Future Internet Architectural Layers to Support Secure Pervasive Technologies

Atta Badii
Intelligent Media Systems and Services Research Centre (IMSS)
School of Systems Engineering, University of Reading,
United Kingdom
http://www.imss.reading.ac.uk
atta.badii@reading.ac.uk

Abstract—In this paper, we describe how the requirements and criteria for enabling seamless integration of all available resources to build the future internet are fulfilled by the MobiPETS application layer and the Hydra middleware hand in hand. We describe how the presented platform enables users, developers and system integrators to easily develop applications for the future internet keeping in consideration the important issues such as security and privacy constraints, user-centric authentication and authorization, self-aggregation of information, policy-based discovery and selection, semantic-cooperative reasoning and agent technology, event-driven agent-based management of tasks and sub-tasks, integration of seamless localisation, etc. The paper also presents the User-Intimate Requirements Hierarchy Resolution Framework (UI-REF), which has been found to be particularly effective in elicitation and determination of priority requirements as well as integrative prioritisation of the evaluation of the most usability-sensitive features of the priority requirements.

Keywords-Future Internet; Middleware; Context; User Intimate Requirements;

I. INTRODUCTION (HEADING 1)

As smart mobile sensors and devices increasingly pervade our every day lives in an emerging ambient world, there is a need to make available smart and context-aware services, trustworthy and personal mobile devices, over seamless and transparent networks. The ambient lifestyle is characterised by users’ interactions with one another as well as with service provisioning platforms in their immediate environment no matter what kind of mobile and wireless sensor, service and network technologies are behind it. The ambient user explicitly expects intelligently personalised services and seamless cooperation with those technologies depending on his or her current situation and context. Furthermore, such a user implicitly requires transparent, but inherently secure communications and trustworthy transactions. Aarts and Marzano summarise the five key technology features that characterise an ambient intelligent system [8] as a) Embedded: networked devices integrated into the environment, b) Context aware: recognises people and their situational context, c) Personalised: can tailor itself to meet people’s needs, d) Adaptive: can change in response to people, and e) Anticipatory: anticipates people’s desires without conscious mediation.

Devices and computing platforms used in the ambient environment are bound to handle personal, highly-confidential data and context information. These must be protected to ensure that the ambient user’s privacy and confidentiality is assured. It is also essential to guarantee the integrity of information transferred in an ambient networking environment as well as to ensure that only authorised principals have access to resources in the environment. Furthermore, information gathered from sensors and data sources in the environment of a roaming ambient user may not necessarily be trustworthy. It is thus expedient that the ambient user should be able to simply specify their security, privacy and trust requirements as policies which must be enforced in an ambient environment. The key question is how we can achieve seamless semantic integration of mobile heterogeneous networked services and devices with privacy-personalisation considerations, and furthermore develop enhanced technologies for delegated service selection and provisioning.

II. REQUIREMENTS

A number of criteria have to be considered for optimal, effective, efficient and secure sensing and information exchange across the networks of heterogeneous distributed devices and services that are to underpin the vision of the future internet. These criteria are envisaged to be as follow:

1. Authentication-Security-Privacy-Trust (ASPT) based interoperability, messaging and processing of information
2. Entropy and green ICT filters, to maximise the efficiency of messaging i.e. transmissions between the sensors around the network.
3. Balancing of local versus central processing of exception reporting and fusion levels for most efficient and effective bandwidth consumption
4. Multi modal context aware event processing: Distinguishing between event classes within the event cloud so that complex event processing remains capable of responsively handling both emergency based as well as routine comfort supporting messaging and responsive behaviour
A. Architectural Layers

Accordingly, the following architectural layers are distinguished as logically and implicitly required:

