Digital Democracy in Knowledge Society: A Proposed Architecture Based on Cloud and Complementary Technologies

Ovidiu TURCOANE
University of Economic Studies, Bucharest
turcoaneovidiu06@stud.ase.ro

This paper introduces some technologies that are fit for an architecture of digital democracy or E-democracy. It aims at proposing an architectural style emerged from tested and validated approaches, without relying on some radical innovation. Firstly, we propose an input-system-output model of E-democracy and knowledge society. This model is subject to permanent optimization following a trial and error paradigm similar to the artificial intelligence method of backpropagation. Secondly, we describe and advocate for some technologies and methodologies such as Cloud, Service-Oriented Architecture, Agile Development, Web-Oriented Architecture, Semantic Web and Linked Data. Finally, we assemble all these technologies and methodologies in an architectural style that follows several key concepts such as flexibility and adapability, citizen-oriented software development or abstract notions like participation, deliberation and inclusion.

Keywords: *E-democracy, Cloud Computing, Services-Oriented Architecture, Knowledge Society, Agile Development, Semantic Web, Linked Data*

1 Introduction

One of the salient matters of computing world is to expand its visions and horizons from a technical to a social dimension. Arguably, information and communication technology (ICT) may arrive with its new approaches and perspectives to build a better society. While ICT has evolved as a human-like universe transposed in mathematical and computational formalizations, it is now time for the public sphere to benefit from the scientific achievements of the virtual world. From a social perspective, these benefits may endeavor the development of E-society, which is either the information society (IS) or knowledge society (KS). Moreover, IS or KS are foundations for a better public sphere by supporting a democratic society.

Digital democracy in knowledge society or E-democracy (ED) is to many an extension of E-government, but we have already defined it as being more than this - a way of living [1]. Based on Maier's research [2] and extending ED with new instruments like E-petition under the guard of Justice, Figure 1 illustrates our proposed model.

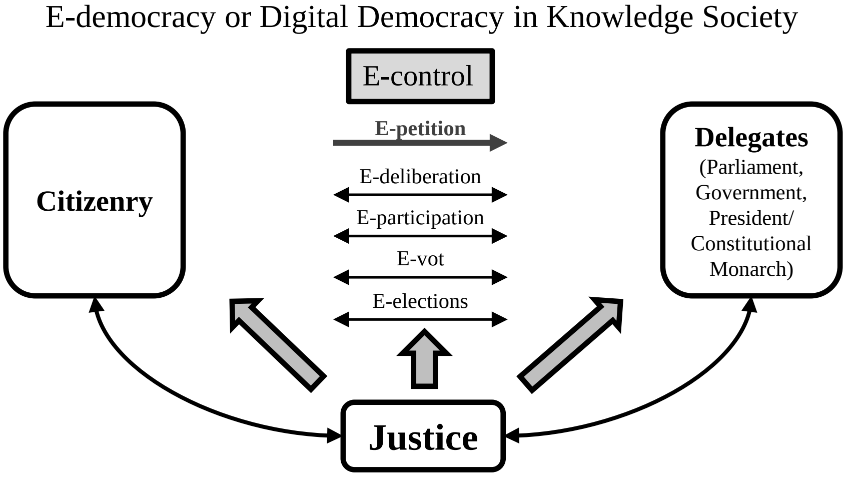

E-democracy or Digital Democracy in Knowledge Society
E-control
Citizenry
E-petition
E-deliberation
E-participation
E-vot
E-elections
Delegates
(Parliament, Government, President/ Constitutional Monarch)
Justice

Fig. 1. Instruments of E-democracy

DOI: 10.12948/issn14531305/18.4.2014.07

While Figure 1 has a generic perspective, ED should focus, on a bottom-up approach, on solving contextual problems (CPs) based on participation, deliberation and inclusion (PDI). Advocated by participative democracy proponents from antiquity [3] to modern [4] and contemporary times [5, 6], PDI is the key to ED and a metaphor for KS. Justice, seen as the backbone of democracy inspired by divinity [3, 4], is herein subject to permanent transformation through PDI on medium-long term, while some stability is required on short-medium term [6]. In addition, Figure 2 depicts the actors involved in the CP processes and their defining inter-relationships: committee of MPs (CMP), committee of citizens (CC), helping committee of citizens (HCC), non-governmental organizations (NGO), social and professional associations (SPA) or political parties (PP) [1]. All of them co-work on a platform of CP (PCP) or instantiate a CP (ICP) only under the surveillance of committee of justice representatives (CJR).

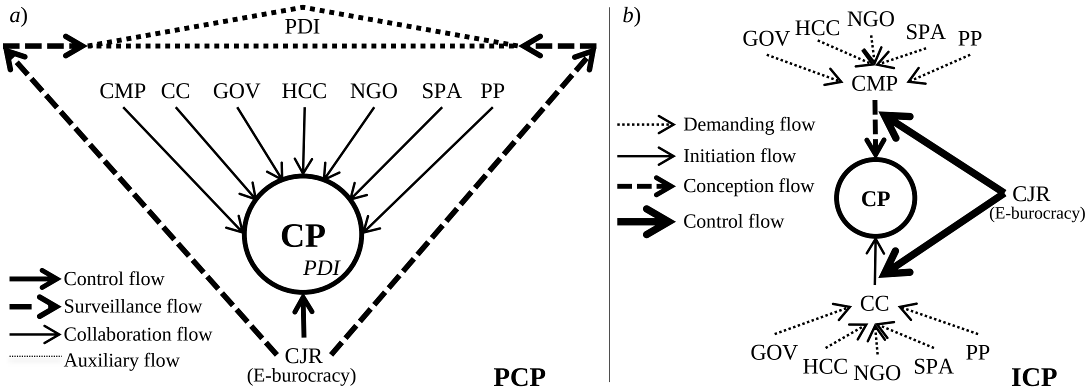


Fig. 2. Components of E-Democracy

We have already brought several important amendments to other (representative) democratic models [1] advocating for: the increased role of citizenry, the way of selecting representatives, the crucial part of justice and the significant aspect of PDI. In addition, we only want to address the issue of establishing a better framework for collaboration and cooperation based on ICT. While this is subject of future extended research, we mention that E-bureaucracy is an improved and more objective (non-Kafkian) form of bureaucracy that helps monitoring ED by CJR and supports the actors involved in PDI. E-bureaucracy incorporates techniques and methods of web semantics, neural language processing, text mining, artificial intelligence (AI) etc. that conceive a substratum for (E-) justice in particular and PDI in general.

A short paperwork like this one could not thoroughly argue for ED or KS and we herein want to highlight an ICT perspective that designates some technologies (substantially visually illustrated, but also literally depicted), which loosely and naturally build a foundation for a democratic better society. Defining the architecture of ED (AED) by an individual or an organization is rather a totalistic approach. AED should build itself on a bottom-up strategy using validated common shared knowledge and technologies. Describing AED, we only want to prove that these technologies already exist as well as some propensity in achieving this goal.

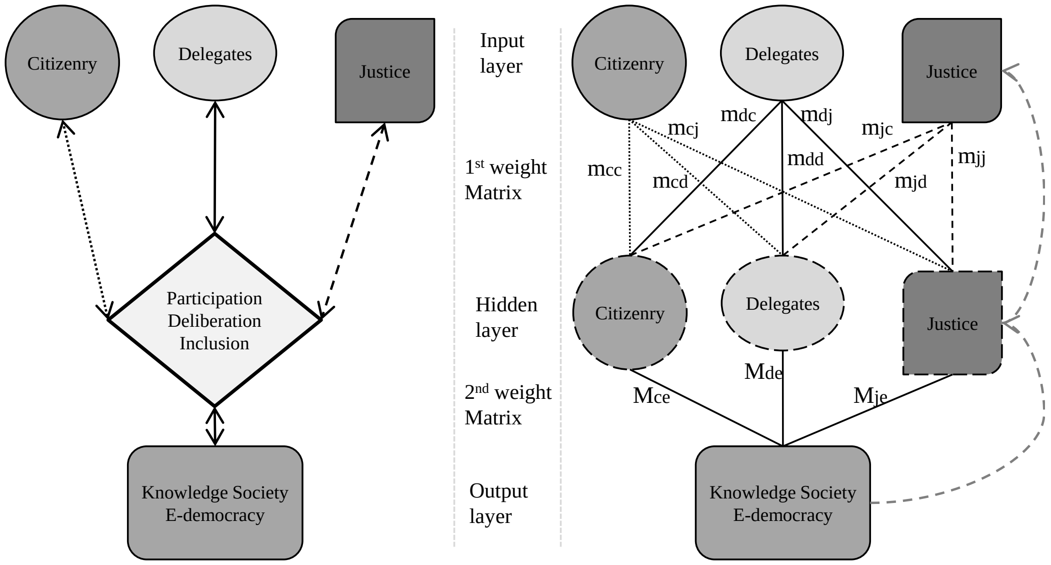
This article has the following structure: the next section discusses KS and its role for supporting ED, in section 3 we present the technologies of AED, section 4 assembles the previous sections framing AED and the final section presents the conclusions of this research.

