Big Iron, Big Data, and Big Identity

Craig A. LEE [a,1], Marcio ASSIS [b], Luiz F. BITTENCOURT [b], Stefano NATIVI [c] and Rafael TOLOSANA-CALASANZ [d]

[a] *The Aerospace Corporation*
[b] *University of Campinas*
[c] *National Research Council of Italy*
[d] *University of Zaragoza*

Abstract. While High-Performance Computing (HPC) typically focuses on very large, parallel machines, i.e., Big Iron, running massive numerical codes, the importance of extracting knowledge from massive amounts of information, i.e., Big Data, has been clearly recognized. While many massive data sets can be produced within a single administrative domain, many more massive data sets can be, and must be, assembled from multiple sources. Aggregating data from multiple sources can be a tedious task. First, the locations of the desired data must be known. Second, access to the data sets must be allowed. For publicly accessible data, this may not pose a serious problem. However, many application domains and user groups may wish to facilitate, and have some degree of control over, how their resources are discovered and shared. Such collaboration requirements are addressed by federation management technologies. In this paper, we argue that effective, widely-adopted federation management tools, i.e., Big Identity, are critical for enabling many Big Data applications, and will be central to how the Internet of Things is managed. To this end, we re-visit the NIST cloud deployment models to extract and identify the fundamental aspects of federation management: crossing trust boundaries, trust topologies, and deployment topologies. We then review possible barriers to adoption and relevant, existing tooling and standards to facilitate the emergence of a common practice for Big Identity.

Keywords. big data, identity, federation management, deployment models

1. Introduction

The need to share data, and computing resources in general, is fundamental. This need has driven the development of computing networks and the World Wide Web. All segments of society – academia, arts, business and government – increasingly rely on electronic communication. All of this communication and the devices involved are, in fact, converging into an *Internet of Things (IoT)*.

[1]Corresponding Author: The Aerospace Corporation M1-102, 2310 East El Segundo Blvd., El Segundo, CA 90245-4691, USA, E-Mail: lee@aero.org.

The *scale* of this electronic communication is global and will only become more pervasive. Clearly there must be an effective way to manage how humans and their devices communicate at this global scale. The notion of scale affects all aspects of computing. The term *Big Iron* has been used to denote massive parallel machines for running very large numerical codes. Not too long ago, the term *Big Data* was coined to denote the ability to examine massive amounts of data that were being made accessible online. To denote the machinery that will be necessary to securely manage the scale of communication and interaction among users and possible accessible resources, we coin the term *Big Identity*.

Clearly, though, Big Identity involves more than just identity credentials. Big Identity will involve methods for flexibly managing dynamic sets of users and resources across various administrative domains. This includes the creation and termination of these dynamic sets, which users and resources are members, and how discovery and resource access policies are defined and enforced. The context for managing a dynamic collaboration can be called a *federation*.

The goal of this paper then is to clearly define a general model of federation that identifies all fundamental requirements and capabilities. To do this, we revisit the NIST cloud deployment models since they are actually different instances of federation. By extracting a more general model, we can define a federation design space where different implementation and deployment approaches can be more readily evaluated and compared.

The realization of a global Big Identity solution will certainly face many *barriers to adoption*. A primary challenge will be the "chicken-and-egg" standards adoption problem. Like many distributed computing capabilities, federation management would be greatly facilitated by widely adopted standards, but nobody wants to adopt a standard that is not already widely adopted. Hence, from a practical perspective, we will consider different implementation and deployment approaches that will make it easier to adopt and use federation management tools, albeit in less general forms, that could nonetheless help facilitate the emergence of a dominant practice for federation management.

2. Application Domains

We begin by reviewing a number of application domains that are representative of the wide applicability and need for the kind of secure, flexible collaborations that will be enabled by Big Identity.

2.1. eScience

E-Science incorporates the need of computing to many research fields. A variety of applications generate unprecedented amount of data to be processed to help in science discoveries.

A notable experiment is the Large Hadron Collider (LHC), which generates on average 5PB of data per week [1]. The feasibility of filtering, processing, and storing such a large amount of data is only achieved through federated computing infrastructures, which can together have computing capacity to extract knowledge from it.

The Advance Proton Source experiment generates up to 1TB of data per day, and it must pass through several systems to be cataloged and analyzed. The amount of data combined with the different tools and systems can bring difficulties such as inefficiency, failures, and errors [2].

The climate change has brought the already important study of climate to the spotlight. The Wold Data Centre for Climate (WDCC) database has more than $1,000$ users and more than $4,000$ TB of information [2]. This resulted in $2,455,880$ GB of data downloaded in 2015. Climate data comes from sources scattered on and above the earth and owned by a variety of entities. Such a collaborative work illustrates the need of a reliable, global-scale dynamic identity management.

2.2. Health Care

As more and more data about patients are digitally stored and processed, medical records become a rich source of information for research and diagnosis. If, on the one hand, this brings many potential benefits to humans in general, on the other hand privacy is an increased concern. The medical information from patient data set belongs to him, but if shared using the proper means can benefit the patient and other people with similar conditions.

Data archiving and storage for medical information contains billions of images [3], many of them in very high resolutions. According to the Institute for Health Technology Transformation (IHT[2]), U.S. health care data alone reached 150 exabytes in 2011 [4]. A worldwide healthcare system that can dynamically and securely share health care information from real-time body sensors with medical databases, allowing expanded knowledge to be acquired about users medical conditions and treatments, brings both identity and big data management challenges.

2.3. Disaster and Emergency Response

Disaster and emergency response are all activities related to the management of calamities – those resulting from natural disasters (e.g. earthquakes, tsunamis, typhoons), as well as human actions (e.g. terrorism and negligence). These kinds of response efforts, especially international response efforts, could be considered a "poster child" application for federation management. Responding to disasters for any type can require the coordination of many stakeholders. These stakeholders can include local, state, and federal governmental agencies. Disasters may also strike with little or no warning, and could occur anywhere in the world. That is to say, there may be no warning of which stakeholders need to collaborate, or what resources they will need to share. Hence, there is a direct need for *on-demand* federation management.

There are multiple examples of projects responding to this need. In response to the Haiti earthquake disaster, the US government sponsored the Network-Centric Operations Industry Consortium (NCOIC) to demonstrate the use of a community cloud for sharing disaster response information, based on managing a stakeholder's access to cloud-based storage containers [5]. This capability

[2]source: https://www.dkrz.de/daten-en/wdcc/statistics/wdcc-statistics-2015

was generalized in the Keystone-based virtual organization management system (KeyVOMS) prototype, which manages access to arbitrary, application-level services based on users authorizations within a given virtual organization. [6]. Another example is the US-JAPAN consortium [7] formed by the U.S. National Science Foundation (NSF) and the Japan Science and Technology Agency (JST) that is promoting projects using Big Data to enhance critical information exchange and situational awareness. All of these projects will depend on some notion of identity management in various data domains whereby stakeholder access can be properly managed.