1. Semantic Device & Domain ontology generalisation hierarchy to model the domain nodes (devices) within the network
2. Applications use-context referenced Requirements Usability Prioritisation Hierarchy of needs (UI-REF-based)
3. Cooperative Agents Communication Semantics (KQML or other) for Knowledge Getting Acts (KGA), Knowledge Reporting Acts (KRA) and Knowledge Integration Acts (KIA)
4. Policies (KGA/KRA/KIA) to support the management of effective Complex Events Processing (e.g. considering resources – CPU, power, green ICT etc. filters for optimal processing per node capabilities) to determine locally at node and hub level FUSION (Fusion Level 0, Fusion Level 1, Fusion Level 2)
5. Filters to provide messaging-receiving Authentication-Security-Privacy-Trust (ASPT) Modelling at the application level (as enabled by MobiPETS GRID modelling of the domain to support the deployment of appropriate ASPT information exchange and service-oriented user/proxy transactions).
6. Ensure appropriate communication security and interoperability across the heterogeneous distributed nodes and/or hubs in the Future Internet environment (as enabled by Hydra LinkSmart for semantic ASPT at middleware layer)
7. Situation Assessment comprising of device states and person states (semantic integration)
8. Pattern Discovery & Case Based Reasoning to support Complex Event Processing, Situation Assessment and Responsive Action e.g. recommendation, adaptation, alerts etc
9. Networks to be supported (Body Area Network, Clothing Area Network, Smart Home Area Network, Car Area Network, Smart City Street Area Network, Office Area Network, Remote Installation Area Network)

III. CONTEXT-AWARE MOBILE SERVICE PROVISIONING FRAMEWORK

To support the future internet, we require technologies for seamless semantic integration of mobile heterogeneous networked services with security-privacy-personalisation-trust considerations and automated service selection and provisioning. An interoperable grid-aware semantic architecture is needed as a robust, flexible and open engineering solution that can integrate the various components involved and offer individualised secure and trustful service provisioning to deliver ubiquitous call-by-call service matchmaking and QoS management. This essentially amounts to a new secure service provisioning communication management layer that incorporates new domain boundary virtualisation paradigms for access control (e.g. trusted/self domain versus external service-provider’s domain) and new breed of middleware supported by semantic-cooperative standards to deliver the service personalisation (Preference Technologies).

Such a solution requires architecting the semantic representation of communicating domains and their enclosures to allow S&T services selection, composition and matchmaking. This entails providing adaptive and personalised protection for each entity through distributed management and delegation of security protection to smart grid-enabled proxy services.

The solution system must include layered context-aware behaviour-based models to facilitate socially intelligent secure user delegation of protection to such smart proxies within a framework for Personalisable Privacy and Trust enhancing technologies (PETS). Such PETS are to deploy user advocacy-delegation services, user security knowledge management support and secure e-services bundling, SLA negotiation, contracting, e-billing and e-ticketing.

MobiPETS [9] is a context-aware mobile service provisioning framework that realises multilateral security for seamless semantic integration of context-aware mobile services. Existing solutions are not capable of providing an integrated framework for context-aware computing re the following requirements in service personalisation call-by-call:

1. Service discovery, selection, composition, matchmaking, SLA negotiation, contracting and billing
2. Dynamic provision of advocacy, privacy, security and trust services
3. Robust context sensing, context switching and context serving (e.g. as may be facilitated through using topic-map technology and inferencing techniques)

There are some fragmentary solutions providing piecemeal approaches towards satisfying some of the above requirements. This diversity of techniques and methods brings added complexity and with it added risks and vulnerabilities that detract from the evolution of a robust integrative framework solution for the above three key requirements. However the MobiPETS-GRID solution framework additionally serves to hide the complexity of security-privacy service provisioning from the application layer. It allows grid nodes in the service provider zone to enter into specific dynamic trust relationships with each other and with the specific user’s private domain (user’s trust domain). This facilitates graceful integration of auxiliary services such as identity management (AAA), global single sign-on through any service provisioning scenario end-to-end (fruition) and service chain decomposition for multi-sourcing and aggregation (e.g. as in call-by-call outsourcing) (cf. [7]).
IV. REFERENCE ARCHITECTURE

We distinguish virtual communication domains across whose boundaries scalable security services could be tasked to manage a user’s dynamic S&T chain in the context of such user’s business logic and value chain. Thus for each such client device/user three interacting domains are virtualised to include all hardware and software modelled under each of:

- **Self-Domain or Intimate-Domain (S):** All HW/SW constituting user’s personal client devices (home computer, office computer, laptop, PDA, mobile phone and native applications running on them (calendar, diaries, profilers etc)
- **Others-Domain (O),** All HW/SW belonging to all other parties transacting with the user including that of peers and grid-enabled services
- **Smart-Middleware-Domain (M):** Any component involved in mediating data exchange across the boundaries between the above two domains

As can be seen in Figure 1, the grid services, located in the service domain, provide their value-added or enabling services to the user through the network infrastructure. To protect the privacy of the user and to secure the interaction at all levels of communication, the access is controlled within the trust domain. It contains the hardware and applications which process and store personal data, and, the mobile devices and the privacy enhancing technologies.

