently, be extensible, in order to accommodate new solution models with minimal difficulty.

DESIGN REQUIREMENT	DESCRIPTION
RQ1: Identify/Match shared interests	Facilitates streamlined and focused collaboration
RQ2: Facilitate contact initiation	Facilitates interaction between possible collaborating entities
RQ3: Components interoperability	Promoting Open systems, Technology/semantic uniformity, Agreement /standardisation towards integration among different representations
RQ4: Real-time federated data analysis and forecasting (predictive/feasibility assessment) for decision making (	Facilitates decision making through streamlined analytics and forecasting
RQ5: Seamless semantic/process/tools integration	Facilitates ubiquitous accessibility of data and transcends beyond problems with exchanging data between applications to semantic integration of understanding those data.
RQ6: Agile process definition/modularisation and configuration	Represent the ability to respond to changes quickly to a given cooperative business process circumstances.
RQ7: Spontaneous communication	Support synchronous/asynchronous discussions and negotiation
RQ8: Support autonomy and loose coupling	Support jurisdictional constraints and desirable preferential connections.
RQ9: Subscription/ Personalised notification and recommendation	Prevent information overload through tailored and streamlined service provision
RQ10: Access control/compliance	Preserve logical autonomy, protect information integrity Clear-cut roles and responsibility domains
RQ11: Augment Shared workspace with Cooperative Object sharing and documentation support.	Asynchronous/synchronous information transfer. Support the realisation that cooperative business processes leads to artefacts (documents, tools) which need to be shared among project community members.
RQ12: Dynamic and adaptive process composition (structured +unstructured) scheduling and execution	The ability to compose services at various levels of granularity, with event-driven and asynchronous styles of interaction that can account for various use scenarios
RQ13: Unified service access point	Single sign on point and access to resources and attain instant visibility into the entire workflow chain via a graphical, user-friendly dashboard.
RQ14: Information diffusion, Context awareness and reporting	From organisational mindfulness, to taking cognisant of objects and their state of affairs in terms of teams and their subsequent activities, resource, and schedules among others, towards informed proactive behaviour.
RQ15: Flexibly/Scalable/extensible/reusable/ distributable	Accounting for a greater degree of variability to support varying scenarios regardless of context + individual participation in shared processes regardless of location using smart endpoints.
RQ16:Knowledge base support, Content management (Smart Archiving/Knowledge sharing)	Managing and storing information, tailoring of content and advertising to a user's specific characteristics based on user information + support and augment, repository with semantic/ontology based indexing, search and retrieval features.
RQ17: Support for usability with User interface adaptation	Allow user interfaces to adapt to various contexts and thereby enabling a flexible and multi-purpose environment.
RQ18: Adaptive/Ad-hoc group formation (structure)	Support dynamic formations of groups to augment governance models and clearly defined policies.
RQ19: Automation and customization	Support levels of customisation and process automation to streamline accelerate and standardise processes (e.g. complex procurement/deployment procedures).
RQ20: Dynamic object administration, tracking and configuration	The ability to design and document goals and administer objects through specifications, monitoring and evaluation, and rules to guide behaviour.

Fig. 2 Abstracted Requirements Summary [18]

To encompass the complexity associated with coordination in a distributed environment, a loosely-coupled approach is employed, taking into account the separation of concerns through modularisation. The principle of loose coupling makes applications more flexible, with increased and easier adaptability and greater responsiveness to changes. The loosely-coupled pattern reflects the modularity principle, indicative of the modular separation of concerns. The modularisation principle focuses on the decomposition of the design into individual functional or logical components, which reveal well-defined communication interfaces containing methods, events, and properties. Functions are partitioned into discrete, scalable, reusable modules, consisting of isolated, self-contained functional elements, with well-defined modular interfaces, to facilitate the interaction necessary to meet certain task objectives. Therefore, components can be deployed into any appropriate environment without affecting other elements or systems. By applying the principles of composability and loose-coupling it is ensured that the design can be *configurable* or re-configurable (to meet varying needs), denoting that it can be highly adaptive and extensible. Components should be designed to be reused in different scenarios with diverse applications. These elements should be capable of being readily substituted with other similar components, thus replaceable and extensible from existing components, in order to provide new behaviours.

Furthermore, the envisioned support infrastructure must be distributable, providing standard procedures or processes for invoking functionality remotely across different platforms, used by several people in different, dispersed locations. It must be scalable, to accommodate growth. It must possess the capabilities and capacities to cope and perform at an increased or expanding participation level or in cases of larger operational demand and, additionally, must be context sensitive. Context-awareness is imperative in a situation where the operating environs are constantly changing. In order to recognise, react rapidly and cope with the unpredictable changes in the environment, the envisioned infrastructure must account for context. Context, principally, refers to all types of information pertaining to a service and/or to the user of the service. The proposed model must take advantage of context information to provide services that will aid in the coordination of collaborative activities, from the recommendation to the execution and monitoring of tasks.

6. The Functional Scope of the Model Architecture

This section overviews a loosely-coupled virtual community-based architecture, designed to host the collaboration life-cycle operations; thereby meeting the coordination needs of a distributed environment. The approach combines and benefits from various technologies, methodologies and tactics, based on certain principles as described in section above. The proposed model architecture employs a design pattern that divides a complex whole into smaller, simpler parts, with the aim of reducing the system complexity. Thus, four fundamental functions of the supporting architecture are introduced and discussed, as they would support the operation and activities of the proposed collaboration life-cycle model geared towards streamlining coordination in a distributed environment.

The underlying design concepts are inspired by the shared workspace, the service-oriented computing, personalisation, and publish/subscribe paradigms, highlighted in section 4 in order to attain the level of support necessary to sustain and reinforce coordination in a dynamic dispersed environment. The proposed model architecture decisively builds on these concepts, as described in William, [50], Kirsch-Pinheiro et al., [34], Maréchaux, [51]; Papazoglou et al., [52], Rocker [36], Linda, [43], Shen, [45], Huang and Garcia-Molina [44], which inform the model architecture design and satisfy several important identified requirements. Zager [53] describes a '*model*' as the amalgamation of certain organising principles, the structure of a system and the elements/objects making up a system. As part of the design process, a pattern towards producing a suitable solution is employed. The architectural perspective employs general design principles, which suggest grouping common things together to effect easier understanding and management. Therefore, considering the solution in this manner suggests several possible ways of contemplating and developing the architecture. For convenience and logic the following choices have been selected.

Since only certain aspects of the virtual community deal directly with the user, such functions will be grouped under 'front-end services'. The specifics of managing the environment fit together as management services, but two sets are distinguishable. These two sets include those that deal with design and configuration aspects, grouped under 'object management and configuration' and those that deal with the more dynamic runtime aspects, grouped under the 'execution and monitoring module'. Furthermore, as certain functions are primarily central or core to the system, they are grouped under 'virtual community infrastructure services'. Essentially, the model uses a hierarchical design pattern, in which the complex whole is divided into smaller and simpler parts, aimed at reducing the complexity of the system. Figure 3 depicts the architecture model, with four hierarchal components, consisting of several mechanisms that will actually do the work. Each of these functions plays an equally important role in the architecture, together representing the solution to the research problem. The following division introduces and motivates these components, discussed as sub-sections.

IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 5 Issue 7, July 2018
ISSN (Online) 2348 – 7968
www.ijiset.com

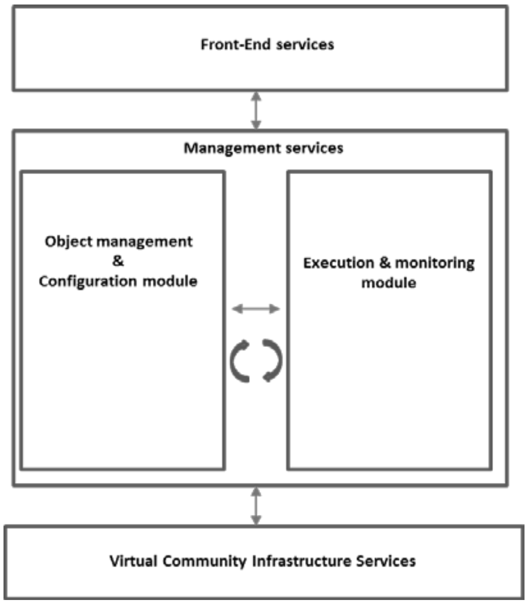

Fig.3 Virtual Community-Centric Coordination Support Architecture Overview

6.1 The Front-End Service

The Front-End Service incorporates the user interface employed to collaborate with others utilising the collaboration support services. It provides a portal service, used simultaneously to access several services. The portal aggregates service programmes as portlets, which are accessible from a single Web-based interface. It accounts for subscription and notification capabilities, as well as facilitating personalisation through content customisation. This component delineates RQ1, 2, 5, 8, 13, 14, 15, 16, 17, 18, 19 and 20. Considering the requirements RQ13 and RQ17 from Table 5.4, the **Portal** collects, organises and distributes information, representing the focal points for information and knowledge exchange. RQ5, 6 and 15 accentuate seamless tools integration, ubiquitous, modular and unified data accessibility, realisable with web portal technology. The next sub-section briefly explores the importance of the portal components, indicating how they satisfy the requirements laid out in figure 2, considering certain sub- components.

6.1.1 Status Monitoring/Reporting

RQ1 and RQ2, RQ14 described in figure 2 emphasise the finding of shared interests, facilitating contact and reporting on environmental context. To facilitate any collaboration project requires initially monitoring and reporting environmental changes, regarding commonalities and interests between potentially participating community members. Thereafter, relative to an established collaborative project, the monitoring and reporting are essential in ensuring that projects are implemented as efficiently and effectively as possible. The continuous process of assessing and reporting on the status of project implementation, in relation to the approved work plan and budget, is made possible by this component, allowing timeous adjustment if required.

6.1.2 Profile Organiser

RQ1 and 2 described in figure 1 initially require establishing how individuals can be contacted, utilising several channels of communication to connect with potential collaborators. Several need profiles can be created, aided with context information filters. Members of interest in the community can be added, organised and categorised within several profiles to easily manage contacts and means of communication. These administrate and monitor potential member collaborators, and non-experts and specialist of interest.

6.1.3 Design/Specification Tools

To mitigate poor design, based on unclear assumptions, in order to improve successful project execution, where even the best monitoring is unlikely to ensure its success, RQ, 18, 19, 20 described in figure 1 emphasise a realties design consistent with its input, process and output. This facilitates traceability, through the systematic documentation of what is required to be accomplished or transformed. Explicit information on how activities, artefacts and resources are put together through modelling, for instance, with other means of configuration and specification employed towards a collaborative end.

6.1.4 Visualisation Tools

RQ3, RQ 4 and RQ 14 described in figure 1 denote access to distributed data, which is aggregately displayed as if from a central source in a meaningful way. Data visualisation tools will aid in identifying trends and patterns utilising graphs, when considering a potentially large volume of information that needs to be graphically aggregated to provide value. These tools are exploited to support evaluation and forecasting, which tracks outputs and measures contributions to results by assessing changes from established baseline conditions or metric.

6.2 The Object Management and Configuration Module

The Object Management and Configuration Module account for both object and activity level coordination and comprises administration and design features. These administration and design features serve to ensure that the coordination requirements of a collaborative effort, or an opportunity to collaborate, can be enforced effectively at runtime. The module serves as a source of context and provides input for the Execution and Monitoring process.

6.2.1 Process Manager

RQ, 5, 6, 8, 12, 15 and 20, portrayed in figure 2, denote the need to provide order to work activities over time. Using design/speciation tools it can manage and utilise both fixed and adaptive processes, to form a collaborative workflow that can benefit from dynamic composition. This, in partnership with workspaces, assists with basic *project management* activities, including task management; calendaring; workflow planning and routing; and time tracking.

6.2.2 The Community Manager

The community manager reflects RQ1, 10, 15 and 18, defined in figure 2 to provide support for the management of users and groups, their registration, profiling and preference specifications. In relation to workspace, it supports Team Definition; the User Profiles of participants; Social Presence Management; Contact Management; and Access Control definition.

6.2.3 The Workspace Manager

The workspace manager supports RQ 6, 7, 8, 11, 17 and 20, described in figure 2, which essentially suggest the administration, configuration, distribution and monitoring of work. It manages the virtual office space; supports the co-creation of goals in real-time or asynchronously; facilitates consensus building through group discussions and polling; while uploading and sharing files. It accommodates a wide variety of activities and behaviours that are not predefined, but can be considered as unique cases unfold.

6.2.4 The Resource Manager

The resource manager denotes RQ4, 5, 17 18 and 20, explained in figure 2, by improving operational efficiency; knowledge optimisation; and improvement. It provides data aggregation and virtualisation capabilities that facilitate new methods of searching, correlating and analysing data from various sources, while taking cognisance of existing resources, their configurations, versions and availability to support decision making.

6.3 The Execution and Monitoring Module

The Execution and Monitoring Module comprises runtime enforcement service characteristics. It receives input from the Object Management and Configuration Module, for instance, serving as context sources. It handles context information, in conjunction with managing action invocation and personalised notification and recommendations to applications or users. The module is responsible for shared workspace, as well as the social and task awareness that informs participants as to the state of affairs, founded on specified policies. It comprises the context manager; the awareness and recommendation manager; and the runtime manager, who work together, undertaking decisions on actions to be executed and/or whether to send notifications to participants.

6.3.1 The Context Manager

The context manager handles context based information to support RQ1, 2, 9, 11, 12, 14, 16, 17, 18 and 19, described in figure 2. Thus, it facilities the shared interest matching of collaborators, artefacts and tools, to populate a shared workspace, enable awareness-based operation, communication and process instance definition. In addition, it enables streamlined subscription and personalised notifications/recommendations, to prevent information overload and unnecessary intrusion through customised service provision.