2.4. Global Earth Observation System of Systems (GEOSS)

Established in 2005, the Group on Earth Observation (GEO) is a voluntary partnership of governments and organizations that envisions a future wherein decisions and actions for the benefit of humankind are informed by coordinated, comprehensive and sustained Earth observations and information [8]. GEO Member governments include 102 nations and the European Commission, and 103 Participating Organizations comprised of international bodies with a mandate in Earth observations. Together, the GEO community is creating a Global Earth Observation System of Systems (GEOSS) that will link Earth observation resources world-wide across multiple Societal Benefit Areas (i.e. Biodiversity and Ecosystem Sustainability, Disaster Resilience, Energy and Mineral Resources Management, Food Security and Sustainable Agriculture, Infrastructure & Transportation Management, Public Health Surveillance, Sustainable Urban Development, Water Resources Management) and make those resources available for better informed decision-making [9].

To build the GEO Information system, the GEOSS program applies a *system of systems* approach: it consists of developing a central GEOSS Common Infrastructure (GCI) that, proactively, links together existing and planned information and processing systems around the world and supports the need for the development of new systems where gaps currently exist. GCI has been facing Big Data challenges [10] and is going to face Big Iron one to generate information and knowledge from observations [11] and contribute to the UN SDGs (Sustainable Development Goals), as required by GEO [9].

As depicted in Figure 1, the GEOSS Community environment considers many stakeholders (i.e. intermediate and final users, resource providers, GCI) who contribute and manage a large heterogeneity of resources (e.g. data, information, applications, services). For their integration and interoperability, a set of GEOSS Data Sharing and Management principles have been introduced; they advocate the use of permanent identifiers and entail the GCI to manage them, across the diverse organizations, to support virtual and transparent discovery, access and (re-)use.

2.5. Smart Electrical Grids

Smart grids represent an evolution of electrical networks towards improved energy efficiency, and controllability and manageability of the electrical resources [12].

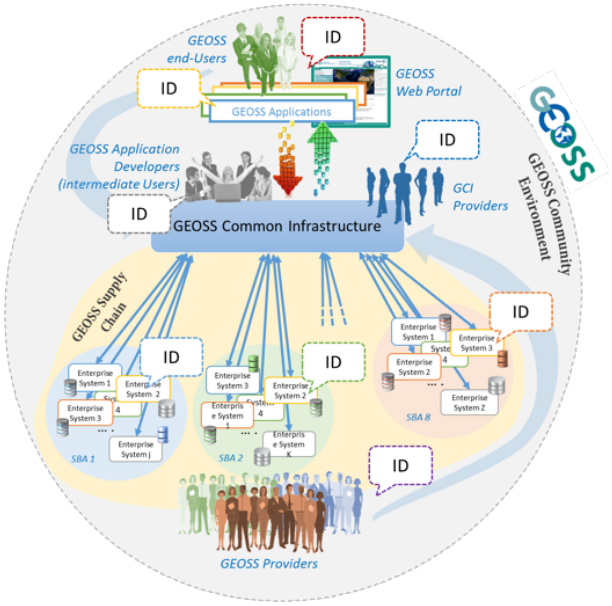


Figure 1. GEOSS Community environment and some of the possible identity.

Unlike current power networks, which were designed to distribute energy in a one-way direction from generator to consumers, smart grids are anticipated to enable a two-way transmission and distribution of energy: Consumers will also play an active role in the generation of energy, for instance by means of photovoltaic panels installed on building roofs, wind turbines, or many other sources.

Such a bidirectional flow of energy generation and consumption, however, requires a more fine-grained and (near) real-time monitoring of energy consumption, which involves the development of sophisticated and reliable communication networks for its realization. Smart meters, an electronic device that monitors consumption of electricity periodically, enable such a real-time monitoring. They differ from traditional metering in that they also support two-way communication with the meter.

Overall, from an architectural point of view [13], smart grids are composed of three main tiers: (a) The physical power tier that aims at the distribution and transmission of electrical energy. It comprises a wide number of resources and consumers. (b) The *Advanced Metering Infrastructure (AMI)* tier, which includes a collection of devices that record, collect, and analyze energy consumption, and interact with smart meters, enabling the two-way communication electric generators and consumers. (c) The application layer, which includes a plethora of applications for managing smart power networks. Many of them can be computationally intensive and often require a distributed computing infrastructure to provide results in a timely manner [14].

In all such applications, the AMI is the key enabling technology for real-time monitoring and bidirectional communication required by future smart grids, but, at the same time, it also raises a number of challenges from the security [15,16] and data privacy [17] perspectives. Electrical grids are critical infrastructures,

but smart electrical grids could present additional vulnerabilities. Associating a consumer's identity with extensive metering could enable usage patterns to be deduced, and opens the opportunity for fraud and theft. When multiple major electrical devices, such a smart air conditioner and a smart electric vehicle, are all managed as a group by an owner, this represents another federation use case with respect to an energy provider.

2.6. Smart Buildings

Real-time collection of measurements of a number of key variables within a building has also enabled intelligent energy consumption planning and management of energy in large building facilities. Building managers can now make use of complex automated control systems within buildings in order to optimize decisions and reduce energy consumption. Energy optimisation in this context therefore involves capturing data (e.g. every 10 or 15 minutes) from a variety of sensors and returning control set points to be implemented through building management systems [18].

The proposal in [18] for smart building automated management makes use of the EnergyPlus model. EnergyPlus building models consider a number of construction parameters and also monitored variables, such as temperature, humidity, or premise occupancy. Then, with all this information, parallel complex simulations need to be accomplished that as a result provide optimal energy controlled values. The execution of such simulations can be particularly computationally intensive as a significant number of computational resources are required. Indeed, the authors in [18] are making use of a cloud federation.

3. Requirements Analysis

All of these application domains require some form of identity and access management to support federation and collaboration, but in some cases with very different emphasis. *Data discovery* is a common problem throughout scientific communities. Science projects generate massive amounts of data. Making these data sets discoverable and accessible by other teams facilitates a much wider range of scientific inquiry, including lines of inquiry that may never have been thought of in years past. While many different tools have been built for cataloging data and making it searchable, doing this on a global scale under changing requirements means there can be inherent *semantic interoperability* issues. Do the metadata schemas for different catalogs have the same meaning? Having a common understanding for such metadata is also critical for establishing *data provenance*, i.e., establishing where data came from and how it was processed.