A homogenous knowledge representation of all data sources is generated, enabling the user to simply access all information, regardless of its location. The representation is provided through the personal server which uses topic maps technology. This is to unify the access to the stored information. So portions of this representation can be transferred to the mobile device which enables the user to browse it. This is where the usage of ambient information comes in to simplify the search process and to enable the clustering of information. To provide these abilities, all stored information is interconnected to each other through the usage of context information.

The power of these access abilities and the distributed information has to be protected against unauthorised interaction. Therefore, an identity/persona management approach is used to store security related information like policies, keys and contacts in extension to the profiles of the individual. This way the user controls the incoming and outgoing information and ensures the trustful communication with other users with respect to their trust domains. The key requirement here for the need of an id/persona management system is the imperative of a secured component (a super-agent /PA) who would handle all requests that are related to profile operations.

MobiPETS GRID powerfully integrates the services of trusted nodes from amongst a computing services grid network referred to as Trusted Grid Network (TDN) whose services are securely harnessed together with resources available from the user’s Intimate Domain Network (IDN) so as to satisfy services and more complex computations invoked by the nomadic professional anywhere whilst using a PDA or mobile as a client device.

A user’s **Intimate Domain Network** comprises the user’s Body Area Network (BAN), Car/Craft Area Networks (CAN)
as appropriate, and hand-held devices normally carried by the user together with all native applications running on such devices which constitute the user’s personal and private information space.

A user’s Trusted Grid Network is not just any GRID network but a privileged grid network that satisfies a particular high Authentication, Security, Privacy and Trust (ASPT) criteria. A TDN is thus a fallback network of computing resources on which the user client devices from within the IDN can rely to satisfy the services and computations required by the user whilst on the move. This is because the mobile devices comprising the IDN such as a heart or kidney function monitor, mobile phone etc. are invariably resource limited in terms of computing power, battery and memory. So the user can designate any ASPT-satisfactory computer as being a node within a Trusted Grid Network (TDN). The TDN’s computational service (CPU, Memory) can be invoked by the user from any node within a user’s IDN so as to reduce the computing and battery power demands on the mobile devices within the TDN. Additionally some of the computing resources of the TDN can be deployed continuously to provide comprehensive frontline security sentry or guard services for the entire IDN-TDN axis end-to-end including secure live application contracting.

The boundary of both the IDN and the TDN can be extended to include other devices designated by the user as being within IDN or TDN. As more devices tend to be used as client terminals by the user whilst on the move, indoors or outdoors, they may be added to the IDN. Also as more and more trusted computers are enlisted by the user to act as potential computation or storage back up then the wavefront of the TDN is expanded to include such devices as nodes within the Trusted Grid. In this way the innovative integration of emergent semantic-cooperative standards, topic map and preference technologies, and Authentication, Security, Privacy and Trust enhancing Technologies (.PETS) as envisaged in MobiPETS allows dynamic context-sensitised boundary management between any user and all other transactors (service providers) i.e. between user(s’) Trust Domain, and, the external Service Domain. This means as new secure service provider nodes are identified they can be included in Trusted Domain Network i.e. join the IDN-TDN Axis; conversely if any nodes within this Trusted GRID are discovered as rogue (insecure, deviant, uncooperative, faulty, low-battery) then it can be excluded, suspended, quarantined or anyhow by-passed and appropriate defensive-recovery action taken to ensure continuous secure operation.

As such the IDN-TDN axis always remains adequately resourced and protected so as to provide the user’s Smart Proxy with robust computation services and adequate storage resources at any time, any place. It also provides the user’s proxy with frontline SPT protection and advocacy services in selecting, aggregating and SLA contracting of any services over the web as required by the user.