2 Knowledge Society (KS) and E-democracy (ED)

In the model herein, KS and ED overlap each other. While they share the same public sphere, they are different as the former deals mostly with the social life and the latter concentrates on political domains. Let us take them altogether as a unitary block for the moment and let us define them as the output in a

DOI: 10.12948/issn14531305/18.4.2014.07

joint model of ED and KS (MEDKS). For more than half a century, political sphere has already been seen [7] as an input - system (processes) - output model (ISOM). MEDKS has the same approach only that we compare it with an AI system that is self-adaptive and subject to optimization through the trial and error process called backpropagation (BPE). Figure 3 illustrates the similarity of MEDKS and BPE, emphasizing the training process of CDJ through PDI, leading to a maximum output (i.e. KS and ED). While the so-called hidden layer of BPE concerns virtual computational intermediary results, MEKDS expects this layer (of PDI) to be transparent in the real public sphere.



Input – system (processes) – output model

Backpropagation - backward propagation of errors

Fig. 3. E-democracy's input-system-output model

Still, following the steps of BPE [8], MEKDS has a similar learning-optimization method:

```
MEDKS 1) Planning the objectives and the margin for the optimum output;
MEDKS 2) Defining CDJ's tasks (weights);
MEDKS 3) Training process of PDI;
MEDKS 4) Checking the intermediary results;
MEDKS 5) Negotiations after verifying the intermediary margin of the output;
MEDKS 6) Checking the errors given by intermediary results;
MEDKS 7) Checking the errors given by CDJ;
MEDKS 8) Planning new objectives closer to the anticipated optimum output;
MEDKS 9) Planning new tasks for CDJ;
MEDKS 10) If intermediary objectives are far from the final ones, go to MEDKS 3;
MEDKS 11) An epoch of negotiations and learning is done. If all objectives
are reached, stop the process; other way, go to MEDKS 3.
```

The first five steps of BPE are part of the feedforward process, which for MEDKS becomes an anticipation (of results) mechanism. Starting from the sixth step, the BPE itself trains the system to optimize the results, which for MEDKS means a trial and error PDI process to ameliorate and refine both outputs and inputs.

The remaining of this section will discuss the role of KS in building ED, their common realm and some points that differentiate them based on several contexts. An acknowledged approach on describing KS belongs to UNESCO that defines three objectives to accomplish four desiderata [9]: cultural diversity, equal access to education, universal access to public information and freedom of speech. The objectives are: *i*) promoting digital opportunities and social inclusion by using ICT; *ii*) increasing capacity for scientific research, information and cultural propagation, performance and cooperation; *iii*) enhancing through ICT the opportunities for (E-) learn-

DOI: 10.12948/issn14531305/18.4.2014.07

ing by enlarging and diversifying the educational system. These objectives comply with PDI and MEDKS, conceptualizing a multi-dimensional realm that supports ED.

UNESCO principles are not idealistic or utopian, but they rather acquiesce to the human needs and aspirations. Developing a cyborg metaphor through an anthropologist prism, Haraway [10] describes a new individual, arisen from reality and fiction and born from human, social and digital. It makes agree with moderate feminism and with environment, while it is a symbol for inclusion (part of PDI) and not a human-machine mutant.

Shifting from individual and somehow mythical views to social and methodical perspectives, we should identify the frame of IS that foresees KS in a general assumption. Although a supporter of information role in society but an opponent of theory of post-industrial or information era, Webster points out on the peculiar assumption that quantitative changes produce qualitatively different social order [11]. While these five mutual non-exclusive perspectives (technology, economy, labor structure, space-demography and culture) are important, he emphasizes the role of theoretical knowledge (a qualitative factor) as a dominant key of nowadays society. He also mentions the drawbacks of false needs, informational garbage and information gap, reminding the importance of knowledge capital in a society of open access.

From a different conceptual position and using an economic approach, Castells firmly asserts that *technology is society*, while making the analogy of contemporary world with a system of social connections axiomatically designated as *network society* [12]. His sociological approach stresses out 'informationalism' that changes society (not quite in a revolutionary way) and the paradigm of knowledge acting over knowledge. More, network society is similar to Habermas' public sphere [5] on general level, but it extends to network of networks that lead to smaller public spheres fit for debate, information transfer and autonomy [11], an approach that supports the concept of CP (see Figure 2). Key elements like economic perspective and flexible labor force that helps integration of women, defines the new society (of *flexible woman* replacing *man organization*) that relies on the following concepts [12]: *i*) technology supports innovation and new employment opportunities; *ii*) self-programmable work leads to innovation and higher productivity; *iii*) new internet sociability creates new human bonds and new socio-politically active individuals; *iv*) helped by ICT, networked individualism is culturally predominant; *v*) media is *a*) global and local (*glocal*), *b*) digitalized, flexible and diversified and *c*) subject to self-directed mass communication; *vi*) political process is transformed by culture of virtual reality and *vii*) network-state replaces nation-state through globalization.

Thanks to ICT, society has been experiencing many transformations in a more accelerated rate than decades ago, becoming quite an E-society for some academics. Through innovation (not a radical but an incremental one at a fast rate) that resides on knowledge, an emergent KS seems to spring forth and this is the stage where PDI acts in order to achieve ED.

KS and ED are related and a delimitation between the two takes into account the fact that KS may support a less democratic political system, while ED may perform even by lacking expertise and savvy. Nevertheless, when the two cooperates, they should manage different, inter-connected and complementary realms. KS deals with social, cultural, educational, professional dimensions leading to politics, while ED endures the political system yielding improved conceptualizations for the former. Metaphorically speaking, KS is practice while ED is theory, alternately modeling each other in a complementary way, like cast iron scaife (charged with diamond powder) polishes diamonds. One needs diamonds or similar composites to burnish other diamonds, in a more or less substantial manner and level of priority.

Figure 4 undertakes the task of visually illustrating the theoretical balance of KS and ED, and the practical and undesirable situations that disturb this balance. The ideal situation is when KS and ED share a common ground and

DOI: 10.12948/issn14531305/18.4.2014.07

the former encompasses a larger field of human interest (e.g. social and cultural life subdues politics). There are situations when one of the two becomes more important (e.g. KS is disproportional larger than ED in case of disaster, when rapid actions should prevail discussions). Unwanted instances are those when either avidity or apathy characterize political process (e.g. the economical flourishing life make people withdraw from social debates).

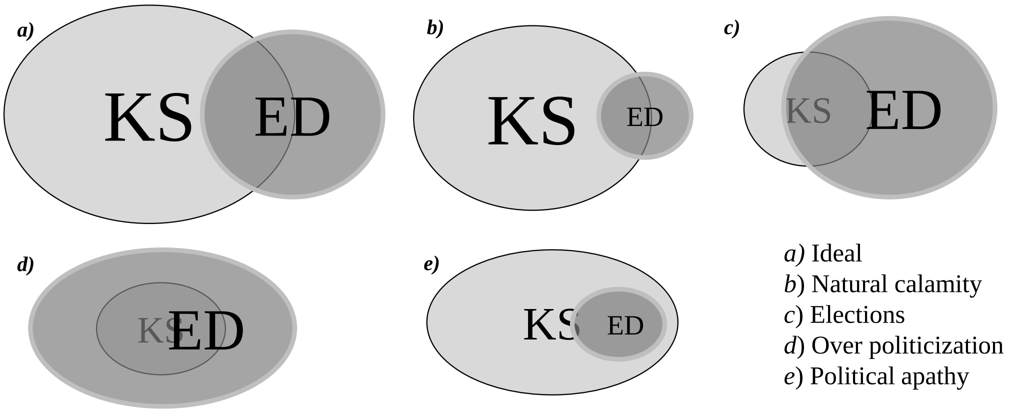


Fig. 4. Perspectives of Knowledge Society and E-democracy balance

ED and KS are complementary in MEKDS and, while ED springs from KS, the former reshapes the latter in permanent cyclical PDI processes. Although both have substantial idealized formation, the paper has shown some practices that proved a real tendency in envisioning them (especially KS).