6.3.2 The Runtime Manager

The runtime manager function coordinates actions and provides awareness information in a shared workspace, while executing multiple process instances, representing RQ6, 9 and 20, expressed in figure 2. It manages sessions and support service invocation, composition, personalisation and scheduling. It essentially orchestrates the actions of other components.

6.3.3 The Recommender/Awareness Service Manager

The recommender and awareness service manager provides personalised notification and recommendations services, accounting for RQ1, 2, 9, 14 and 16, defined in figure 2. Opportunities for collaboration are identified and recommended, with progress continually monitored during execution. Additionally, it accentuates streamlined information diffusion and content management.

6.4 The Virtual Community Infrastructure Service

The Virtual Community Infrastructure Service function accounts for loosely-coupled integration and communication between interacting components and applications. It employs a service-oriented computing pattern to seamlessly integrate collaborative service applications. The need for ontologies is emphasised, to ensure that applications can understand and interpret the information they access. This transcends simple system interoperation to semantic integration, in order for disparate systems to be able to gain and maintain the same understanding of any particular set of data or its representation. The virtual infrastructure advocates the use of ontologies to achieve interoperability at the semantic/process level, in pursuit of seamless data integration. This level is targeted towards accounting for data, process, application/portal and sematic level integration. The level adopts a service-oriented computing model to seamlessly integrate applications and share or reuse generic applications. Furthermore, the infrastructure advocates the publish/subscribe communication paradigm - a message-oriented middleware that promises synchronisation between users. The sub-components of this intermediary service are discussed in the sub-sections.

6.4.1 The Communication Component

The communication component facilitates real-time and asynchronous text, voice, and video communication, based on context. In addition, it supports query/notification management services, denoting RQ 2, 7, 9 11 and 16, expressed in figure 2.

6.4.2 The Integration Broker

The integration broker facilitates the dynamic process, application, data, and infrastructure integration services. It promotes an open system by facilitating interoperability at different levels of granularity**.** It emphasises registration**,** discovery and dynamic service composition, accounting for RQ3, 5, 8, 12, 15, 16 and 17, explained in figure 2.

6.4.3 Repositories

Repositories represent accessible**,** scalable, flexible service and context information storage services. This tier employs the use of ontologies to achieve deeper interoperability integration, from the data to the process level. It focuses on the use of ontology to solve issues of semantic heterogeneity, providing a rich, predefined vocabulary; thus enabling interoperability and representing RQ3, 4, 5, 15 and 16, described in figure 2.

6.4.4 The Security Component

The security module is employed for access control and authentication purposes. It accounts for access to common information spaces, whether central or distributed, along with work space tools or resources. It is intended to preserve logical autonomy, ensure clear cut roles, responsibility, streamlined service provision and general compliance to policies, underscoring RQ4, 8, 9, 10 and 14, designated in figure 2.

The proposed model architecture assumes that by providing a flexible and agile infrastructure, enhanced by context-aware middleware services, through a one-stop access point (portal), will allow the realisation of coordination benefits. The model architecture encompasses a flexible infrastructure that hosts modular component services, which should interoperate to provide a seamless coordination support service. A detailed depiction of how the components fit together is provided in Figure 4.

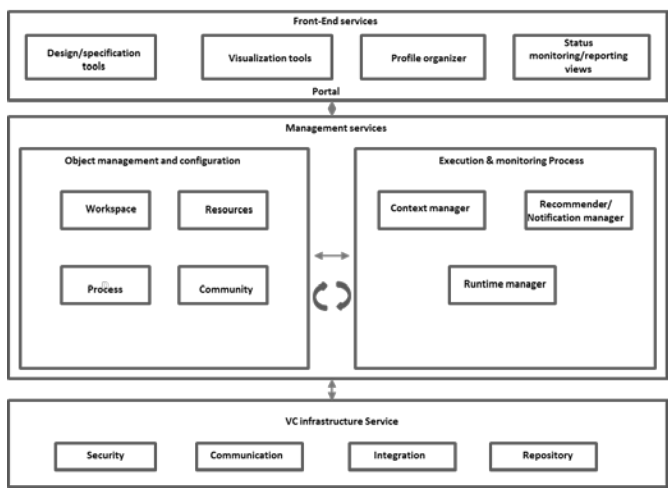

Fig. 4: Architecture Model Summary

7. The Architecture Components and CCLCM Functions Mapping from Awareness Lens Perspective

Awareness represents an important construct in the CSCW research [16], as for instance to assist people move between individual and shared work spaces thus, providing a context which can be used to interpret other's expressions, allow anticipation of others actions which in turn reduces the effort needed to coordinate tasks and resource. This section illustrates how the components of the proposed architecture can support the CCLCM phases and actions, relative to awareness levels of what is necessary before, during and after collaboration to streamline coordination and to promote sustainability. The CCLCM actions presented aided in the summative evaluation of the CCLCM model activities as detailed in Thomas [54]. The discovery of opportunities for collaboration from the environment initiates the process. Fundamentally, the environment is scanned for needs similarities and recommendations are then made to entities for collaboration. The CCLCM, as mentioned, consists of the initiation, planning and design, implementation and assessment stages of a collaboration process.

The needs awareness that eventually results in opportunities for collaboration must be realised prior to the initiation phase. Awareness denotes that if the proper information concerning what other people are doing is sent at the correct time, to the right people, coordination can be facilitated effectively. The decision to integrate awareness as a core aspect of the solution platform is reflected in the collaboration process life-cycle, with the most significant feature comprising the monitoring and reporting cycle. The cycle reflects the three different stages of awareness information required, viz. pre-awareness, necessary before a collaborative project can be initiated; in-awareness, which occurs after the project is initiated; and post-awareness, which follows once a project is completed. The monitoring and reporting component, in the middle of Figure 1, indicates the continuous awareness capability that drives, manages and improves the collaboration process. The current or contemporary insight that collaborators have concerning occurrences within the community and task environment, the comprehension of its meaning, and an understanding of its future implications regarding the task, are all necessary for coordination.

Essentially, team awareness of a certain situation, at any particular point in time, can yield collaborative success. Streamlined awareness may be achieved by engaging architectural components, for instance the community, context and recommender/notification managers. The following sub-sections illustrate how the architectural components collate or systematise information to provide such support. Examples of team awareness encompass knowing about an upcoming deliverable deadline or knowledge of the progress status of a particular project. To aid in comprehending the operation and functionalities of the model, the architectural components involved are discussed relative to the aforementioned levels of awareness. The discussion highlights the type of services to be expected, with the succeeding division focussing on the form of awareness necessary to instigate a collaborative activity. How the architecture components services can support the CCLCM phases and possible actions is encapsulated in figure 5, which illustrates the possible mappings.