While data discovery is very important, it is actually an aspect of *service discovery*. Since all data in these scenarios are accessed through services, data discovery becomes a special case of service discovery. Querying a catalog for useful data requires knowing where the query service endpoint is, and knowing the query semantics it understands. Assuming useful data is found, the retrieval service semantics must also be understood.

Hence, service discovery, in the most general sense, requires knowing what the service does and the associated interaction semantics. Again, in the most general sense, federations can consist of *arbitrary* services at any level in the system stack. Like data services, cloud infrastructure services are just services with specific semantics – they are "factory" services that simpy produce other services of interest, e.g., creating a VM with an ssh daemon listening on port 22, or creating a storage container that responds to HTTP PUTs and GETs. Given this very general nature of services, it is straight-forward, then, that service discovery can involve the discovery of arbitrary sets of services that may be related for specific organizational purposes.

The notion of some sort of structured service discovery will be critical for the Internet of Things (IoT). While many of the things on the IoT will be statically managed by hand, the sheer potential scale of the IoT will motivate the development of discovery mechanisms. While the scale of integration among things in the IoT will be a challenge and goal by itself, most people, however, will actually be concerned with an *Internet of Important Things* [19]. That is to say, there will have to be methods to structure the discovery and accessibility of on-line resources. This structured discovery process will inherently involve *discovery policies*. Resource or service providers may wish only users with appropriate authorization attributes that are understood and trusted to be able to discover the provider's resources. All of these capabilities will hinge on the availability of Big Identity.

What does it mean to establish identity in a Big Identity environment? When considering *distributed* Big Data, users will come from many different administrative domains, with identity credentials issued by many different Identity Providers (IdPs). Hence, some notion of federated identity management is necessary, i.e., being able to validate and understand identity credentials regardless of where a user is from. If a Big Data or IoT environment is essentially within one administrative domain, then governance is much easier. GE's Predix system [20], for example, creates IoT environments by coupling distributed industrial machines in the field with commercial cloud computing (Cloud Foundry) using VPNs. While such approaches are clearly useful and cover a wide range of applications, they cannot be used in situations where the federation of identities is necessary.

While most times we may think of establishing identity as a binary decision (you either are or aren't who you say you are), there are other times where a user may wish to limit the number of identity and authorization attributes that are divulged or exposed to a service provider. This is certainly the case in health care and also in cases like smart metering and EV charging. For privacy reasons, any health care data used in scientific inquiries must be anonymized to prevent the release of personal identifying information. Managing large-scale charging of electric vehicles may require each vehicle to report a number of parameters, however, these parameters do not have to be associated with other identity attributes of the vehicle itself, of the vehicle's owner. Hence, Big Identity will also have to manage the exposure or release of identity attributes for any given authentication or authorization event.

Besides managing the scale and discoverability of resources, and identity attribute exposure, the effectiveness of any Big Identity mechanisms will depend on

the establishment of *trust relationships* among relevant users, IdPs and SPs for any given application domain. Establishing and maintaining such trust relationships ahead of need is essentially the purpose of *trust federations*. The e-science community has an important precedent in the *International Global Trust Federation* [21]. The IGTF defines requirements for PKI Certificate Authority (CA) operation. When an IGTF member demonstrates that their PKI CA is in compliance, other members will trust certificates signed by that CA. It is easy to extend the model to other application domains. The international disaster response community, for example, could very well benefit from an international disaster trust federation whereby when a disaster strikes, the member stakeholders involved could quickly assemble into a response team. Within this response team, all data and service access policies would be enforced according to trust federation rules.

4. General Federation Requirements

The entire federation management design space can be defined by a set of general federation requirements. Based on the requirements defined in [22], we briefly review them and classify such requirements into two sets: (a) discovery; and (b) access.

4.1. Discovery

Discovery includes federation discovery, semantics discovery and interoperability, and federated resource discovery:

- *Federation Discovery.* One must know about a federation in order to participate in it. While many federations will become known through traditional out-of-band methods, as they become more widely deployed and used, there will be a greater need for online methods of cataloging and discovery. Discovery mechanisms can include preferences and characteristics that can be filtered to enhance federation discovery results that are most interesting in a given context.
- *Semantics Discovery and Interoperability.* Once a federation is known, the *semantics* of how the federation operates must also be known. That is to say, there must be some set of *commonly understood authorization attributes* and *joint policies* that are coordinated among participants. As the federation evolves, its semantics can change. Discovery mechanisms must be able not only to allow new participants to learn the federation semantics, but also to dynamically update it to maintain interoperability and semantic understanding among current participants.
- *Federated Resource Discovery.* Once a participant joins a federation, it must be able to discover all available resources within that federation. Clearly some type of *service catalog* and *search capability* must be maintained. However, a member should only be able to discover only those resources for which they have authorization to use.

From a practical perspective, most federation discovery and semantics discovery will be done using traditional out-of-band methods. However, these requirements are essentially the same as for *semantic web* [23] and *semantic grids* [24], and the same approaches could be applied. Federated resource discovery could be manage through a number of catalog, repository or database methods.

4.2. Access

Once all necessary discovery has been done, user access to any federated resources must be managed. This has the following requirements:

- *Membership, Governance and Trust.* Since a federation is a set of collaborating sites, there is inherently some notion of *membership*. This, in turn, implies there is a *governance policy* and *mechanism* whereby membership is decided. All federated governance is underpinned by the concept of *trust* where a protocol is used to determine that the necessary trust conditions exist. Membership governance and trust can be supported by policies and mechanisms that incentivize participants to share resources to avoid free-riders [25] and improve the federation capacity.
- *Federated Identity Management.* Once a federation has been established, virtual organization (VO) membership will have to present identity credentials to Service Providers (SPs). Since these identity credentials may come from different Identity Providers (IdPs), there must be some way of knowing how to validate the credentials and if the IdP is trusted.
- *Federated Resource Access.* Once resources have been discovered, a user selects a resource of interest. To enable this utilization of resources, the users need to obtain access credentials to the relevant domains. The service owner must then know how to validate the presented credentials, which may have been issued by different IdPs. Once identity has been established, credentials verified, and authorization attributes extracted, the service owner must decide whether the user is authorized to use the resources as requested.

Addressing these requirements in any practical implementation will certainly have a number of implementation requirements. Access monitoring is needed for accounting purposes. Access to federated resources may be subject to limits or economic models that improve resource sharing and utilization [26]. If authorized, resource access and use may need to be reported to federation accounting mechanisms for future reference or processing. Fault tolerance will also be important. Since federation management systems will be inherently distributed, many of the same fault tolerance techniques for clouds and networks will be applicable [27].