After introducing elements of ED, discussing KS and ED in an ISOM approach and then delimiting the two, the next section will propose several technologies that accommodate AED.

3 Cloud and complementary technologies (CCT)

The previous section emphasized the interconnectivity (e.g. network of networks) and dynamics of KS (e.g. PDI) which require a technological approach that relies on flexibility, adaptability, stability etc. Considering these, on a large scale and with wide perspectives, the paradigm of Cloud Computing (or Cloud) seems appropriate for AED.

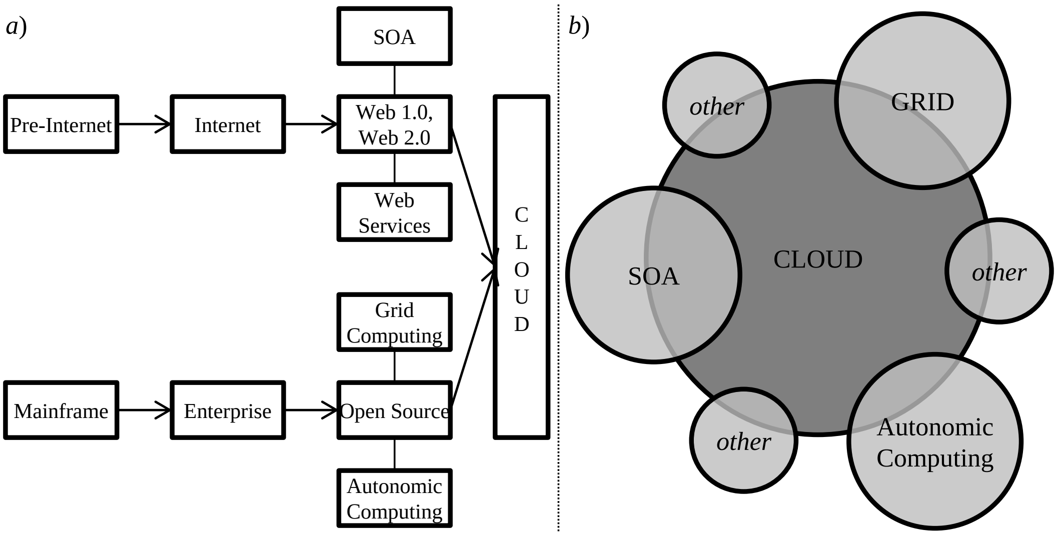


Fig. 5. Cloud's emergence

Figure 5 presents Cloud's emergence and we notice that is practically a conglomerate of other technologies and paradigms; however, it is not a simple aggregation of them, sharing

DOI: 10.12948/issn14531305/18.4.2014.07

only some areas (see example of item *b*). The most comprehensive definition of Cloud, according to the National Institute of Standards and Technology Laboratory, points out several characteristics [13]: on demand self-service, ubiquitous network access, location independence, resource pooling, rapid elasticity and measured service (pay per use). In network of networks society with large and small public spheres (e.g. KS), Cloud provides room for everybody, offering four deployment models: private (enterprise owned or leased), community (shared infrastructure), public (mega-infrastructure), hybrid (two or more clouds). On a joint approach of IBM, Google and Intel, Cloud provides the ability of end-user to benefit from technology without managing its complexity [14] and this is a key aspect for inclusion in KS.

There are three basic services model for Cloud: *i*) Software as a Service (SaaS), providing user applications over a network; *ii*) Platform as a Service (PaaS), deploying custom applications and *iii*) Infrastructure as a Service (IaaS), supplying computing resources.

Figure 6 illustrates cloud levels in a more detailed approach and from two perspectives: provider [13] and user [15]. The former is a bottom-up view with IaaS as foundation and the latter is a top-down view emphasizing the role of SaaS in commanding the other services.

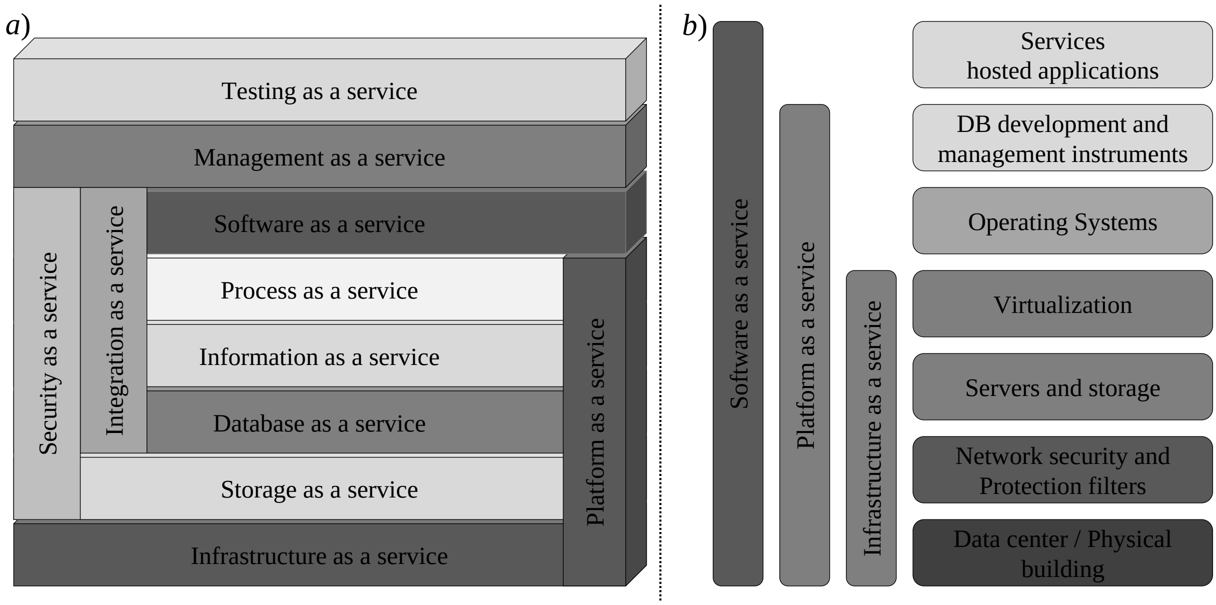


Fig. 6. Perspectives of Cloud: a) provider and b) user

Cloud is not a miraculous solution that solves all the problems and it still needs complementary technologies (CT) in order to address issues of AED (and other complex systems).

Table 1 illustrates most of the pros and cons of Cloud [13] and it is obvious that it is adequate for AED. CT will fix some of the issues that make Cloud unsuitable for AED: dependent and old applications, security level, control and cost.

DOI: 10.12948/issn14531305/18.4.2014.07

Table 1. Opportunity and inopportunity for Cloud

Adequate	Inadequate
independent processes, applications, data	dependent processes, applications, data
well defined integration points	poor defined integration points
low security level	high security level
healthy non-Cloud architecture	dysfunctional internal architecture
internet as desired platform	high control
high cost for own implementation	high cost
new applications	native interface dependent, old or inherited applications

Table 2 presents the benefits and drawbacks of Cloud Computing [13], which relate to pros and cons of Table 1 (some of the items are common to both flanks).

Table 2. Cloud: benefits and drawbacks

Benefits		Drawbacks	
Cost	conceptually, is always cheaper, physically not all the time	Security	not under direct control, a special regime may be needed for state secrets
Network	mix and match several services across the internet	Control	handle personal IT infrastructure aspects
Innovative	up to date; new, modern, good for startups, too	Cost	expensive applications on some occasions
Expandability	related to cost; fulfill the needs in terms of software and hardware resources	Openness	locked-in; transfer from one provider to the other is difficult
Speed to implementation	related to expandability; get quickly new configuration	Compliance	respecting the legislation for some data, documents
Ecologic ("green")	more environmentally green by sharing resources	Service-level agreement (SLA)	warranties from provider based on SLA criteria

In Table 1 and Table 2 we notice that security, control, cost and openness are some important issues to address. ED relies on transparency and citizenry control (see E-control in Figure 1) so the first two issues are negotiable on conceptual level. Normally, every cost should diminish with shared resources and even high investments, subject to permanent analysis and improvement, will bring a decent return on a medium and long term (see subsection 3.2). All elements of PDI need openness in a manner that overtakes transparency and requires direct access, a problem that this paper will address, through presentation of CT, partially in subsection 3.1 and highly in subsection 3.3.