	Architecture components: Front End Services				Object management & Configuration				Execution & Monitoring			VC Infrastructure service			
Collaboration life-cycle model phases and possible actions	Design/specification tools	Visualization tools	Profile manager	Status monitoring & Reporting	Workspace manager	Resource manager	Process manager	Community manager	Context manager	Recommender/notification manager	Runtime manager	Security	communication	intergration	Repository
The Pre-awareness level															
Phase1															
1.1 Subscribe to a community platform that identifies opportunity for collaboration			x					x			x	x	x		x
1.2 Provide user with a configuration flexibility to specify interest and preferred means of notification is necessary.			x					x	x						x
1.3 The ability to use different communication notification means as desired is useful			x	x									x		
1.4 The ability to able to use different communication tools for contact purposes is desirable.			x	x									x		
1.5 Providing a shared space to discuss, goals and objectives is useful	x				x	x							x	x	x
1.6 There is a need for a situation based recommendation that provides the required information as required			x	x					x	x			x		
1.7 support for social interaction is desirable					x								x		
1.8 The ability to prioritize is desirable			x		x			x			x				
1.9 Ensuring a common language is desirable	x													x	x
The In-awareness level															
Phase 2:															
2.1 The ability to dynamically manage projects and tasks is desirable	x	x			x										
2.2 There is a need to create and reuse templates	x				x				x	x	x				x
2.3 There is a need to define groups/teams			x		x			x							
2.4 There is a need to dynamically add/remove/invite participants and groups	x				x			x							x
2.5 There is a need to restrict participants options												x			
2.6 The ability to design a decision making, communication and reporting structure is desirable	x			x	x										x
2.7 The ability to detect resource allocation and load balancing conflict is desirable		x										x			x
2.8 There is a need to dynamically create a group or team specific shared workspace			x		x										x
2.9 The flexibility to configure required tools before and after the creation of a workspace is desirable	x				x	x									x
2.10 It is desirable the participants configure suitable notification means depending on several possible situations	x		x		x				x	x			x		
2.11 Secure notification is desirable				x								x			
2.12 It is desirable to schedule and organize tasks in the order of occurrence	x						x								x
2.13 It is desirable to have multiple views of tasks and resources configuration	x	x			x	x									x
2.14 It is useful to have a global workflow process view and private workflow process view	x	x													x
2.15 It desirable to support both structured workflows and support ad-hoc intervention for unstructured tasks					x		x								
2.16 It is desirable to flexibly address all types of processes to support all forms of work predefined or dynamic	x						x								x
2.17 The ability to divide , assign and trace task is desirable					x				x		x				
Phase 3															
3.1 The ability to describe tasks, procedural steps, organizations or people involved, required input and output information, and tools needed for each step in a business process is desirable		x					x				x				x
3.2 The ability to be able to reorganize or rearrange task during execution as situation changes is desirable	x	x					x				x				
3.3 Automating tasks is desirable in some cases															
3.4 workflow automation and components such as online forms to manage and enforce the consistent handling of work is desirable in some cases.							x				x	x			x
3.5 Automating the approval steps a form has to go through is desirable							x				x				x
3.6 workflow automation that can dynamically arrange the services required based on set rules is desirable	x						x							x	x
3.7 The ability to track status, and know quickly who is doing what, the associated resources and tools is desirable.				x					x	x	x		x		x
3.8 A shared place where you can see everything your team has left to work on, or has worked on is desirable.				x	x										
3.9 The ability to dynamically contact personnel during execution desirable									x	x			x		x
3.10 The Task and Workflow Execution detailed information in a calendar format with dynamic timeline views is desirable				x	x					x					x
3.11 The Task/Workflow execution successes and failures color coding are desirable				x											x
Post-awareness level															
Phase 4															
4.1 Dynamic archiving of information for subsequent analytics is desirable		x				x									x
4.2 Dynamic, intelligent integration of distributed data for analysis is desirable		x									x			x	x
4.3 Dynamic integration or search for suitable analytic tools is desirable the flexiblity to add and remove tools needed for analysis is relevant	x					x									x
4.4 multi reporting and analytic views is desirable				x		x									x
4.5 Dynamic resource cataloguing /information aggregation is desirable		x				x									x
4.6 Knowledge management actions, such as searching, categorizing, and storing information is desirable	x	x				x									x

Fig.5 Architecture components and CCLCM operations mapping

7.1 The Pre-Awareness Level

This section considers the context-aware matchmaking service capability, which is intended to unite possible collaborators according to the needs requirement in the environment. This phase allows for the identification of potential members to the communities who have specified interests, and facilitates the easy identification of others who might share an interest in collaboration. This is courtesy of the preferences enabled by the **profile manager**, which establishes a collaborator identity prior to joining a **community** of members that may have a shared interest. The **security** specification emphasised is centred

ISSN (Online) 2348 – 7968
www.ijiset.com

on the established reference identity to ensure controlled access to resources.

Awareness of the environmental requirement is significant at this level. It reflects a needs assessment to provide the basis of identifying collaborative opportunities at the initiation phase. This is undertaken and occurs through intelligently sifting through the dynamic distributed information sources, which characterise several need scenarios. A virtual community should support and facilitate the identification and selection of potential collaboration partners. This connotes that the envisaged platform must find individuals with the correct skills, willing to collaborate and to exchange information. This tier leverages the recommendation service, according to the defined interest of a user and informed on specified notification mechanisms. The virtual community platform serves as a medium for initiating contact with known or unknown collaborators who share similar interests and preferences. At this stage the awareness component facilitates contact building for future cooperation towards a shared objective by taking advantage of the **communication component**.

The pre-awareness level encapsulates the necessary awareness information, which stimulates the opportunity to collaborate. This opportunity for identification is aided and made visible by a variety of **status monitoring and reporting views** as part of the front-end service. The services of interest, functional at this level, include recommendation and contact management services. In accordance with the context information offered by the **context manager** and the user-defined profile from the **repository**, the context aware application (**recommender and notification service system)** provides recommendations of interest to the user. There is a connexion between the needs (or preferences) of the user and the content delivered, allowing resources to be tailored for the user in a personalised way such that only the subset of information that is most relevant to the team members will be presented. Thus, by employing intelligent information sharing techniques that can identify information that is most relevant to each potential collaborating member, better coordination can be realized. Additionally, a recommendation should be offered to the user through a usable and accessible user interface. Figure 6 illustrates how recommendations are achieved based on context information. For instance, given the South African training need context, two or more municipalities who are registered on such a platform can be brought together based on their common training interest so they can collaborate and maximize the resource use. Essentially, instead of separately employing different training providers for the training provision they can pool their resources together and have a single training session shared by all participating municipalities. Thus, maximizing resource use through the economies of scale, facilitating knowledge sharing and ensuring common knowledge base across the municipalities. Essentially, in an ideal scenario, Municipality A will have an interest or need in very specific training for their staff. Also Municipality B may also have interest in the same specific training for their staff. It would therefore only make sense for them to combine efforts and engage one training provider. By doing that, they will not duplicate efforts nor waste money among other things. However given the lack of coordination among the municipalities such opportunities for collaboration are hidden such that each municipality engages their own training providers resulting in inefficiency and waste of resource among other things. Thus, the objective of the pre awareness is to perform activities that will make visible opportunities to collaborate.