Thus MobiPETS GRID powerfully supports efficient pervasive computing, effective deployment of ambient intelligence, secure interoperability over heterogeneous distributed networks and self-configurable, self-evolving ad hoc networks as underpinned by the .PETS and Preference Technologies.

V. HYDRA: MIDDLEWARE FOR NETWORKED DEVICES

The HYDRA Middleware Project provides a middleware for intelligent networked embedded systems, deployable on both new and existing networks of distributed wireless and wired devices. It provides the necessary technologies for enabling context-aware applications and services, while being domain-agnostic and easily adaptable. Hydra takes a hybrid approach to context-awareness, by providing the capabilities for both high-level, powerful reasoning, based on the use of ontologies, as well as the relatively lower-level semantic processing enabled by the modelling of contextual data using an object-oriented/key-value approach.

The Hydra Middleware is based on a Service-oriented Architecture (SoA), to which the underlying communication layer is transparent. The middleware includes support for distributed as well as centralised architectures, security and trust, reflective properties and model-driven development of applications.

The vision of the HYDRA project is to create a widely deployed middleware for intelligent networked embedded systems that enables: 1) Cost effective development of innovative ambient intelligence solutions, 2) Low cost for device manufacturers to be part of “Internet of things”, 3) Secure and reliable services for end-users.

The technical objectives of the project also include support for reflective properties of components of the middleware and support for security and trust enabling components. Furthermore, the project includes development of a generic semantic model-based architecture supporting model-driven development of ambient intelligence applications. Hydra provides tools for solutions providers (SDK) that are easy to integrate and use devices in applications that hide complexity of underlying network and device access protocols and are integrated into familiar programming environments. Hydra also provides tools for device manufacturers (DDK) that incur low cost for networking devices and offer support for their devices to be part of “Internet of Things”.

Hydra middleware Architecture includes Core Hydra Components/Managers such as:

- **Device discovery**: Discovery Managers, Application Device Manager, Application Ontology Manager
- **Secure Hydra networking**: Network Manager, Crypto Manager, Trust Manager
- **Resource access control**: Access Control Policy Framework
- **Context data handling**: Context Manager

Figure 2 provides an overview of the various managers in the Hydra Middleware, with respect to their relative positions within the communication stack. As can be seen, the middleware provides a layer between the physical communication with devices, and the developed "Hydra" Application, using the middleware.
A model-driven approach has been chosen which allows the representation of security requirements, policies and capabilities at a semantic level. With the help of security ontologies, these specifications are translated to concrete protocols and mechanisms that are supported in the ambient environment as the basis of negotiation between a service producer and a consumer to arrive at common protocols and mechanisms during service provisioning and consumption. The security ontologies used themselves are not provided by the middleware architecture, but the generic ontology-based reasoning provided by the middleware is used to perform the security resolution at runtime based on the supplied ontologies to achieve this interoperability goal.


A. Energy Efficiency in Hydra

The Hydra Middleware can be utilised to build systems capable of powerful Energy monitoring and action, in order to reduce the end-users carbon footprint, by configuration. Hydra-enabled devices provide several services, one if which is an EnergyService, providing full energy profiles for devices, including energy consumption and remaining lifetime. This information forms the Energy Profile of a Device Context, as modelled in the Context Manager.

Reasoning over this contextual information can yield powerful results, depending on the application domain involved. In the domain of Home Automation, the Context Manager can record a historical view of the overall energy consumption of devices, in addition to the environmental context it which they were used. The Context Manager can be configured to intelligently control usage of energy-consuming devices, sensing environmental data (temperature, humidity), as well as the state of devices in the home - windows, heaters etc - to select the most appropriate form of achieving the desired result, with minimal energy consumption. For example, if the temperature in the user's home is significantly higher than his desired living temperature for that time of day, and the temperature outside is cooler, then the decision should be made to open windows to provide the cooling, rather than activating the air conditioning.
VI. UI-REF: USER-INTIMATE REQUIREMENTS HIERARCHY RESOLUTION FRAMEWORK

UI-REF is an Integrative Interpretivist Framework for User-Intimate Requirements Hierarchy Resolution and Usability Evaluation based on earlier work (Badii 1997-2008) to optimise the requirements engineering process particularly to support user-intimate interactive systems co-design. UI-REF has been established to ensure that the most-deeply-valued needs of the majority of stakeholders are elicited and ranked, and, the root rationale for requirements evolution is trace-able and contextualised so as to help resolve stakeholder conflicts. Requirements prioritisation in UI-REF is fully resolved while a promotion path for lower priority requirements is delineated so as to ensure that as the requirements evolve so will their resolution and prioritisation.