5. Federation Deployment Models

Given these fundamental federation requirements, how can they be realized? What are the possible federation deployment models? In 2011, NIST officially published a definition of cloud computing that included cloud deployment models [28]. We will begin by analyzing these deployment models since, as we shall see, they are actually just specific use cases of federation deployment models.

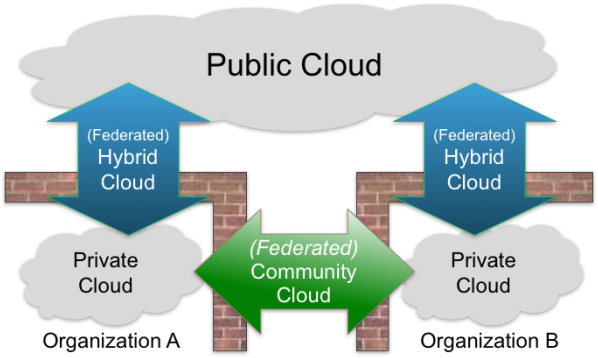


Figure 2. The NIST Cloud Deployment Models.

5.1. Analyzing the NIST Cloud Deployment Models

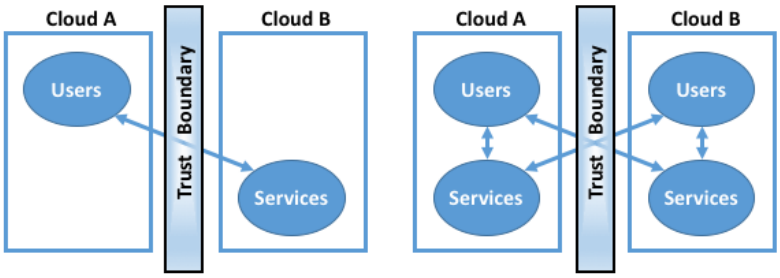
The NIST deployment models of *private, public, community*, and *hybrid* clouds are well-known and established terms, and are illustrated in Figure 2. First, we argue that the distinction between private and public clouds is really a relevant one based on different *governance properties*. Commercial public clouds can be said to have the weakest membership requirement: a valid credit card. A private cloud could be said to have much stronger membership requirements. Private cloud membership must already be a member of a "parent" organization. A private cloud and its provisioned resources are also not accessible from the open Internet, but rather only from a specific subnet behind a firewall, both managed by the parent organization. The parent organization may also define any other membership properties that they want, e.g., prior membership in a particular business unit or active project.

Second, we argue that both hybrid and community clouds are instances of cloud federation. The primary difference concerns the governance properties of the clouds being federated. In both instances, a *trust boundary* is being crossed. One side must trust the other side with regards to being an IdP that will vouch for its users, or being a service provider, or both. If two parties in a pair-wise federation provide both users and services to each other, this can be called a *symmetric* federation. If only users or services are being made available, but not both, this can be called an *asymmetric* federation.

In either case, when crossing a trust boundary, the federating parties should have an agreed upon trust relationship concerning their respective governance properties. When a community cloud is formed from two (or more) private clouds, the parties may have an out-of-band understanding of which potential users may be admitted, and the resources are to be made available. There may also be some agreement concerning resource discovery and usage policies. While not strictly necessary, most community clouds will be symmetric.

Hybrid clouds, on the other hand, are asymmetric federations: the public cloud is only provisioning resources for the private cloud users. Also, a public cloud does not have to be a *commercial* public cloud. It could be any cloud provider that makes resources available to the client. Depending on the relationship between the provider and client (e.g., both are part of some larger organization), there

Figure 3. The Private-Public Cloud Governance Spectrum.



(a) Asymmetric federation, i.e., a hybrid cloud.

(b) Symmetric federation, i.e., a community cloud.

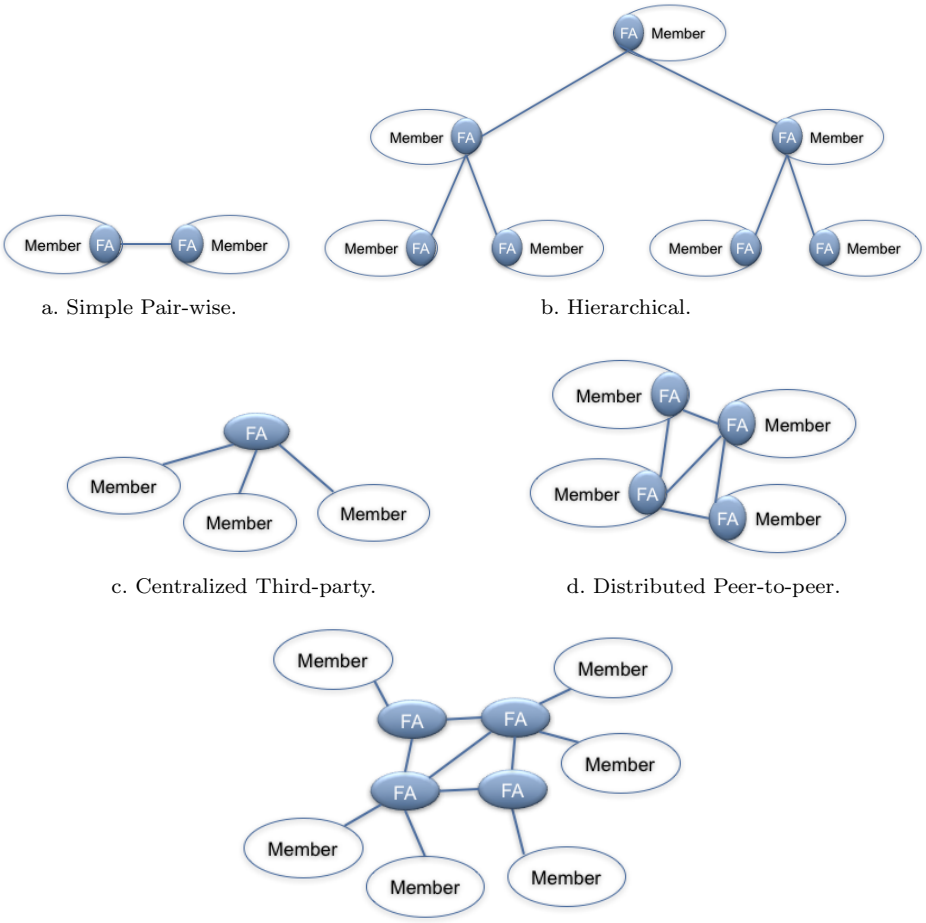
Figure 4. Federation Trust Topologies.

may be some agreement on potential users and resource discovery and access policies. With a commercial public cloud provider, any such prior relationships would probably not exist.