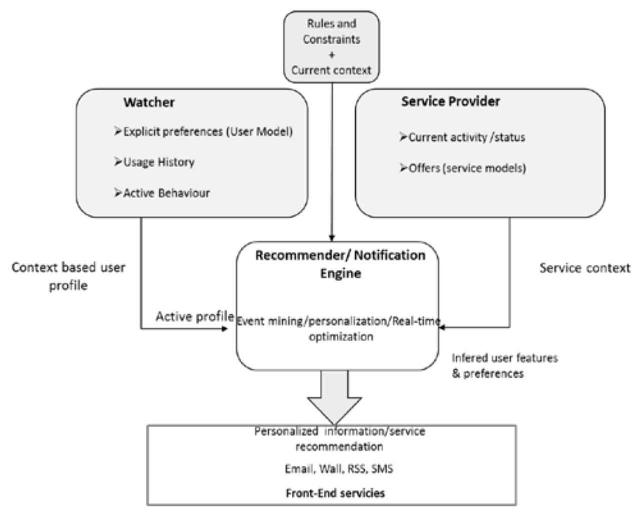

Fig. 6 Recommendation Process

Presence or awareness information concerning the state of members (availability or absence) may be required to initiate a conversation with a potential candidate. Informal awareness is necessary, as is support for direct communication, which can be realised with standard **synchronous and asynchronous** methods of computer and network-based communication, viz. telephone calls; video and audio conferences; text talk; and email or news. Informal awareness is a general sense of community members, concerning what individuals are doing, as opposed to what team members are doing - central to the subsequent sub-section. Awareness facilitates casual interaction, which may form the backbone of everyday coordination and work.

7.2 The In-Awareness Level

This is the predominant awareness level and is essential when members agree to collaborate. The **planning and design phase**, as well as the *implementation phase*, are the focus of this section. They employ the services of the **workspace manager**, which creates a shared workspace for discussion and for establishing the agreement of objectives, roles and responsibilities in relation the engaged collaborative project. Furthermore, the front-end service **design and specification tool** and the **process manager** assist in defining the adaptive workflow reference template or process model to guide the actions of the **runtime manager** at implementation. The services of interest constitute the task and process management; in-awareness advocates process awareness, which involves, *inter alia*, context information pertaining to process instances; the team configuration (i.e. participants and their roles); and their associated artefacts and tasks.

Work lists generated to fulfil the collaborative activity are produced for each participant. This is followed by their arrangement, according to scheduling and the order of execution, within an overall global process. In situations where templates exist, a work list is introduced from a **knowledge repository**. Activities may then be selected for a specific business. These activities or tasks are presented in the form of services. Keeping track of the activity processes and execution engenders the need for awareness: knowing who is responsible for what, and when, to send alerts or reminders. This form of notification is necessary during the course of execution.

Establishing an opportunity to collaborate results in contact initiation and the creation of a **workspace**, where collaborating members conduct planning and design. There are certain steps, with the initial building of a common understanding, followed by the identification of a goal, in conjunction with the manner by which the objective should be attained. The execution of individual work, and communication between co-workers in order to coordinate activities and work plans, is necessary. However, where there is no abstract work model detailing the steps necessary for performing a task, the system must offer as much flexibility as possible to team members, enabling them to conduct what measures they deem necessary to achieve a particular goal. This requires a high degree of group awareness, with co-workers aware of the history of each other, and their current and potential future activities within the shared environment. The propagation and exchange of group awareness information results in the implicit coordination of team work. Users may have their personalised workflow defined, as it relates to the general collaborative objective.

At this level, process awareness is critical. When collaborators agree to work together towards a shared goal, they generally agree on certain well-defined tasks, which are more or less formalised. Members require knowledge pertaining to the multiple relationships between the artefacts and the context in which they were created, shared, and distributed (i.e. who, what, when and in which context), making organisational awareness (e.g. roles) relevant. The systems facilitate the mobility of content and the context of activities in the business process to group members. Essentially, they provide information concerning process instances, the team configuration (i.e. participants and their roles), their associated artefacts, and connectivity modes of group members. A project manager and other relevant stakeholders may need to be informed constantly regarding all work activities and status information. Overall, members typically require status information relating to all work activities performed by other team members in a joint project (process-awareness).

The support for process variety and adaptive workflow modelling and composition is also emphasised at this level, aided by the **process manger**. It should be possible for a virtual team to initiate an ad-hoc process and, from any particular activity, to link it back to the defined process model. Additionally, the system should also allow starting from a process template, as well as to permit deviation, for example, through simply deleting activities modelled in the process template or by adding new activities from a given task library. With an event-driven service design, dynamic composition and integration of heterogeneous services can be achieved. Business processes are dynamic in nature owing to changes and alterations in polices, rules, partners, and events. An event-driven service-oriented architecture should be capable of providing seamless integration; the automation of business processes; support for state management; transaction and notification; and services monitoring execution. The event-driven automation of business processes allows service provision regarding reactions to events, which activate according to defined rules or configurations. For example, based on the specification of a training type (online or physical), a given training process template can be activated.

At the 'in-awareness' level the proposed platform presents a way to manage the flow of activities or events, which are passed to appropriate partners for service provision. The platform proposes the use of dynamic compilation service-based modules that support workflow management. As a user-oriented design, users may customise their services as required.

The dynamic compilations are carried out as a back-end operations, as instigated by the **integration broker** and the event monitor. This is aided by the **design/specification tools.** In the front-end, users can complete their workflow requirements through interaction with their respective portal applications. The operation, therefore, consists of the three layers, viz. the data layer (database servers), the business process layer (service driven composition) and the presentation layer (aggregated portal). The service broker should select, compute and determine the sequences of the tasks to be conducted and, based on the business logics, a suitable activity schedule for participants can be generated and presented as a web application. Thereafter, participants may adhere to their work schedule, in accordance with the correct sequence, to complete the assigned work. The presentation layer represents a portal interface layer, which provides the facilities to construct and customise user interfaces, suitable for each level of users. The dynamically compiled work schedule is communicated to, and interacted with, by the users, who can observe changes to their portal pages via the media. Figure 6 depicts the functional relationships of the service elements, ranging through the profession or business profile, function or activity, and function or activity design and operation services.