3.1 Service-oriented architecture (SOA)

SOA is not a new concept and, although Web Services (WS) are its key factor nowadays, it has its roots in technologies like Common Object Request Broker Architecture (CORBA), Message-Oriented Middleware or Java Messaging Server (JMS). In a simplified approach, it is mostly a business architecture relying on well linked (i.e. defined processes), though loosely coupled (i.e. simplicity and autonomy in a wide range of particular services), black boxes (i.e. hidden complexity) components, assessing four key concepts [16]. They are: *i*) reusability (keeping old technology); *ii*) superior quality through safety, accuracy, predictability and regularity; *iii*) non-professional

DOI: 10.12948/issn14531305/18.4.2014.07

user access and *iv*) compliance and adaptability to external rules.

Using the same simplified approach, SOA has two major components: (Web) Service Client or Consumer (SC) and (Web) Service Provider (SP). However, due to transferred message complexity, scope or environment, new components (NC), belonging to SP or externalized in Cloud, may emerge. NC and their purposes are [16]: *i*) broker: mediator that keeps track of all WS on logical and conceptual level (e.g. identification, role, semantic, links of WS); *ii*) registry: a (physical) database of WS; *iii*) manager: registry or NC administrator; *iv*) security: rules and monitoring of WS transfer, protection of messages and NC and *v*) monitoring (useful for broker, security and manager): traffic, statistics, correlations and analysis.

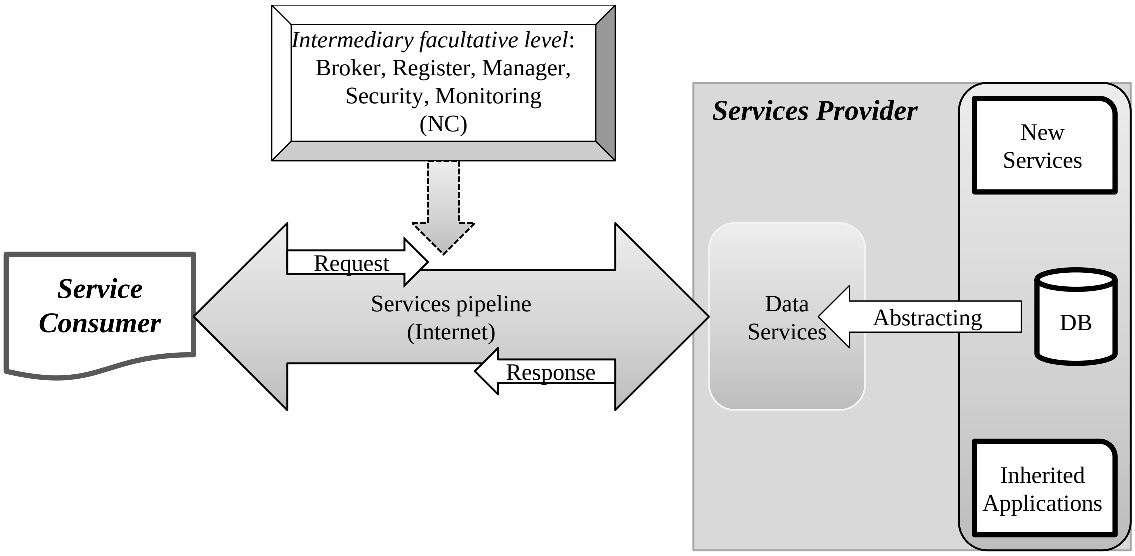


Fig. 7. SOA approach

Figure **7** illustrates SOA with NC, depending on three important functionalities or tiers: *i*) demand or request (from SC); *ii*) abstracting (blend data from different mutual unaware interfaces at middle tier) and *iii*) answer or response (from SP).

Table 3 succinctly presents several deployment models [17], through different methodologies that (auspiciously) prove that there is a multitude of approaches for communication in AED.

In the beginnings, SOA used to rely on CORBA or JMS, but nowadays WS absorb most of the market and this is the reason many take the former and the latter as analogous. Yet, they only share some common realms (not all WS are part of SOA and SOA may use other services) and the novelty WS has brought to SOA consists in decoupling from operating systems or specific (programming) languages with benefits on flexibility and adaptability to re-conceptualization. Three concepts are important on a WS [15]: *i*) atomicity or finer granularity means that a service is self-contained, independent from the state of other services; *ii*) composability is ability to (re)compose services that become compound, thus having a large granularity and *iii*) loosely coupling is a design approach where complexity is hidden and external behavior is available for other services.

Subsection 3.3 elaborates more on WS, but for the moment, we wrap up Cloud and SOA in a joint approach. SOA is an assembly of architectural methodologies that can provide Cloud instances of heavy business to external end-users, outside the perimeter of firewalled enterprise.

From AED point of view, Cloud provides (formatted) data for simple requests that stay in delimited area and SOA may mediate interconnectivity between clouds. In order to leverage Cloud through services, five concepts of SOA incremental analysis (SIA) are needed

[15]: *i*) designing small projects; *ii*) involving stakeholders; *iii*) postponed decision; *iv*) reducing drawbacks instead of driving on benefits and *v*) permanent design as response to change. Yet, the requests are well defined and standardized by SOA and Cloud, but hardly by end-user.

Table 3. SOA methodologies

Methodology	Description	Key Concepts
OASIS Reference Model for SOA (*SOA-RM*)	Proposal for a conceptualization of implementation, offering basic terminologies	- No modeling languages, no identification and composition; - Basic and abstract concepts
OASIS Architecture Foundation for SOA (*SOA-RFA*)	Abstract of business process from services perspective	- Basic concepts, UML2 - No service identification
Open Group SOA Ontology (*SOA Ontology*)	Standard defining terms, semantics, concepts, in a business and technological framework	- OWL, UML, basic concepts for implementation - No service identification
Service-oriented Modeling Framework (*SOMF*)	Specific model methodology for services analysis and identification	- Support for SoaML, service identification, analysis, design - No portability
Platform-independent Model for SOA (*PIM4SOA*)	Meta-model describing services, processes, control, communication and quality	- WSDL, XSD, QoS - meta-model
SOA Modeling Language (*SoaMl*), proposed by Object Management Group	Meta-model based on UML for design and specifications of implementation	- UML, QoS - defining SOA in detail
Service-oriented Modeling and Architecture (*SOMA*), proposed by IBM	Modeling technique for services' identification, specification, reusability, components and flows	- SoaML, QoS, identification, specifications and realization - business and engineering

Next subsection proposes a way of mediating user involvement as well as acquiescing SIA.

3.2 Agile Development

Before discussing Agile, let us introduce the concept of Lean architecture that will also help configuring AED by departing from computational information concepts towards wider informational and knowledge-oriented perspectives. Cloud and SOA are solid techniques that mostly deal with computers for the benefit of users, while Lean and Agile are ways of integrating human outsider approaches to system building, a necessity for AED that must not be a grant of programmers or computer scientists. Having its roots in Japanese culture and auto industry, Lean brings to (Agile) software development seven key concepts. They are [18]: *i*) deferring (classical) engineering concerns and paying attention to form; *ii*) empowerment of developers to make decisions regarding architecture; *iii*) using light application programming interfaces (APIs) and protocols; *iv*) simple documentation; *v*) people driven model; *vi*) collective planning and collaboration and *vii*) end-user mental model. The last concept is probably the most important for AED as a citizen-oriented endeavor. Taking into account Table 4 that presents Agile Manifesto's aims [19], we notice that both Lean and Agile support the notion of PDI involved also in building the digital democracy (not only in sociopolitical systems).

DOI: 10.12948/issn14531305/18.4.2014.07

Table 4. Agile declaration

Human elements	Relationship	**Technical elements**
Individuals and interaction	over	processes and tools
Working software		comprehensive documentation
Customer collaboration		contract negotiation
Responding to change		following a plan

While the relationship between elements of Table 4 is at row level in its original form, the herein approach stresses both the author and manifesto aggregated preference for the items of the left side, although there is value in the items on the right. There are 12 principles described in Agile Manifesto and key concepts for AED refer to frequently or adaptable to change continuous delivery of software, (face-to-face) collaboration between customers and developers, simplicity, self-improvement or sustainable development. Although the word semantic implies flexibility and speed, Agile, matching the pattern of MEDKS, is about self-organizing and feedback [18] or change through feedback by using four aspects of a shared mental model [20]: knowing, learning, understanding and executing. More, paradigm of Agile is a fit for KS requiring flexible strategies of testing, verification and validation [21], which also match the processes of the herein introduced MEDKS.

Figure 8 illustrates a comparison of Agile and two classical development methodologies, using a simplified ADCT model. It also describes a more detailed structure of Agile development which somehow proves that is not a radical innovation but an improvement.