Hence, to summarize, we argue that the fundamental properties of cloud deployment models are (1) *Membership/Governance*, and (2) *Trust Topology*. As illustrated in Figure 3, NIST private and commercial public clouds are really two ends of a spectrum. Private clouds have the most restrictive membership requirements since they are restricted to a specific, relatively small group of users. However, private cloud governance is also the most flexible since the private cloud operators can be more autonomous, and a smaller user base is involved. Commercial public clouds, on the other hand, have the least restrictive membership requirements – anybody with a valid credit card can get an account. Commercial public cloud governance, however, is the least flexible since it is intended to serve an enormous user base. Any potential user must either accept or decline the provider's terms of use without any discussion or ability to negotiate.

In the middle are *organizational clouds* (for lack of a better term). These are clouds that are not commercial public clouds, yet are accessible to a larger potential user base than the most restrictive private clouds. An example might be a government agency that provides cloud resources for other areas of the government that might nonetheless operate their own clouds. In terms of crossing a trust boundary, the distinction between private and public is really a *relative* distinction. The membership and governance issues concerning federation of two clouds depends on where they sit on this spectrum.

Federation trust topologies are illustrated in Figure 4. In an asymmetric federation (Figure 4a), the users of one cloud can access the services of another cloud, but not vice-versa. This defines a hybrid cloud. In a symmetric federation (Fig-



a. Simple Pair-wise.

b. Hierarchical.

c. Centralized Third-party.

d. Distributed Peer-to-peer.

e. Brokers, Proxies, and Gateways.

Figure 5. Federation Deployment Topologies.

ure 4b), the users of both clouds can access the services of the other cloud. This defines a community cloud. (We note that services that call other services could be considered a user, but this does not affect the intended distinction between symmetric and asymmetric federations.)

5.2. Federation Deployment Topologies

Having identified governance compatibility and symmetry/asymmetry as fundamental aspects of federation, we now recognize that federations can be deployed in different *topologies* [22]. These deployment topologies vary according to the properties of *organization* and access. If we assume that every entity, or *member*, participating in a federation does so through a *federation agent (FA)*, then it is possible to illustrate these deployment topologies in Figure 5, and summarize them here:

a) *Simple Pair-wise Federation.* The simplest and smallest scale deployment model is between just two sites that is manually managed by administrators who establish trust out-of-band. As noted above, federation can be either *symmetric* or *asymmetric*, i.e., where a site provides either users or resources, but not both. Cloud-bursting is a key example of an asymmetric federation.

b) *Hierarchical Federation.* Federations can also be managed hierarchically, where policies "flow down" from the root through parent/child relationships. In hierarchical federations, the child's policies must be considered consistent with the parent, even though they may be tailored for the child's context.

c) *Centralized, Third-party Federations.* In this deployment topology, all information necessary to manage a federation is maintained in a *centralized, trusted third-party.* This third-party records which sites are participating in a federation, i.e., which resource services are being made available, and which users from which sites can use them. They could also implement mediation and adaptation tasks, where necessary.

d) *Distributed, Peer-to-Peer Federations.* All federation state could also be managed in a distributed, peer-to-peer manner. In this topology, federation state may be partitioned and replicated among the participating peers.

e) *Brokers, Proxies and Gateways.* This deployment topology relies on an *intermediary* between any given site and everybody else. As an extension of a trusted, third-party, the notion here is that multiple brokers, proxies or gateways communicate among themselves, while supporting multiple Keystone federation clients.

5.3. Scale

Simple scale will be an issue in each of these deployment topologies (except for, obviously, simple pair-wise federations). The size of centralized, third-party federations will be limited by the capacity of the one, centralized server. The others, however, will have no structural bound.

We can consider that cloud federations could be scaled out *horizontally* or *vertically.* In a horizontal scale, resources may increase or decrease within the same stack (IaaS - IaaS for example). An adaptation of the requirements is necessary for the horizontal scale. For example, in an environment with IaaS vertical scale providers may offer different types of flavors, so adaptability in a number of resources is required to use a distinct flavor that meets the request needs. In the second mode, the vertical scale, there is a constant sharing of services belong to different classes (e.g. IaaS - PaaS). In this case, a converting of the request is required because different classes have different properties.

A federation at a global scale, i.e., an *intercloud*, could perhaps use different organization and access methods that are open-ended among an essentially unbounded number of sites. By analogy with the Internet, the Intercloud is inherently distributed with no centralized point of control, and is essentially unknowable with certainty. The emergence of an *Internet of Things* will only increase the scale of resources that will need to be organized and managed through federation technologies.

6. Achieving a Common Practice for Big Identity

While the fundamental requirements and implementation approaches for federation management can be clearly identified, the establishment of a dominant best practice for Big Identity is a separate issue. A wide-spread, dominant practice would certainly require common standards, but this presents an inherent chicken-and-egg problem: users don't want to adopt a standard before it is widely adopted, but a standard will never be widely adopted until users start adopting it. To facilitate the emergence of a dominant best practice in the long term, we must find ways to avoid or ameliorate this circular dependency in the near-term.

6.1. What Are the Barriers to Adoption?

Large-scale adoption means (a) the same software (or interoperable implementations of the same standard) running almost everywhere, and (b) that software is being maintained by the operators at each site. Hence, adopting a standard can require a significant investment of resources. Potential adopters may not want to risk that investment if they feel they may not realize a tangible benefit in a reasonable period of time; besides, they do not want to limit their system evolvability. How can these requirements be avoided or ameliorated? How can (a) the required investment be reduced, or (b) the tangible benefit be made more feasible, or (c) how can the time period be reduced, and (d) the capacity of a system for adaptive evolution be preserved?

In the case of federation management, the necessary investment can be reduced by using a *brokerage approach*. Rather than requiring that an organization on-board and maintain a standards-based capability themselves, a smaller number of federation brokers could be deployed and maintained by organizations that have the resources and desire to promote such an approach. In a nutshell, a broker is an intermediary software that assists a client application to navigate through a complex supply environment of many options. Hence, an organization using a broker can incur much less direct costs, but at the expense of using a system that not integrated into their own environment and over which they may have little control. Trust and governance aspects must be considered [29].

The Broker model is commonly used to structure distributed systems with decoupled components, which interact by remote service invocations. Broker middleware is responsible for the coordination of communication among components: it forwards requests and transmits results and exceptions. Using the Broker paradigm means that no other component other than the broker needs to focus on low-level inter-process-communication. Thus brokering middleware can be used to add functionality to the exchange of information to a relatively unstructured and uncoordinated set of components containing data sets, for example.