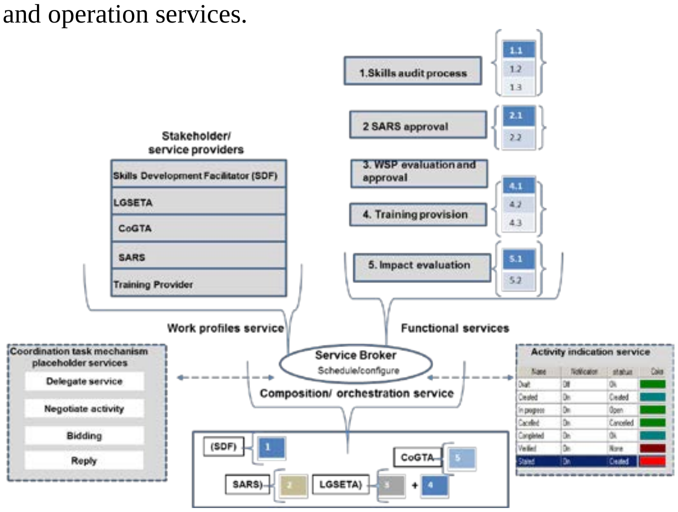

Fig. 6 Dynamic Process Configurations

The profession or work profile service contains all potential job descriptions, as well as the service briefs of registered stakeholders. For instance figure 5 top left consist of a list of stakeholders in the South African public sector who one way or the other contribute the skills development of the municipalities in the country. The function or activity service encompasses the required service activities, which define all business processes and their function tasks. For example as shown in figure 5 the functional service tagged '(1) A skills audit process' often performed by the 'skills Development facilitator (SDF)' is a functional service that consist of three or more sub-functional or activity services which include: (1.1) Identifying opportunity for training or skills development; (1.2) Interview line managers to determine level of training; (1.3) Set skills development objectives, etc. The service design or composition should comply with the work practice requirements during the design-build-store process. Essentially, during a new business process composition or orchestration, function services are arranged in the order to be executed from the database, the role players are assigned to the tasks and the coordination mechanisms to be employed the various stages. For instance the SDF stakeholder will handle functional service '1 Skills audit process', followed by '2 approval activity' by the SARS stakeholder, then '3' an evaluation and approval by the LGSETA, followed by '4 training provision activity' which may employ a coordination mechanism of tender bidding from external training providers or simply delegate to an in-house trainer. The activity indicator service is employed to indicate a composed task/activity status quo during the course of execution whether it has begun, it is ongoing, halted or cancelled. Finally '5 impact evaluation activity' undertaken by the CoGTA stakeholder to measure whether the intervention was properly carried and projected impact realized or not.

At run-time the project manager can assign new tasks or modify previous assignments whenever necessary. The platform should recompile the changes into the best possible schedule, according to the work functions required, assigning newly arranged customised tasks to a role player. Templates can be designed and used for routine assignments, which provide standard sets of services and work specifications. Participants are afforded full control in customising the workflow schedules and the deliverables required. Through monitoring and evaluation progress can be reviewed, and problems in planning and execution can be identified and adjustments made.

7.3 Post-Awareness Level

This level reiterates the need to monitor and to evaluate the quality and impact of work, in accordance with the agreed collaborative objective. This level highlights the requirements of the assessment phase, as illustrated in Figure 1. While tracking project status the supporting of artefacts is important, to ensure that the planned execution is occurring correctly. The evaluation resulting at the end of project implementation, determining the impact of the project on the shared cooperative goal, is also significant. This necessitates familiarity with the circumstances of the beneficiaries prior to a project being implemented, providing the baseline data, which is collected during the needs or requirement assessment before project initiation. This information allows the assessment of improvements instigated by the project implementation over time. An impact assessment informs the efficacy of an intervention, whether it has made a difference to the problem situation that was being addressed. Through comparing data describing the situation before an intervention was initiated and information generated after completion of the intervention project, changes in the circumstances of the beneficiaries can be measured. Furthermore, lessons may be drawn from the changes linked to the implementation of the project, towards facilitating other collaborative projects; the identification of inconsistencies and missed feedback loops, as well as other factors and issues. This emphasises the advantages of having access to repositories and tools for intelligent analysis to support decision making.

Performing an intelligence analysis requires access to tools and applications that perform knowledge mining, for analysis and decision-support sessions. This feature engenders the defining of new opportunities. The capabilities of the **resource manager** functions, to support data virtualisation and analytics are relevant. The seamless integration of analytics and **visualisation tools** to support the requests of various managers is emphasised. This level aims to facilitate new methods of correlating and analysing data, in an easy and understandable manner, while providing the ability to make decisions, as well as predicting future interventions and the resources required. Further details of this type of integration and virtualisation can be seen in Thomas and Botha [55]. By employing a service approach, the solution allows organisations to use whatever data sources they may require, ensuring that services can encapsulate several, assorted data sources.

7. Conclusion

This paper presents a conceptual overview of a prescriptive virtual community centric architecture geared towards addressing the problem of coordination in a distributed environment. The model is prescriptive, in that it defines the core features and functionality, from which an implementation can be developed. The core is drawn from previous research and relevant technology architectures. The characteristics that influence the design of the model are derived through analysis of the distributed environment. The correlation and combination of proven theories and existing technologies, allows the achievement of a unique combination that promises to facilitate coordination a large scale distributed environment like the SA public sector.

The architecture presented has the potential to host the dynamic collaboration life-cycle model operations, which engender the support and promotion of sustainable coordination in the distributed South African public sector. The discourse presents a high-level look at the fundamental functions of the model, its constructs and their relationships. Founded on service-oriented computing principles, the architecture makes it possible for organisations to develop coordination support systems through the utilisation of visualisation tools, databases and other system components, which these entities may currently employ, that are open source or are affordable, prompting re-use. The architecture advocates the use of loosely-coupled services and context information, to make applications more flexible, while allowing for quicker responses to environmental changes. Thus, tracking awareness information prior to and subsequent to the initiation of projects ensures that coordination relative to the collaboration life-cycle may be better streamlined.

The architecture essentially advocates a distributed context-ware service system to support coordination in a distributed environment. The attributes of the architecture promise to support and promote coordination in a heterogeneous and distributed environment. However, the limited scope of the architecture evaluation suggests a limitation. Increasing the evaluation scope of the artefact towards generalisation is necessary. Thus, future work should see the application of the model through the implementation of an actual support system and would also assist in exploring and understanding the technicalities involved, which could perhaps lead to refinement of the architecture. For instance, the virtual community-centric model could be leveraged, extending the shared services concept through taking advantage of cloud computing services, transcending infrastructure and distance concerns.

References

[1] A. Espinosa, S. A. Slaughter, J. D. Herbsleb, and R. E Kraut: "Team Knowledge and Coordination in Geographically Distributed Software Development", Journal of Management Information Systems, Vol. No 1, 2007, pp.135–169.

[2] C. Cramton: "The mutual knowledge problem and it consequences for dispersed collaboration", Organization Science. Vol. 12, 2001, pp.346–371.

[3] O. Amir, "Intelligent Information Sharing to Support Loosely-Coupled Teamwork", PhD thesis, Harvard University, Massachusetts, Cambridge, USA, pp. 3, 46, 2016.

[4] P. Hinds, & C. McGrath, "Structures that Work: Social Structure, Work Structure and Coordination Ease in Geographically Distributed Teams", Proceedings of the 2006 Conference on Computer Supported Cooperative Work, Banff, Alberta Canada, November, 2006.