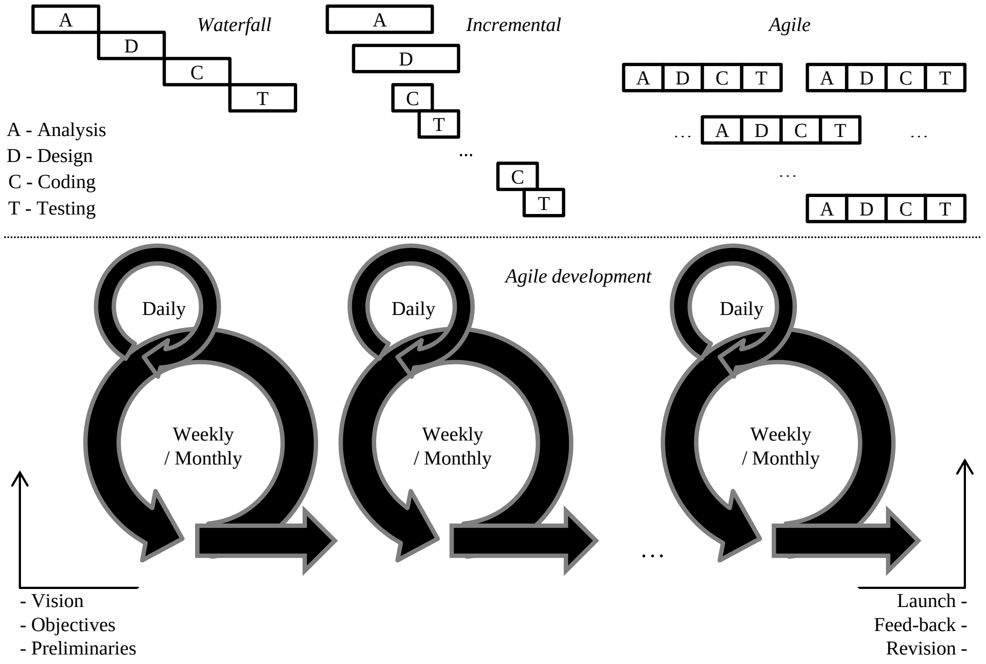


Fig. 8. Agile development

Actually, Agile is an umbrella of methodologies like Scrum, Extreme Programming (XP), Crystal, Lean etc., with the first two as the most prominent and mixed sometimes in practice [22] because Scrum exposes risks to improve project management and XP is about team activities practices [20]. More, practice prove to proponents of Agile-and-architecture incompatibility that Agile develops *i*) components such as communication among team members, inputs to subsequent design decisions, documenting design assumptions, evaluating design alternatives and even architectural documentation related to extended geographical distribution, multiple demands and beneficiaries [23] or *ii*) aspects like planning,

DOI: 10.12948/issn14531305/18.4.2014.07

leader briefings, reflexivity, cross-training and guided self-correction [20].

From SOA view and complying with new approach of SIA, Agile comes as an improvement based on adaptability, flexibility and empowerment by reasonably and dynamically modifying, on a temporal axe, the static robustness and resilience of the former, though keeping an architectonic perspective [23]. SOA (mocked at as Same Old Architecture), based on SIA, benefits from Agile that focuses on client and direct action. More, Lean adds new value to SOA and Agile by taking into consideration the end-user and focusing on the thinking process over complicated, but predictable, aspects [18].

We conclude this subsection by stating that, from ED and KS views, PDI may benefit not only from AED implementation using the herein discussed technologies, but at conceptual level it may import their patterns: participation of all stakeholders in a service-oriented cloud, agile deliberation between CDJ and lean inclusion of marginal citizens.

3.3 Web-Oriented Architecture (WOA), Semantic Web (SW) and Linked Data (LD)

WOA, an extension of and relying on SOA, refers the same concepts (e.g. reusability, interoperability, loose coupling, abstraction etc.), being an emergent global modular software architecture, but not defined by any standard body. It employs only WS and it has its roots in the architectural style of Representational Stateless Transfer (REST), in HyperText Transfer Protocol (HTTP) and Internationalized Resource Identifier (IRI) and in communication dependent on client state [24]. REST is a guide that provides some semantics to communication (using the verbs GET, POST, PUT and DELETE) based on four principles [25]: *i*) using IRI; *ii*) resources through representations; *iii*) self-descriptive messages and *iv*) hypermedia as the engine of application state (HATEOAS). In the beginnings of SOA, the standard of almost all WS heavily relied on Simple Object Access Protocol (SOAP), Web Service Description Language (WSDL) and Universal Description, Discovery and Integration (UDDI), but the latter, an intended WS registry, has been lately abandoned. While REST / HTTP WS (RHWS) may accept any protocol and is data-driven, using syntax light JSON as preferred data format (XML and YAML also work fine), SOAP and WSDL (SOWS) use only (heavy syntax) XML and become a burden because the latter obliterates the former's flexibility [26].

Figure 9 illustrates WOA in a simplified way with emphasis on three tiers: web server, WS and client browser, all of them complying with REST and HTTP 1.1 principles.

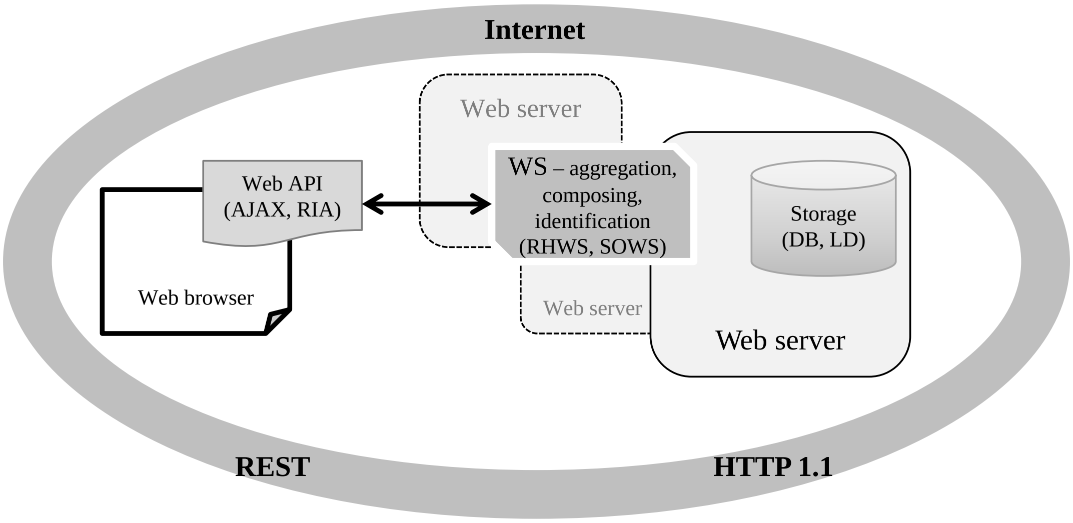


Fig. 9. Simplified WOA

DOI: 10.12948/issn14531305/18.4.2014.07

Although RHWS may rely on Web Application Description Language (WADL) that is optional and not very generous, it has an important feature that SOWS does not have: caching mechanism. Thus, not only that is more flexible, but also RHWS supports openness (a key concept for AED), while SOWS are suitable for APIs that require high security using a well defined standard (RHWS is hardly fit for this). RHWS has most of its consumers, if not all, bound to web browsers and this assures a high portability that leads to wide-open access.

REST provides some meaning on operations (using the verbs), but what is more important for end-user and computer is the semantic of data transferred on the internet. SW or Web 3.0 is not a substitute for Web 2.0, but a facility for knowledge management of content that is supposed to be machine-accessible through some meaning [27].

Figure 10 illustrates SW (item *a*) and two types of databases that SW could rely on: regular (item *b*) and LD (item *c*). While the former has a predefined structure that assures easy management of data and it benefits from established expertise, the latter is irregular, more difficult to manage, but it provides meaning of data for all systems that accede to a model such as WOA. Regular databases are locked-in and accessible through Cloud and SOA and sometimes WOA, while, on the contrary, LD is subject to transparency over WOA.