This is essentially the experience of the Global Earth Observation System of Systems (GEOSS) [30]. Rather than requiring all members to install and run the same software everywhere, the GEOSS Brokering Framework (i.e. the Discovery and Access Broker – DAB) was deployed [10]. This enables participants to simply register their data sets to make them available to other users. The GEOSS Brokering Framework supports a number of brokering services required

by the community: a Discovery broker, an Access broker, a Semantic Discovery broker, a Quality broker, a Policy broker, and finally a Business Process broker. By presenting a lower barrier to adoption, the GEOSS Broker Framework is now managing over a million geospatial data sets.

In terms of what "tangible benefits" or "reasonable time periods" might be, these will differ for different organizations – considering also the maturity level of their interoperability technology. Benefits could be realized as either direct monetary benefits or operational benefits. It is probably safe to say that corporations will not be able to make a profit marketing and supporting federation tools until there is an economically self-sufficient market. However, many organizations (commercial or otherwise) may realize operational benefits by being able to more effectively manage their partner collaborations. The smaller an organization and its circle of collaborators are, the less time that it will take to realize operational benefits.

Hence, the way forward for establishing Big Identity starts by identifying groups that are large enough to have definite operational needs for federation, yet are small enough to make the adoption process feasible within a reasonable time period. Some possible groups include:

- *Scientific organizations.* National and international scientific organizations have long had a need for collaboration, as demonstrated by the *grid computing era*. In fact, the experience and knowledge gained from those efforts are largely applicable to the cloud computing arena.
- *Specific government organizations.* Larger organizations, such as governments, can actually make top-down decisions concerning their own use of collaboration technologies to address internal requirements. While using commercial, off-the-shelf (COTS) software is usually desireable, more immediate collaboration needs may merit the development of a government capability.
- *Non-Governmental Organizations (NGOs).* While NGOs may typically have fewer resources to bring to bear, they nonetheless may have far reaching collaboration requirements. For example, International disaster response efforts have been called the "poster child" of federation use cases since they may require the collaboration of a set of NGO stakeholders with little or no advance warning.
- *Other niche markets.* Generally speaking, there may also be what are considered niche markets where its easier to get everyone to adopt that same design conventions and approach to federation.

Finally, we wish to note that the use of *open source software* may also facilitate the development and adoption of federation technologies. One of the general advantages of open source software is that development costs are spread over a number of organizations. Such collaborative development can also promote the growth of a user community around the emerging capability.

6.2. Relevant Tooling and Standards

We clearly recognize that there are a number of existing, relevant tools and standards for federation management that will serve as a starting point for achieving a common practice for Big Identity. These have been more thoroughly reviewed in [31] and [22], so we will only summarize here.

Discovery is a fundamental function for all distributed systems and has been addressed by many standards targeting different usage scenarios. Service discovery in local environments has many examples. The Dynamic Host Configuration Protocol (DHCP) Discovery protocol enables a client device to find a DHCP server. The Bluetooth Service Discovery Protocol enables local bluetooth devices to discover what each other can do. Similarly the Simple Service Discovery Protocol is the discovery service for the Universal Plug and Play (UPnP) protocol stack [32] and is intended for residential or other small, local environments. The Service Location Protocol [33] is another discovery protocol for devices to announce services in larger enterprise environments. Importantly SLP devices and services exist within one or more *scopes*. A device in one scope cannot discover a service in a different scope.

Discovery was also a central function in the web services arena. The Universal Description Discovery Integration (UDDI) standard [34] provided an XML-based registry of services. The Web Service Inspection Language (WS-Inspection) [35] could be used to publish additional information to facilitate discovery. The Web Services Dynamic Discovery protocol (WS-Discovery) [36] provided a multicast-based discovery service for local environments that did not need centralized registries. More widely used, though, is the Lightweight Directory Access Protocol (LDAP) [37].

Of more importance to this discussion, however, are examples of service discovery in larger, distributed environments. XEP-0347 [38] is the discovery service for the eXtensible Messaging and Presence Protocol (XMPP). While originally built to support chat services, XMPP is being applied to IoT applications. To enable a "thing" to find an XMPP server, XEP-0347 supports the DHCP, multicast DNS, and SSDP/UPnP discovery protocols. Perhaps most relevant is XRD-based Service Discovery [39]. This uses XDRS (eXtensible Resource Descriptor Sequence) documents to resolve XRIs (eXtensible Resource Identifiers). This mechanism can be used for many purposes, such as resolving a user's OpenID identifier to discover the location of the OpenID identity provider.

While such standards and tools are valuable, outstanding questions exist concerning the ability to define and enforce arbitrary, user-defined discovery policies for arbitrary resource types. This must also be at the scale of larger peer-to-peer federations and interclouds.

Semantic discovery could be supported by tools such as Resource Description Framework (RDF) [40], Web Ontology Language (OWL) [41] and SPARQL (a recursive acronym for SPARQL Protocol and RDF Query Language) [42]. Clearly though, applying these tools at the scale of a global intercloud or IoT would require the development of widely adopted schemas and ontologies.

Governance will certainly entail auditing and accounting. In a distributed environment, where delegation of trust may have created chains of custody, au-

diting and accounting will be more complicated. To possibly map events back to their originating identities at originating sites, the DMTF has defined the Cloud Auditing Data Federation standard [43].

Federations based on identity and membership should eventually be able to manage their network traffic through *software-defined networks (SDN)*. This could be supported by Open vSwitch [44] and Open vSwitch [45]. A high-level SDN architecture has also been produced by the Open Networking Foundation [46]. While technically not a standard, the *slice* concept from the Global Environment for Network Innovations (GENI) could be also used [47].

Simply managing trust relationships is central. WS-Federation [48] specifies the brokering of identities, attribute discovery and retrieval, and the secure transport of claims among realms. To accomplish this, WS-Trust [49] is used where incoming messages must be able to *prove a set of claims* to be trusted, such as name or possession of a key, permission, or capability. It is also possible to manage trust using *reputation systems* [50] where the "opinions of others" can be cast into more formal metrics [51]. Trust can also be established through social networks [52], where *friend-of-a-friend* relationships are used to build *trust graphs*. As the scale of federations increase, the use of such decentralized approaches will become concomitantly necessary.

With regards to federated identity management, standards such as OpenID [53] and OpenID Connect [54] are relevant. The widely used X.509 Public Key Infrastructure [55] is also relevant, especially when used in conjunction with *proxy certificates* [56] to enable the delegation of trust and the creation of chains of trust. These tools can be used with the Security Assertion Markup Language (SAML) [57] to define authentication and attribute assections in XML, along with protocols for their exchange.

Federated resource access can be manged through the use of the eXtensible Access Control Markup Language (XACML) [58], which is often used in conjunction with SAML. OAuth [59] is another access control protocol that enables the delegation of trust. We note that OpenID Connect is a profile of OAuth v2 being defined specifically to support federated identity management and single sign-on.