[5] N. D. Mehandjiev, A. A. Karageorgos, and G. Tsang (2003), "Designing Coordination Systems for Distributed Teamwork", Proceedings of the Twelfth International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises pp. 117, Washington: IEEE Computer Society.

[6] V. Acha, and L. Cusmano,: "Governance and co-ordination of distributed innovation processes: patterns of R&D co-operation in the upstream petroleum industry", Economics of Innovation and New Technology, Taylor & Francis Journals, vol. 14 No. 1-2, pp.1-21, 2005.

[7] S. Kiesler, and J. N. Cummings: "What do we know about proximity in work groups? A legacy of research on physical distance. In P. Hinds & S. Kiesler (Eds.) Distributed Work, pp. 57-80. MIT Press, 2002.

[8] H.J. Happel: "Towards need-driven knowledge sharing in distributed teams", In Proceedings of the 9th International Conference on Knowledge Management, pp.128–139, JUCS, October, 2009.

[9] G.M. Olson, and J.S. Olson: "Distance matters", Human-Computer Interaction, Vol. 15 No. 2/3, pp.139–178, 2000.

[10] NCBF: "National Capacity Building Framework for Local Government", Department of Provincial and local government, 2004-2008, 2012-2016, Pretoria.

[11] The South African Government. Constitution of the Republic of South Africa No. 108 of 1996, Retrieved July 6, 2011, from Statutes Of The Republic Of South Africa-Constitutional Law, February , 1997, sourced from, http://www.gov.za/documents/constitution/1996/a108-96.pdf

[12] K. Schmidt, D. Randall, "Preface to the Special Issue on 'Reconsidering Awareness in CSCW". Computer Supported Cooperative Work (CSCW), 25, pp. 229-233, 2016. doi:DOI 10.1007/s10606-016-9257-6

[13] H. Tellioglu: "Collaboration Life Cycle", Proceedings of the 2008 International Symposium on Collaborative Technologies and Systems, Irvine, California, USA, pp.121-126. May, 2008.

[14] A. de Moor, and, H. Weigand: "Effective Communication in Virtual Adversarial CollaborativeCommunities", The Journal of Community Informatics ,Vol. 2 No., 2, pp.117-134, 2006.

[15] G. Thomas, and R.A. Botha,: "Virtual-Communities-as-a-Mechanism-for Sustainable- Coordination-within-the-South-African-Public-Sector". In J. Marijn, _L. _Winfried, P.H. Jan, & R. Michael_ (Ed.), IFIP Advances Information_and_Communication_Technology:E-Government, Eservices and Global Processes.334, Springer Boston, Brisbane, pp. 62-75, September, 2010.

[16] J. Warshaw, S. Whittaker, T. Matthews, B. A. Smith, "When Distance Doesn't Really Matter: Effects of Geographic Dispersion on Participation in Online Enterprise Communities". CSCW conference. San Francisco: ACM pp. 1-11, 2016.

[17] G.Thomas, R.A. Botha, D. vanGreunen, "Towards Addressing the requirement for coordination in large scale distributed environment: A virtual community-centric collaboration process lifecycle perspective". IEEE, IST-Africa 2015 Conference, Lilongwe, Malawi, 2015. 10.1109/ISTAFRICA.2015.7190581, Electronic ISBN: 978-1-9058-2451-9.

[18] G.Thomas, R.A. Botha, D. vanGreunen, "Understanding the Problem of Coordination in a Large Scale Distributed Environment from a Service Lens Perspective- A Case of the South African Public Sector e-Administration criteria for coordination support". Government Information Quarterly Vol.32, no. 4, pp.526–538, 2015. http://dx.doi.org/10.1016/j.giq.2015.08.002

[19] A. Alfaris, A. Siddiqi, C. Rizk, O. de Weck, & , D. Svetinovic Hierarchical Decomposition and Multidomain Formulation for the Design of Complex Sustainable Systems. Journal of Mechanical Design, 132(091003), pp.1-13, 2010.

[20] S. Balaji, S. M. Murugaiyan, "WateerfallVs V-Model Vs Agile: A Comparative Study on Sdlc". International Journal of Information Technology and Business Management, Vol.2, no. 1, 2012.

[21] D. D. Gajski, S. Abdi, A. Gerstlauer, G. Schirner, "Embedded System Design - Modeling, Synthesis and Verification". Dordrecht Heidelberg London New York: Springer, 2009.

[22] J. H. Gittell, F. Hagigi, D. B. Weinberg, C. Kautz, & W. R. Lusenhop, "Modularity And Coordination of Complex Work". Alfred P.Slaon Foundation industry study annual conference. Boston: MIT Sloan 2008.

[23] D. Lapakko: Argumentation: Critical Thinking in Action. New York, Bloomington: iUniverse, Inc., October, 2009.

[24] D. Walton: "Argumentation Theory: A Very Short Introduction". Argumentation in Artificial Intelligence, ed. Iyad Rahwan and Guillermo Simari,Berlin, Springer, pp.1-24, 2009.

[25] A. A. Alassiri, M. B. Muda, R. B. Ghazali, & U. C. Ahamefula; Usage of Social Networking Sites and Technological Impact on the Interaction- Enabling interactions. International Journal of Humanities and Social Science, Vol. 4 no.4, 46-61 2014.

[26] T. W. Malone, & K. Crowston, "The interdisciplinary study of coordination". ACM Computing Surveys, 26(1), pp.87–119, 1994. doi:10.1145/174666.174668

[27] A. Mowshowitz, Virtual organization: A vision of management in the information age. Inf. Soc. Vol.10, no. 4, pp.267-288 1994.

[28] N. D. Mehandjiev, A. A. Karageorgos, & G. Tsang, "Designing Coordination Systems for Distributed Teamwork", Twelfth International Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises Washington: IEEE Computer Society, pp. 117, 2003.

[29] A. W. Holt, "Diplans: a new language for the study and implementation of coordination". ACM Transactions on Information Systems, Vol. 6 no.2, pp.109-125, 1988.

[30] Bannon, Liam, & Schmidt (eds.), "Issues of Supporting Organizational Context in CSCW Systems, - COMIC Deliverable 1.1", Esprit Basic Research Project, Computing Department, Lancaster University, Lancaster, U.K., 1993.

[31] P. F. Marty, "Factors Influencing the Co-Evolution of Computer-Mediated Collaborative Practices and Systems: A Museum Case Study". Journal of Computer Mediated Communication 2006 .doi:10.1111/j.10836101.2005.tb00275.x

[32] P. Schubert, & M. Ginsburg, "Virtual communities of transaction: The role of personalization in electronic commerce". Electronic Markets, Vol. 10 no.1, pp.45-55, 2000.

[33] Y. Sumi, K. Mase, "Supporting awareness of shared interests and experiences in community", ACM 2001 International Conference on Support Group Work, Boulder, USA, pp.35-42, 2001.