UI-REF follows an inclusivist and interpretivist approach to stakeholder participation and requirements elicitation and evaluation. This is to ensure optimal user-centred co-design and evolution of the resulting user-intimate systems. It addresses the requirements hierarchy elicitation, analysis, generalisation and resolution of users’ needs priorities taking into account that what is more “intimate” to a user is also what is perceived as more relevant in the user’s own context. Thus once preliminary steps have been completed, it is necessary to analyse the domain usage-context spectrum so as to prioritise the objectives of the requirements collection and analysis. To this end it is necessary at least to 1) agree and set out the list of domain prototypical entities or actors and objects as stakeholders, 2) define the characteristics of the prototypical entities which will have to include any entities that are involved, i.e. implicated, in any way whatsoever in the typical usage arena envisaged for the target system (i.e. need to set out respective capabilities, roles, rights, responsibilities for each entity i.e. each actor/object/process and respective entity lifecycles and life-courses), 3) establish the generalisation ontology of usage-contexts each related to their distinct Man-Machine interactivity requirements as needed by the relevant user (sub)groups in their target prototypical scenarios, 4) define the key differentiators of usage-contexts (context switches) and the prototypical actors’ needs hierarchies in each of the identified prototypical target context-scenarios, 5) define the prototypical workflows and state diagrams, thus establishing the domain user/practitioner’s (sub)-goal and (sub)-task hierarchies, and 6) deduce the user’s needs priorities in terms of ICT-enabled features to facilitate users’ task fulfilment in each situated-context-scenario of the application domain as identified and demarcated (situated-usage-class) under respectively 3 and 4 above.

Such steps can also be grouped in terms of the kind of understanding and knowledge that they provide about the contextualised system to be designed. Besides, they can be more finely-grained and profitably grouped to provide a valuable level of abstraction and management of detail which is very important in deep-inspection / introspection of requirements elicitation and in ranked-and-weighted usability evaluation of the most usability-sensitive requirements. Responsive adaptation of the intended functionalities of the system (its affordances or effects), and, their respective situated side-effects, and concomitant human affects and their perceived system usability and therefore appropriation destiny demands an integrated requirements engineering and usability evaluation approach as adopted in a User-intimate Co-design framework such as UI-REF.

UI-REF advocates that the requirements are classified into the following descending-order priority categories for implementation.

**Mandatory** – These are those ICT design features which are perceived by the majority of a user group as offering the most needed added-value(s) and that can be accommodated by the target system. This category is expected to include the functionalities that are the essential common core to all the usage-contexts within the target usage spectrum; including features supporting scalability, modularity and open design so as to enable the incremental evolution of the system to offer further features to satisfy future requirements and customisation as appropriate. These include both selected functional and selected non-functional requirements.

**Desirable** – These are those features that are desirable, but are not the highest priority design features; these are candidates to be accommodated as far as possible, within the resources and technological constraints appertaining to the lifecycle of the project.
Optional – These are those features said by some users to be of the lowest priority and/or are anyway highly contextualised to particular (sub-)sectors of the user group and as such falling into the less common, and/or possibly more controversial and confliction category.

Once the raw user-stated requirements are aggregated from all elicitation channels and modalities, they have to be transcribed, tabulated and cross-checked to prune duplications and delete clearly out-of-scope requirements. UI-REF promotes a negotiation-based resolution of requirements into the above three categories to reflect the priorities of the majority of users [3,4,6].