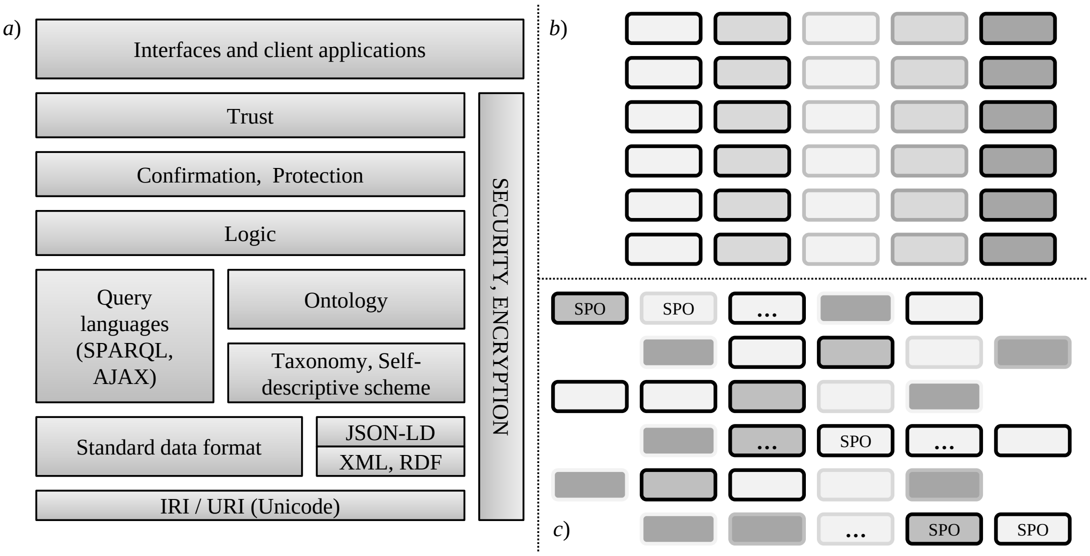


Fig. 10. WS, DB and LD

While LD repositories are irregular, see Figure 10.c, a LD record has the structure of a triple (a graph with three nodes and two arcs): Subject - Predicate - Object (SPO). IRI (unique decentralized global names and means of accessing information) is the foundation of LD that refers four principles [28]: *i*) use IRIs as names for things; *ii*) use HTTP IRIs to look up names; *iii*) use standards to provide useful information from looked up IRI and *iv*) include links to other IRI for discoverability. Regarding standards, we advocate for JSON-LD and JavaScript / AJAX and we have already done a comprehensive exploration on this field [29].

DOI: 10.12948/issn14531305/18.4.2014.07

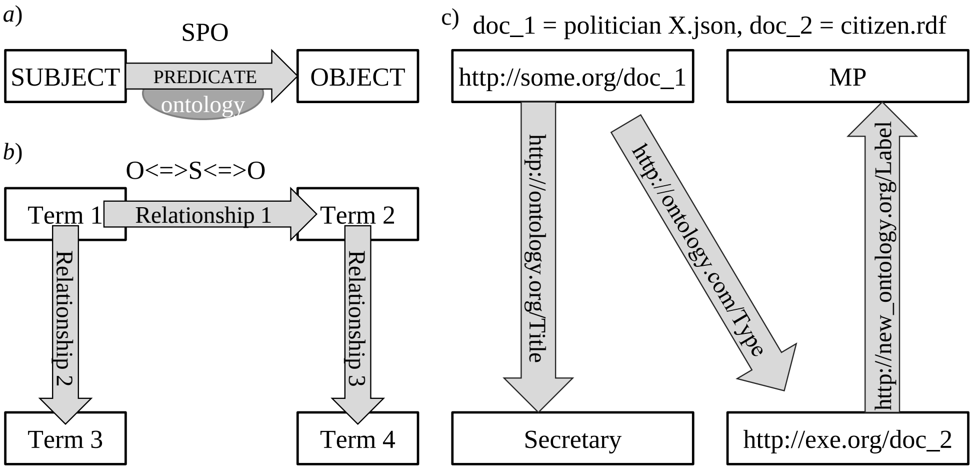


Fig. 11. Linked Data: SPO

Figure 11 illustrates SPO (item *a*), presents link-ability in LD, when *S* becomes *O* and vice versa (item *b*) and an example of LD - SPO (item *c*).

It is practical to use different IRIs for objects themselves and for documents that describe them and this coherence depends on three types of LD links: *i*) relational; *ii*) identity and *iii*) self-descriptive [28]. The web of linked data in a WOA paradigm provides a large flexible open database, but the management (especially querying data) is more complicated and this is another reason the old format of XML should make more room for JSON-LD [29].

This section has presented the technological aspects (i.e. CCT) supporting AED: Cloud, SOA, Agile, WOA, WS and LD. The next section will integrate them in a model fit for ED and KS.

4 CCT for KS and ED, an architecture proposal

AED actually represents the integration of functionalities and methodologies of CCT through two mixed perspectives: validated digital business organization systems and end-user web micro-entities. Cloud and SOA receive an openness-oriented enhancement from WOA with the goal of inter-connecting three types of applications. They are: *i*) Web API, browser application that can follow a paradigm such as client-server; *ii*) Client API, private application developed by business companies or any other organization (e.g. NGO) and *iii*) Cloud API, any application that either leverages Cloud or depends on Cloud (private, public, hybrid etc.).

Web APIs mediate data and information access that is subject to openness and does not require plenty of processing or transformation and aggregation. They easily rely on WOA and RHWS and probably do not ask for a high level of security; SOA and SOWS may also be an option, though hardly a choice. Client APIs stay in Cloud, in a client-server model or mostly following SOA and SOWS approach with two types of end-user: public (e.g. accessing a political application like E-voting) and private (e.g. enterprises or external users). Cloud APIs may rely on SOA and provide distribution of data and information for other APIs or they may follow SaaS approach. SaaS is a great opportunity as it supports ubiquitous devices with low processing power (smart phones, tablets etc.). Mobile devices seem to keep their ascending trend and, in response, more big companies move to Cloud some of their technologies [30].

Figure 12 illustrates CCT applied on KS, in a paradigm with one end-user. KS is all about openness without neglecting privacy. Thus, the end-user may have different statuses and roles that imply utilization of web application (e.g. news, feeds etc.), client application (e.g. financial statistics and analysis of some organization under EU umbrella) or SaaS for public information or comparative analysis of health systems at local level, for example.

DOI: 10.12948/issn14531305/18.4.2014.07

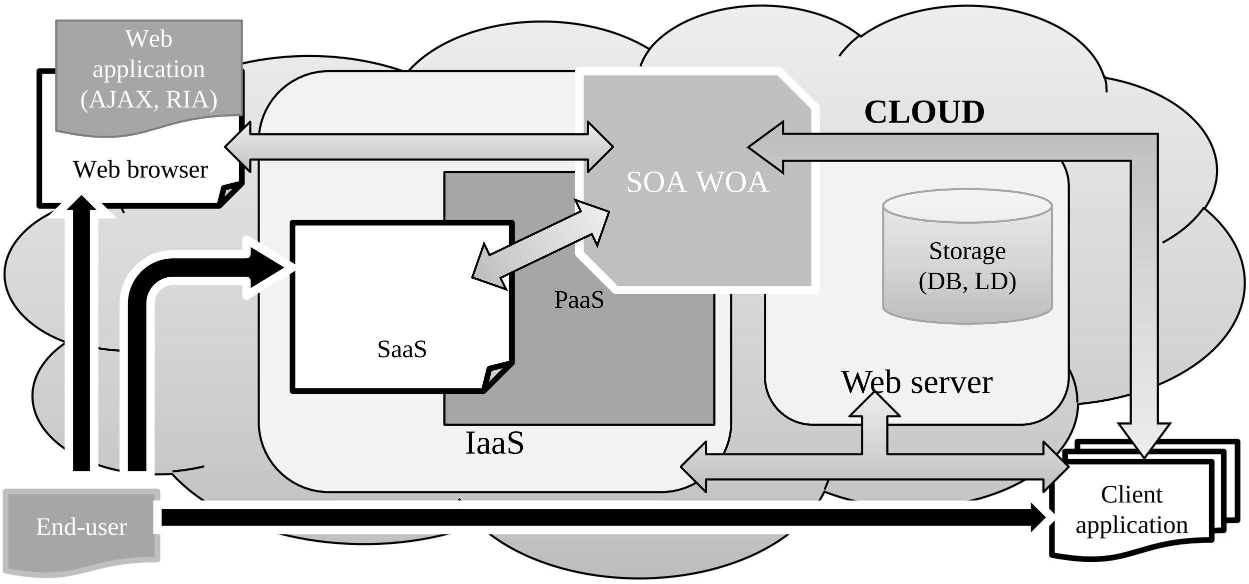


Fig. 12. CCT and KS

Things are more complicated for CCT and ED and Figure 13 illustrates their interaction. There are two classes of users: citizen and E-voter, with the latter as an extension of the former. Citizens participate actively through PDI (which may follow the paradigm of Agile) and has the characteristics of the end-user in KS, see Figure 12. Although it is not compulsory, citizens are political non-apathetic and they constantly debate not only at sociopolitical conceptual level but also, involving software developers, about the API functionalities that serve them as ED's stakeholders (e.g. through Agile). Moreover, when a citizen has also the assumed role of voter (again, not compulsory), it has the facility (a euphemism for obligation) of getting informed. Cloud and one or several SaaS provide info for E-voters and citizens, by casting data into PaaS that subsequently integrates them. In addition, SaaS facilitates the process of online voting, which requires a higher level of security (using SOWS, while RHWS would be the choice for other processes).