Finally we observe that despite the many existing, relevant tools and standards, there are really no established *user-facing* abstractions or standards for managing federations. Many relevant "piece part" standards are available, but these have yet to be organized into a profile and used in an abstraction that is easy for ordinary users to use. We argue that the *virtual organization* concept is a good candidate for such a user-facing model [22].

7. Conclusions

With the unprecedent generation of data in science and engineering, and with the proliferation of instruments and sensors, a number of applications have emerged in different areas such as eScience, health care, disaster and emergency response, Earth observation, or smart grids and buildings. In this paper, we argue that effective, widely-adopted federation management tools, i.e., *Big Identity*, are critical for enabling many Big Data applications.

Such applications often require managing the scale and discoverability of resources, and a sophisticated management of the identity of the generated data: (a) In some cases, they must consider identity as a binary property; (b) while there are other times, where a user may wish to limit the number of identity and authorization attributes that are divulged or exposed to a service provider. This is the case in health care and smart electrical grids and smart buildings. For privacy reasons, any health care data used in scientific inquiries must be anonymized to prevent the release of personal identifying information. Similarly, prior to any automated control procedure, data records gathered from smart metering in power networks may also require anonymization. Furthermore, managing large-scale charging of electric vehicles may require each vehicle to report a number of parameters, however, these parameters do not have to be associated with other identity attributes of the vehicle itself, of the vehicle's owner.

To effectively support all of these highly distributed application domains and essentially achieve a global *Intercloud* or *Internet of Things*, a minimally complete set of federation management tools and standards will have to be developed and widely adopted. We have used the term *Big Identity* to concisely denote such a set of tools and standards. Such Big Identity mechanisms will enable the establishment and management of *trust relationships* among relevant users, IdPs and SPs for any given application domain. In addition to reviewing relevant existing standards, we have identified how possible early adoptors may be able to ameliorate the circular dependencies of standards adoption. At the end of the day, users must tell their vendors what they need, i.e., federation management tools.

Finally, we must emphasize that Big Identity must not be construed to be Big Brother. At all times, an individual must have complete control over which federations they participate in, and how their on-line identities are exposed and used. Developing such federation capabilities and establishing the human trust that they work must be concomitantly addressed.

Acknowledgments

This work was supported in part by: The Industry and Innovation department of the Aragonese Government and European Social Funds (COSMOS group, ref. T93) and the Spanish Ministry of Economy (Programa de I+D+i Estatal de Investigaci'on, Desarrollo e innovaci on Orientada a los Retos de la Sociedad TIN2013-40809-R).

References

[1] Yuri Demchenko, Canh Ngo, Paola Grosso, Cees de Laat, and Peter Membrey. Cloud based infrastructure for data intensive e-science applications: Requirements and architecture. *Cloud Computing with e-Science Applications*, page 17, 2015.

[2] A. Simonet, K. Chard, G. Fedak, and I. Foster. Using active data to provide smart data surveillance to e-science users. In *2015 23rd Euromicro International Conference on Parallel, Distributed, and Network-Based Processing*, pages 269–273, March 2015.

[3] Omar Y. Al-Jarrah, Paul D. Yoo, Sami Muhaidat, George K. Karagiannidis, and Kamal Taha. Efficient machine learning for big data: A review. *Big Data Research*, 2(3):87 – 93, 2015. Big Data, Analytics, and High-Performance Computing.

[4] Marty KohnTrevor Strome Mike Cottle, Shadaab Kanwal and Neil W. Treister. Transforming health care through big data. Technical report, Institute for Health Technology Transformation, 2012.

[5] C. Lee and D. Chadwick. The Virtual Organization Concept for Authorization Management in Federated Clouds, 2013. Published on-line: `https://www.openstack.org/assets/presentation-media/VOs-in-OpenStack-Design-Summit-v5.pptx`.

[6] C. Lee, N. Desai, and A. Brethorst. A Keystone-Based Virtual Organization Management System. In *CloudCom*, December 2014.

[7] NSF. New u.s.-japan collaborations bring big data approaches to disaster response. `https://www.nsf.gov/news/news_summ.jsp?cntn_id=134609`, 2015.

[8] GEO. The Global Earth Observation System of Systems (GEOSS): 10-Year Implementation Plan. `https://www.earthobservations.org/documents/10-Year_Implementation_Plan.pdf`, 2005.

[9] GEO. GEO Strategic Plan 2016-2025: Implementing GEOSS. `http://www.earthobservations.org/documents/GEO_Strategic_Plan_2016_2025_Implementing_GEOSS.pdf`, 2016.

[10] Nativi, S., P. Mazzetti, M. Santoro, F. Papeschi, M. Craglia and O. Ochiai. Big Data challenges in building the Global Earth Observation System of Systems. *Environmental Modelling & Software*, 68:1–26, 2015.

[11] Mattia Santoro and Stefano Nativi and Paolo Mazzetti. Contributing to the geo model web implementation: A brokering service for business processes. *Environmental Modelling & Software*, 84:18–34, October 2016. ISSN 1364-8152.

[12] Hassan Farhangi. The path of the smart grid. *IEEE power and energy magazine*, 8(1):18–28, 2010.

[13] David J. Leeds. The smart grid in 2010: Market segments, applications and industry players. Technical report, GTM Research, 2009.

[14] R. Tolosana-Calasanz, J. Diaz-Montes, O. Rana, and M. Parashar. Feedback-control queueing theory-based resource management for streaming applications. *IEEE Transactions on Parallel and Distributed Systems*, PP(99):1–1, 2016.

[15] Patrick McDaniel and Stephen McLaughlin. Security and privacy challenges in the smart grid. *IEEE Security and Privacy*, 7(3):75–77, 2009.

[16] Elias Leake Quinn. Privacy and the new energy infrastructure. *Available at SSRN 1370731*, 2009.

[17] The Smart Grid Interoperability Panel Cyber Security Working Group. Nistr 7628 guidelines for smart grid cyber security: Vol. 2, privacy and the smart grid. Technical report, NIST, 2010.

[18] I. Petri, Mengsong Zou, A.R. Zamani, J. Diaz-Montes, O. Rana, and M. Parashar. Integrating software defined networks within a cloud federation. In *IEEE/ACM CCGrid*, pages 179–188, 2015.

[19] Carl Kesselman. Big data and the internet of important things. In *High Performance Computing: from clouds and big data to exascale and beyond*, 2016. `http://www.hpcc.unical.it/hpc2016/prsnts/kesselman.ppt`.

[20] General Electric. Predix. `https://www.predix.com/`.

[21] The Interoperable Global Trust Federation. http://www.igtf.net.

[22] Craig A. Lee. Cloud Federation Management and Beyond: Requirements, Relevant Standards, and Gaps. *IEEE Cloud Computing*, 3(1):42–49, Jan-Feb 2016.