[34] M. Kirsch-Pinheiro, J. Gensel, and H. Martin: "Representing Context for an Adaptive Awareness Mechanism", In G.-J. De Vreede, L. Guerrero, & G. Marín Raventós (Eds.), Groupware: Design, Implementation, and Use Lecture Notes in Computer Science Volume 3198, pp. 339-348, 2004.

[35] C. Gutwin, & S. Greenberg, "Effects of Awareness Support on Groupware Usability". Human Factors in Computing Systems. New York: ACM Press/Addison-Wesley Publishing Co. pp. 511- 518, 1998.

[36] C. Rocker: "Requirements and Guidelines for the Design of Team Awareness Systems", International Journal of Human and Social Sciences, Vol. 5, No., 4, pp.240-248, 2010.

[37] Q. A. Zhao, "Opportunistic Interfaces for Promoting Community Awareness". PhD Thesis. Atlanta, GA: Georgia Institute of Technology, 2001.

[38] E. R. Pedersen, & T. Sokoler, "AROMA: abstract representation of presence supporting mutual awareness". SIGCHI conference on Human factors in computing systems. New York: ACM, 1997.

[39] V.Mohanty, & D. Jagatram, "Personalization versus Customization User experience issues- A case study" 2009. Accessed October 24, 2011, from Slideshare: http://www.slideshare.net/djagatram/personalization-vs-customization-comparative-study-presentation

[40] J. S. Breese, D. Heckerman, & C. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering". Microsoft Research, 1998.

[41] S. Abbar, M. Bouzeghoub, & S. Lopes, "Context-Aware Recommender Systems: A Service-Oriented Approach". The 3rd International Workshop on PersDB, in conjunction with VLDB'09 Lyon: ACM. Pp.31-36, 2009.

[42] S. Abbar, S. Lopes, M. Bouzeghoub, A. Aghasaryan, D. Kostadinov, & S. Betge-Brezetz, "Personalized Access Model: Concepts and Services for Content Delivery Platforms". 10th International Conference on Information Integration and Web-based Applications \& Services Linz: ACM. pp. 41-47 2008.

[43] M. B. Linda: Modelling personalizations in the design of mobile publish / subscribe architectural framework for or South African National Parks, (012298). Thesis (M.A.- Computer Science) - University of Zululand, 2006. Sourced from: http://hdl.handle.net/10530/121

[44] Y. Huang and, H. Garcia-Molina: "Publish/subscribe in a mobile environment", Wireless Networks, Vol. 10, No., 6, pp.643-652, 2004.

[45] H. Shen: "Content-Based Publish/Subscribe Systems", In X. Shen, H. Yu, J. Buford, M. Akon, editors, Handbook of Peer-to-Peer Networking, pp. 1333–1366, Springer, Boston, MA, 2010. doi:10.1007/978-0-387-09751-0_49.

[46] H. Yoon, "A Convergence of Context-Awareness and Service-Orientation in Ubiquitous Computing". International Journal of Computer Science and Network Security, Vol.7, no. 3, pp.253-257, 2007.

[47] E. Thomas, Service-Oriented Architecture: Concepts, Technology, and Design. New Jersey: Prentice Hall, 2005.

[48] O. Zimmermann, N. Schlimm, G. Waller, M. Pestel, "Analysis and Design Techniques for Service-Oriented Development and Integration", in Proc. GI Jahrestagung Vol.2, pp.606-611, 2005.

[49] M. Momotko, W. Izdebski, E.S. Tambouris, K. Tarabanis, M. Vintar, "An Architecture of Active Life Event Portals: Generic Workflow Approach", In Wimmer, M., Scholl, J., Gronlund, E., (eds.) ,EGOV 2007, LNCS 4656, Springer pp.104-115, 2007.

[50] J. P. Williams: "Groupware: Shared Thoughts, Shared Media, and Shared Models", Technical report, School of

Information, University of Texas at Austin, 2003. Sourced from http://www.ischool.utexas.edu/~i385tkms//blog / archi ves/patrick/groupwarepaper.html.

[51] J, L. Maréchaux: Combining Service Oriented Architecture and Event-Driven Architecture using an Enterprise Service Bus; 2006. Sourced from, http://www.ibm.com/developerworks/library/ws-soa-eda-esb/.11, October 2010

[52] M. P. Papazoglou, P. Traverso, S. Dustdar, and F. Leymann: "Service-Oriented Computing: a Research Roadmap", International Journal of Cooperative Information Systems, Vol. 17 No.2, pp.223-255, 2008.

[53] D. Zager: "Collaboration as an-Activity", Computer Supported Cooperative Work, Vol. 11, No. 1–2, pp.181–204, 2002.

[54] G. Thomas, "A Virtual-Community-Centric Model For Coordination In The South African Public Sector" Ph.D. thesis, School of Information and Communication Technology, Nelson Mandela University, Port Elizabeth, South Africa, 2014.

[55] G. Thomas, and R.A. Botha: Towards a virtual organization for intergovernmental training coordination in the South African public sector. Proceeding of the 11th annual Conference on World Wide Web Applications (ZA-WWW2009), Port Elizabeth, 2009.

Biography

Godwin A. Thomas (Dr) Earned his PhD in IT at the Nelson Mandela University Port Elizabeth, South Africa in 2014. Prior to that he earned his BTech in 2006 and subsequently his MTech (cum laude) in 2007 at the same University. He currently works as a Lecturer in the Department of Computer Science at the University of Jos in Nigeria, where he had previously earned his National Diploma in Computer Science in 2001. He is currently a member of the Internet society (ISOC), and Nigerian Computer Society (NCS). His research interest include Computer support cooperative work (CSCW), Information security governance, Mobile Computing and IT service management. He is ITIL certified.

Reinhardt A. Botha (Prof) Earned is MSc and PhD in Computer science from the Rand Afrikaans University (RAU) in South Africa. His BSC (Hons) was the University of Port Elizabeth (UPE) now the Nelson Mandela University where he is employed as a Professor of Information Technology at the Institute for ICT Advancement and School of ICT in Port Elizabeth South Africa. His expertise and research interest are in the areas of Information Security, Privacy and IT Service Management, Workflow and End to end service Design. He is a certified ITIL expert.

Darrell van Greunen (Prof) Earned her PhD (Computer Science), at UNISA; M A (Computer Science) Cum Laude B.A. Honours (Computer Science), 1999, UPEFDE Educational Computing, 1994, UPEHDE, 1989, at the University of Port Elizabeth (UPE). Prof. van Greunen is currently employed as a Professor of Information Technology at the Institute for ICT Advancement and School of ICT in Port Elizabeth South Africa. She is also the the Director: Centre for Community Technologies at Department of Information Technology. Her research interests and Expertis are in the areas of User Experience, User Interface Design, Impact Assessment, Rural communities, Living Labs, eHealth, Health Informatics, ICT4D, ICT for Socio Economic Development.