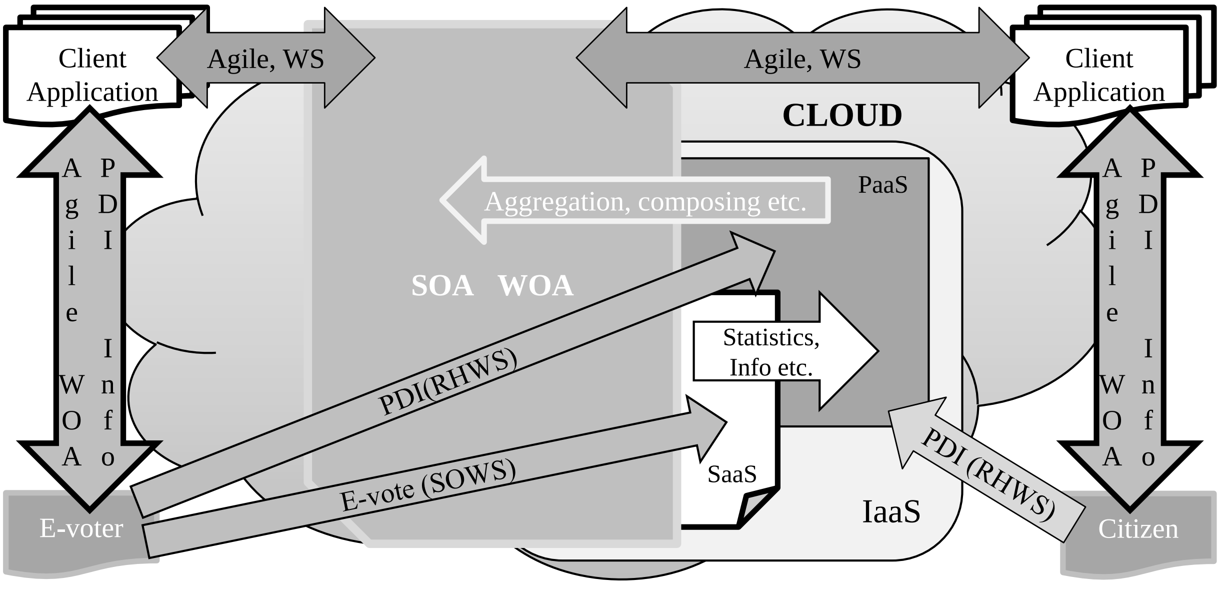


Fig. 13. Cloud and ED

DOI: 10.12948/issn14531305/18.4.2014.07

At conceptual point of view, Cloud encompasses the whole and, in addition, the notion of Web of Data gives a new flavor by disseminating the former to the end-user level through SOA and, especially, WOA. While CCT follow a less collaborative development model of APIs at KS level (though marketing strategies may function to identifying client needs), ED requires a feedback-oriented approach. Thus, through Agile (a concrete reliable and established model mapping the more abstract PDI) citizens cooperate with owners (private enterprises or public organizations) and developers to enhance or build new APIs. While for private APIs, cooperation between citizen and owner is not compulsory, if they are not sanctioned by public organizations, public APIs are definitely subject to citizen-oriented approach.

E-voting is an instrument that deserve a research on its own and we only point out some aspects that are important for ED at architectural level. There are many actors that participate in an electoral process (candidates, voters, governmental organizations, NGOs, PPs etc.) and the herein E-voting is more than just the activity of one-day voting process. Firstly, there are two main types of voting: E-referendum and electing representatives. Secondly, E-vote is part of a bigger picture whose name is E-control (see Figure 1), and this involves several APIs regarding analysis, candidate descriptions, archives etc. Thirdly, there are security and compliance issues and these two demand APIs for validations, testing, identity checking, E-voter training etc. While many info aspects of E-voting could be subject to private organization applications, the herein proposal is a multi-SaaS approach, which means that a PaaS/IaaS is the solution to implement applications, repositories, management systems etc. that support E-voting. E-voter is an extension of (E-)citizen that have one particular duty: to participate to the decision-making process directly (referendum) or indirectly (through electing representatives). Concluding the short review on E-voting, Cloud solutions (presumably IaaS, PaaS and SaaS) extended with SOA and SOWS are the probable approaches to implementing this.

Figure 14 is the final illustration of CCT that supports AED, integrating the previous two approaches of CCT for KS (see Figure 12) and CCT backing ED (see Figure 13).

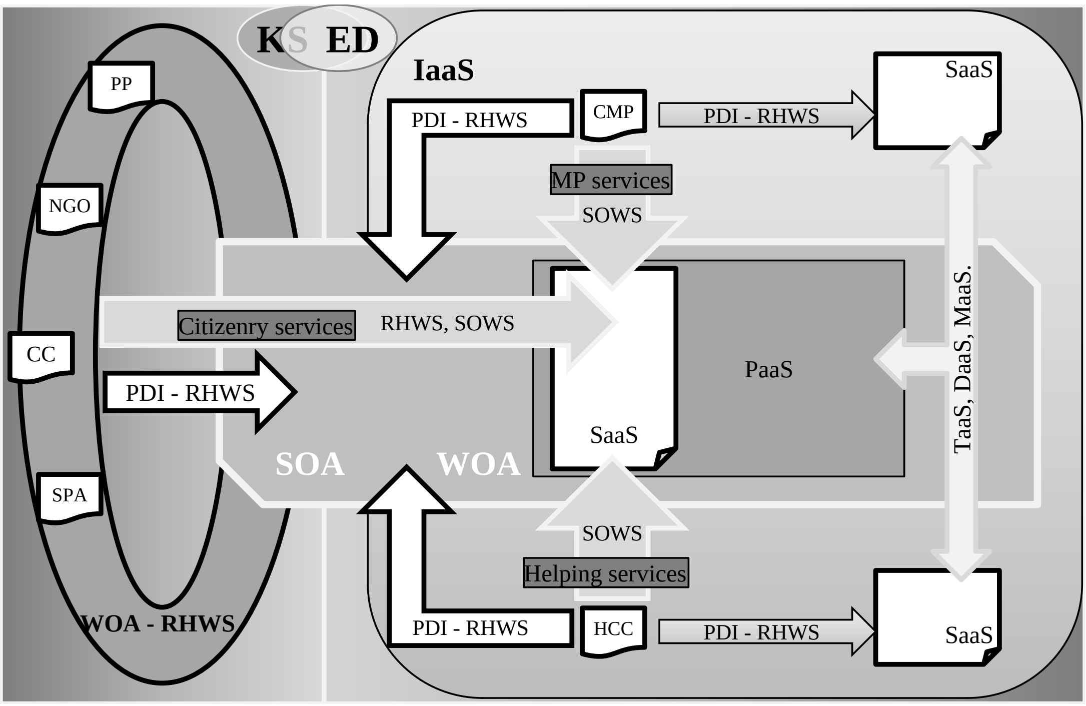


Fig. 14. E-democracy's architecture

There are two large conceptual-overlapped units of AED relying on CCT: KS and ED.

DOI: 10.12948/issn14531305/18.4.2014.07

Actors involved in PDI in non-political activities (e.g. NGO, SPA) communicate through WOA and RHWS in an open-access paradigm. The political activities of ED refer three actors HCC, CMP and CJR (at conceptual level they are similar to CDJ), but the latter uses the roles of the former two (being the supervisor of the political process, see Figure 1 and Figure 2). Political actors' services rely on SOA and SOWS as already explained, see Figure 13, but there are two other types of relationships. One regards PDI with the non-direct decision makers (i.e. SPA, PP) through open-access RHWS (for transparency). The other takes into consideration APIs that help PDI between the decision makers (i.e. HCC and CMP) and follow SaaS model. In addition, these SaaS APIs use facilities of IaaS (e.g. Testing as a Service (TaaS), Database as Service (DaaS), Management as a Service (MaaS), see Figure 6.a) derogating software development and maintenance to Cloud. Using PaaS, citizenry services access info by RHWS and vote (in a referendum) through SOWS.