[23] Nigel Shadbolt, Wendy Hall, and Tim Berners-Lee. The Semantic Web Revisited. *IEEE Intelligent Systems*, May/June 2006.

[24] David De Roure. The semantic grid: Past, present and future. In Asunción Gómez-Pérez and Jérôme Euzenat, editors, *The Semantic Web: Research and Applications: Second European Semantic Web Conference, ESWC 2005, Heraklion, Crete, Greece, May 29–June 1, 2005. Proceedings*, pages 726–726. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[25] Marcio Roberto Miranda Assis and Luiz Fernando Bittencourt. Avoiding free riders in the cloud federation highways. In *Proceedings of the 7th International Conference on Cloud Computing and Services Science - CLOSER*, volume 1, pages 197–207. INSTICC, ScitePress, 2017.

[26] Rajkumar Buyya, David Abramson, Jonathan Giddy, and Heinz Stockinger. Economic models for resource management and scheduling in grid computing. *Concurrency and computation: practice and experience*, 14(13-15):1507–1542, 2002.

[27] Mehdi Nazari Cheraghloua, Ahmad Khadem-Zadehb, and Majid Haghparastc. A survey of fault tolerance architecture in cloud computing. *Journal of Network and Computer Applications*, 61:81–92, February 2016.

[28] P. Mell and T. Grance. The NIST Definition of Cloud Computing. `http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf`, September 2011. Special Publication 800-145.

[29] K.Benedict and M.Best and S.Fyfe and S.Habtezion and C.Jacobs and W.Michener and S.Nativi and J.Pearlman and L.Powers and A.Turner. Sustainable business models for brokering middleware to support research interoperability. RDA report, 2015. DOI: 10.13140/RG.2.2.20854.40003.

[30] S. Nativi, M. Craglia, and J. Pearlman. Earth Science Infrastructures Interoperability: The Brokering Approach. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 6(3):1118–1129, June 2013.

[31] D. Chadwick, K. Siu, C. Lee, Y. Fouillat, and D. Germonville. Adding Federated Identity Management to OpenStack. *J. of Grid Computing*, December 2013. Published on-line: `http://rd.springer.com/article/10.1007/s10723-013-9283-2`.

[32] ISO/IEC. UPnP Device Architecture. `https://www.iso.org/standard/57494.html`, 2011. ISO/IEC 29341-1-1:2011.

[33] IETF. Vendor Extensions for Service Location Protocol, Version 2. `https://tools.ietf.org/html/rfc3224`, 2002. IETF RFC 3224.

[34] OASIS. UDDI Specification, V 3.0.2. `http://www.uddi.org/pubs/uddi-v3.0.2-20041019.htm`, 2004.

[35] IBM and Microscort. Web Services Inspection Language (WS-Inspection) 1.0. `https://svn.apache.org/repos/asf/webservices/archive/wsil4j/trunk/java/docs/wsinspection.html`, 2002.

[36] OASIS. Web Services Dynamic Discovery (WS-Discovery) Version 1.1. `http://docs.oasis-open.org/ws-dd/discovery/1.1/os/wsdd-discovery-1.1-spec-os.html`, 2009.

[37] IETF. Lightweight Directory Access Protocol (LDAP): The Protocol. `https://tools.ietf.org/html/rfc4511`, 2006. IETF RFC 4511.

[38] Peter Waher an Ronny Klauck. XEP-0347: Internet of Things - Discovery. `https://xmpp.org/extensions/xep-0347.html`, 2016.

[39] OASIS. Extensible Resource Descriptor (XRD) Version 1.0. `http://docs.oasis-open.org/xri/xrd/v1.0/cd01/xrd-1.0-cd01.pdf`, 2009.

[40] Resource Description Framework. `http://www.w3.org/RDF`.

[41] OWL 2 Web Ontology Language Document Overview (Second Edition). `http://www.w3.org/TR/2012/REC-owl2-overview-20121211/`, 11 December 2012.

[42] SPARQL Query Language for RDF. `http://www.w3.org/TR/rdf-sparql-query`, 15 January 2008.

[43] M. Rutkowski, (ed.). Cloud Auditing Data Federation - Data Format and Interface Definitions Specification (DSP0262). `http://www.dmtf.org/sites/default/files/standards/documents/DSP0262_1.0.0.pdf`, June 19 2014.

[44] Production Quality, Multilayer Open Virtual Switch. `http://openvswitch.org`.

[45] Openflow. `https://www.opennetworking.org/ja/sdn-resources-ja/onf-specifications/openflow`.

[46] SDN Architecture Overview, Version 1.0. `http://www.opennetworking.org`, 12 December 2013.

[47] Mark Berman et al. Geni: A federated testbed for innovative network experiments. *Computer Networks*, 61(0):5 – 23, 2014.

[48] C. Kaler and A. Nadalin (ed.). Web Services Federation Language (WS-Federation) Ver-

sion 1.2. http://docs.oasis-open.org/wsfed/federation/v1.2/os/ws-federation-1.2-spec-os.html, 22 May 2009.

[49] A. Nadalin and others (ed.). WS-Trust 1.3. http://docs.oasis-open.org/ws-sx/ws-trust/200512, March 2007.

[50] A. Jøsang and R. Ismail and C. Boyd. A Survey of Trust and Reputation Systems for Online Service Provision. *Decision Support Systems*, 43:618–644, March 2007.

[51] V. Shmatikov and C. Talcott. Reputation-Based Trust Management. *J. Computer Security*, 2005.

[52] K. Chard and K. Bubendorfer and S. Caton and O. Rana. Social Cloud Computing: A Vision for Socially Motivated Resource Sharing. *IEEE Trans. on Services Computing*, 5(4):551–563, October-December 2012.

[53] OpenID Authentication 2.0 Final. http://openid.net/specs/openid-authentication-2_0.html, Dec 5, 2007.

[54] N. Sakimura et al. OpenID Connect Standard 1.0 - draft 13. http://openid.net/specs/openid-connect-standard-1_0.html, Aug. 16, 2012.

[55] R. Housley and others. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List Profile. http://www.ietf.org/rfc/rfc3280.txt, April 2002.

[56] S. Tuecke et al. Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile, IETF RFC 3820. http://www.ietf.org/rfc/rfc3820.txt, June 2004.

[57] OASIS. Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0. http://docs.oasis-open.org/security/saml/v2.0, March 2005.

[58] E. Rissanen, (ed.). eXtensible Access Control Markup Language (XACML) Version 3.0. http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.pdf, January 22 2013.

[59] IETF. The OAuth 2.0 Authorization Framework. https://tools.ietf.org/html/rfc6749, 2012. IETF RFC 6749.