Next, additional checks have to be done to flag up for negotiation with the stakeholders the possible deletions, demotions, promotions and new additions of specific requirements to be consensually resolved into the set of mandatory, desirable and optional requirements for the first prototype [3,4,6]. The need for the following refinement steps arises as a natural consequence of the fact that the users in stating their requirements can not be expected to be either exhaustive or factor in technology, market and practice constraints (State-of-Art, State-of-Market, State-of-Practice) and trends of which they are not necessarily expected to be fully aware. Further, users are expected to articulate their own perceived requirements which may or may not be complete and may be incompatible with other users’ requirements or project resources or in conflict with the technological and/or market imperatives and trends [2,5,6].

Prioritisation Process

1) User’s statement of preferences assisted by the Requirement Engineer, mediated by instruments such as:
   a) Nirvana
   b) Ablation
   c) Noah’s Arc

2) Filters deployed by the Requirement Engineer
   a) FPFilter (Frequency, Purpose, Hurry)
   b) Life/Workstyle
   c) Patterns/Entity Lifecycles

UI-REF, as an evolutionary methodology for requirements engineering and co-design, advocates that the above filters/augmenters are periodically revisited. This is necessary so as to keep up with market and technology evolution through a market-technology-watch task. This task has to report any further filtering/augmenting or other modification suggestions in time to be integrated with the usability evaluation results for the current prototype so as to conclude the requirement engineering update that shall inform the re-engineering and refinement of the next prototype. Thus the application of the above Filters/Augmenters will ensure that additional negotiation is undertaken to resolve those user-stated requirements that are unachievable within the scope of the project and/or will militate against the market trends or mean that obvious current and emergent gaps in the market will not be addressed.

<table>
<thead>
<tr>
<th>Dynamic Usability Relationship Evaluation: point-of-experience and latent-affect informed co-design</th>
<th>Prioritised [Usage-Context]</th>
<th>Prioritised [Effect]</th>
<th>[Side-Effect]</th>
<th>[Affect]</th>
<th>[Usability Relationship Impact]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usability Relationship Goal Function Evaluation with Example metrics</td>
<td>Maximising Usability Relationship a f {safety, security, autonomy, dignity-trust-privacy-preserving, comfort, satisfaction}</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Prioritisation in UI-REF is performed at several levels including the consideration of stakeholder type, intimate or non-intimate usage-context-type, and specific usage-contexts within each type, usage-context implementation sequencing and usability-sensitivity. Evaluation of system functionalities involves ranking and weighting their usability assessment such that it would reflect their priorities in the order of the most deeply-valued needs of the user as prioritised per the UI-REF requirements engineering process. Such prioritisation processes will use a variety of situated techniques as appropriate to best suit particular users and their usage-contexts for example Nirvana, Ablation and Noah’s Arc, virtual user, nested videos, FPH-based analysis of user specified needs and Effect-Side-Effect-Affect matrices which are described elsewhere as part of the C-Assure methodology [1].

VII. CONCLUSION

A multitude of heterogeneous mobile, personal devices and virtual services is at hand to today’s tech user, and it is high time to invest effort in technologies that seamlessly integrate all available resources to build the future internet. An Internet of things connected in pervasive, context-aware, secure, privacy-sensitive, efficient and effective manner to support the ambient lifestyle of the user.

As described briefly in this paper, the envisioned and listed requirements and criteria for such the future internet, are able to be fulfilled by MobiPETS application layer and Hydra middleware hand in hand. This platform provides users, developers and system integrators to easily develop applications for future internet keeping in consideration the important issues such as security and privacy constraints, user-centric authentication and authorisation, self-aggregation of information, policy-based discovery and selection, semantic-cooperative reasoning and agent technology, event-driven agent-based management of tasks and sub-tasks, integration of seamless localisation (sensor fusion) etc.

Hydra, an open source, developer friendly, secure, and intelligent middleware platform that has been proven to be successful in the domains it has targeted, such as in Home
Automation, Healthcare, Energy Management and Food Safety (farm-to-fork traceability), through various demonstrators over the course of the project. Finally, UI-REF has been found to be particularly effective in elicitation and determination of priority requirements as well as integrative prioritisation of the evaluation of the most usability-sensitive features of the priority requirements. This has served to accelerate the refinement of the highest priority features of the system through rapid prototyping. It has established the efficacy of a UI-REF empowered agile evolutionary user-centred co-design approach to Intimate Systems innovation as is essential for the vision of future internet to be acceptably realised.

REFERENCES