On the other hand, when there is abstract PDI involved, this may also require Agile methods in software development and maintenance. While the herein proposed model of AED is also subject to change, every API or technology should follow a collaborative citizen-oriented approach such as Agile. AED or any other model of E-democracy may have one certainty (at least for the short and medium term up to new findings): it must follow a service-oriented model that provides room for reconfiguration, flexibility and adaptability.

5 Conclusions

This paper has undertaken the difficult task of conceiving an architectural model of digital democracy in knowledge society. While an architecture of such entity may not be subject of an independent research by an individual or an organization, we have identified some technology that better fits E-democracy and that are components of a trend described by human society. This endeavor only assembled validated methodologies and technologies belonging to digital world. Moreover, when we succinctly introduced an input-system-output model of E-democracy (i.e. MEDKS), we borrowed an artificial intelligence approach to describe the evolution of the model. Based on backpropagation analogy, MEDKS is subject to permanent optimization seeking for improvement through participation, deliberation and inclusion, which find an equivalent in a software development methodology (i.e. Agile).

Tested and validated business models (e.g. Cloud, SOA) and methodologies (e.g. Agile) or architecture styles (e.g. REST) borrowed from digital world serve the purpose of building an architectural model of E-democracy. Our model relies on two large components: knowledge society and E-democracy itself and we introduce an architectural pattern for each of them in order to subsequently combining them in one piece.

There are some compulsory aspects for E-democracy: transparency for politics and openness for public information systems, citizen-oriented application development based on feedback or flexibility and adaptability at architectural level. Giving the complexity of this endeavor, an implementation of a system similar to the herein proposed one should start from local community (laboratory-oriented) level, which will provide useful insights of technological nature. Moreover, we will observe social and political behavior of the community and we will measure the short and medium term outcomes (defined at community level) of such system in order to improve (as a permanent required task) E-democracy.

Acknowledgement

This work was cofinanced from the European Social Fund through Sectoral Operational Programme Human Resources Development 2007-2013, project number POSDRU/159/1.5/S/134197 „Performance and excellence in doctoral and postdoctoral research in Romanian economics science domain”.

References

[1] O. Turcoane, "E-Democracy's Representatives, A Proposal And A Case

DOI: 10.12948/issn14531305/18.4.2014.07

Study," Economy Informatics, vol. 13, no. 1, pp. 73-90, 2013.

[2] A. Maier, eDemocracy & eGovernment. Stages of a Democratic Knowledge Society, Heidelberg: Springer, 2012.

[3] Aristotel, Politica, Bucureşti: Univers Enciclopedic Gold, 2010.

[4] J.-J. Rousseau, Despre contractul social sau principiile dreptului politic, Filipeştii de Târg: Antet, 2006.

[5] J. Habermas, "Three Normative Models of Democracy," in Democracy and Difference. Contesting the Boundaries of the Political, S. Benhabib, Ed., New Jersey, Princeton University Press, 1996, pp. 21-30.

[6] A. Sen, The idea of justice, Cambridge: The Belknap Press of Harvard University, 2009.

[7] D. Easton, "An Approach to the Analysis of Political Systems," World Politics, vol. 9, no. 3, pp. 383-400, 1957.

[8] V. Kecman, "Multilayer Perceptrons," in Learning and Soft Computing: Support Vector Machines, Neural Networks and Fuzzy Logic Models, Cambridge, The MIT Press, 2001, pp. 255-312.

[9] UNESCO, Toward knowledge societies, Condé-sur-Noireau, France: Corlet, 2005.

[10] D. Haraway, "A Cyborg Manifesto: Science, technology and socialism-feminism in the late twentieth century," in The Cybercultures Reader, 2nd ed., D. Bell and B. M. Kennedy, Eds., London, Routledge, 2001, pp. 291-324.

[11] F. Webster, Theories of the Information Society, New York: Routledge, 2006.

[12] M. Castells, "The Network Society: from Knowledge to Policy," in The Network Society: From Knowledge to Policy, M. Castells and G. Cardoso, Eds., Washington, DC, Center for Transatlantic Relations, 2005, pp. 3-22.

[13] D. S. Linthicum, Cloud computing and SOA convergence in your enterprise: a step-by-step guide, Boston: Pearson Education, Inc, 2010.

[14] R. L. Krutz and R. D. Vines, Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Indianapolis: Wiley Publishing, Inc., 2010.

[15] D. K. Barry and D. Dick, Web Services, Service-Oriented Architectures, and Cloud Computing. The Savvy Manager's Guide, 2nd ed., Waltham: Morgan Kaufmann, 2013.

[16] J. Hurwitz, R. Bloor, M. Kaufman and F. Halper, Service Oriented Architecture For Dummies, 2nd ed., Indianapolis: Wiley Publishing, Inc., 2009.

[17] M. Mohammadi and M. Mukhtar, "A Review of SOA Modeling Approaches for Enterprise Information Systems," in The 4th International Conference on Electrical Engineering and Informatics (ICEEI 2013), Bangi, 2013.

[18] J. Coplien and G. Bjørnvig, Lean Architecture for Agile Software Development, Padstow, England: TJ International, 2010.

[19] M. Beedle, A. v. Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, K. Schwaber, J. Sutherland and D. Thomas, "Manifestul pentru dezvoltarea agilă de software," The Agile Alliance, 2001. [Online]. Available: http://agilemanifesto.org/iso/ro/. [Accessed 11 May 2014].

[20] X. Yu and S. Petter, "Understanding agile software development practices using shared mental models theory," Information and Software Technology, vol. 56, no. 8, p. 911–921, 2014.

[21] B. Ghilic-Micu, M. Stoica and M. Mircea, "Collaborative Environment and Agile Development," Informatica Economică, vol. 18, no. 2, pp. 32-41, 2014.

[22] J. Sutherland, "Agile Principles and Values," Microsoft, 2014. [Online]. Available: http://msdn.microsoft.com/en-us/library/dd997578.aspx. [Accessed 15 May 2014].

[23] P. Abrahamsson, M. A. Babar and P. Kruchten, "Agility and Architecture: Can They Coexist?," IEEE Software, vol. 27, no. 2, pp. 16-22, March-April 2010.

[24] D. Hinchcliffe, "Web-Oriented

Architecture (WOA)," QCon2009, 7 September 2009. [Online]. Available: http://www.infoq.com/presentations/Web-Oriented-Architecture-Dion-Hinchcliffe. [Accessed 1 May 2014].

[25] R. T. Fielding, "Architectural Styles and the Design of Network-based Software Architectures," 2000. [Online]. Available: https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf. [Accessed 14 12 2014].

[26] SoapUI, "SOAP vs. REST Challenges," 14 July 2014. [Online]. Available: http://www.soapui.org/The-World-Of-API-Testing/soap-vs-rest-challenges.html. [Accessed 20 October 2014].

[27] T. Berners-Lee, "Linked Data," 18 June 2009. [Online]. Available: http://www.w3.org/DesignIssues/LinkedData.html. [Accessed 13 May 2014].

[28] T. Heath and C. Bizer, Linked Data: Evolving the Web into a Global Data Space, J. Hendler, Ed., Ebook: Morgan & Claypool Publishers, 2011.

[29] O. Turcoane, "Linked data, JSON-LD and the semantics of cultural and scientific heritage," in The Fourth International Conference on Digital Presentation and Preservation of Cultural and Scientific Heritage, September 18-21, Veliko-Tarnovo, Bulgaria, 2014.

[30] I. Verdelis, "6 Mobile Trends To Watch For In 2014," 27 February 2014. [Online]. Available: http://www.fastcompany.com/3027080/productivity-in-motion/6-mobile-trends-to-watch-for-in-2014. [Accessed 20 October 2014].

Ovidiu TURCOANE graduated the Faculty of Cybernetics, Statistics and Economic Informatics in 2009. He holds a Masters Degree in Computer Science and a second specialization degree in Finances from 2011. He is currently studying, as a PhD Candidate of Doctoral School of Academy of Economic Studies, digital democracy in knowledge society. His fields of interest are methods and techniques of optimization in economics and social choice problems. He also studies fuzzy logics as an alternative to bivalent logic, especially in the field of development of a model of E-democracy that better fits inclusion, participation and deliberation.

DOI: 10.12948/issn14531305/18.4.2014.07