CLOSER 2015

5th INTERNATIONAL CONFERENCE ON CLOUD COMPUTING AND SERVICES SCIENCE

Doctoral Consortium

Lisbon, Portugal

20 - 22 May, 2015

www.closer.scitevents.org

SPONSORED BY:

INSTICC

LOGISTICS PARTNER:

SCITEVENTS
SCIENCE AND TECHNOLOGY EVENTS

PAPERS AVAILABLE AT:

SCITEPRESS
DIGITAL LIBRARY

CLOSER 2015

Doctoral Consortium

Lisbon, Portugal

20 - 22 May, 2015

Doctoral Consortium Chair

Paulo Novais, Universidade do Minho, Portugal

Advisory Board

Bernd Amann, LIP6 - Pierre and Marie Curie University, France
Cornel Klein, Siemens AG, Germany
Omer Rana, Cardiff University, U.K.
Paolo Traverso, Center for Information Technology - IRST (FBK-ICT), Italy

http://closer.scitevents.org

closer.secretariat@insticc.org

CONTENTS

PAPERS

PAPERS

Insights for Manage Geospatial Big Data in Ecosystem Monitoring using Processing Chains and High Performance Computing

Fabián Santos[1] and Gunther Menz[2]
[1]*Center for Remote Sensing of Land Surfaces, University of Bonn, Walter Flex Straße 3, Bonn, Germany*
[2]*Remote Sensing Research Group, Department of Geography, University of Bonn,*
Meckenheimer Allee 166, 53115 Bonn, Germany
{s7fasant, g.menz}@uni-bonn.de

1 RESEARCH PROBLEM

Big data (BD) is nowadays a research frontier and a strategic technology trend, which is still emerging as a new scientific paradigm in many fields (Chen and Zang, 2014). It is commonly conceptualized by the 3V´s model of (Laney, 2009), who defines three dimensions in BD known as: volume (data size), variety (data types) and velocity (production rate); that could be challenging to analyze, especially in large quantities. For these reasons, processing and analysis of BD requires new approaches, which are not suitable for conventional software and hardware.

According to (Percival, 2009) Geospatial data (GD) has always been BD but not as it is today, due the accelerated increase and accessibility of geographical technologies (as for example: state-of-art earth observation satellites, mobile devices, ocean-exploring robots, unmanned aerial vehicles, etc.). Moreover, the distribution policies in favor of free and open access to archives are giving way to automated mass processing of large collections of Geospatial data (Hansen and Loveland, 2012). Thereby, this type of data can be considered (under certain conditions), as a synonym of BD which requires not only powerful processors, software, algorithms and skilled data researchers (European Commission, 2014) but also a set of conditions that are not fully met to make possible and accessible the data-intensive scientific discovery.

For these reasons, this research aims to explore the link between Geospatial data and BD, especially in the design and programming of processing chains, which usually do not show explicit considerations to manage the BD dimensions, moreover applied to ecosystem monitoring research.

2 OUTLINE OF OBJECTIVES

Due that the BD and Geospatial data involves many combinations; only two cases will be analyzed on this research. The first case involves the analysis of a large set of satellite images, so the volume dimension of BD will be the main challenge. The second case implies a unique large database of different Geospatial data sources and types, thus the variety dimension of BD will be the main problem. For these reasons, this research is divided in two empirical objectives and one theoretical, whose purposes will be the following:

- Analyze the restoration process of disturbed tropical forests in Ecuador. For this reason, a processing chain for prepare a large collection of Landsat images and a time series analysis will be developed, applying the high performance computing approach.
- Identify the environmental drivers that influence the restoration process of disturbed tropical forests in Ecuador. To this purpose, an exploratory statistical analysis and pattern extraction from a database composed by different sources and types of Geospatial data will be review. This will require the development of another processing chain using the high performance computing approach for harmonize and extract the patterns inside the data.
- Describe the linkages of BD in Geospatial data and the benefits of the high performance computing approach in processing chains. Due this reason, the processing chains already developed will be debugged and optimized in order to guarantee their reproducibility and distribution as open source software. Moreover, a detailed description of their design and processing efficiency will allow the conclusion of the technical aspects of this objective.

3 STATE OF THE ART

In the early seventies, the term “information

overload" was mention by (Toffler, 1970) to explain the difficulties associated with decision making due the presence of excessive information. After that, a concern for the management and interpretation of large volumes of data became more relevant; however, without a proper solution.

When computer systems were developed enough for recognize or predict patterns on data (Denning, 1990), the scientific community was able to describe with more details the properties of the BD. The first known scientists who conceptualized the term were (Cox and Ellsworth, 1997) and they described it as the large data sets which exceed the capacities of main memory, local disk, and even remote disk. Consequently, the term was mainly associated with the size or volume of the data but (Laney, 2001) proposed two additional properties for describe BD calling them: variety, for refer to the diversity of data types and; velocity, for indicate the production rate of data. This concept approach is called the three V´s of BD and nowadays inspires most of the BD management strategies. However, other authors as (Assunção et al., 2013) suggest additional V´s properties and considerations for BD management calling them: veracity, value, visualization and vulnerability.

Regarding to Geospatial data as BD, its use constitutes a research frontier which is making conventional processing and spatial data analysis methods no longer viable. The increasingly data collection and complexity of sensors aboard the earth observation satellites and other technology devices based in Geospatial data production is nowadays demanding new platforms for processing, which are now accessible through cloud computing services (Sultan, 2010). However, the scientific literature related to this field is not so numerous than the research done over individual or small collections of satellite images using conventional computing methods. A decline on this tendency over time is concluded by (Hansen et al., 2012), who affirms that methods in the future will evolve and adapt to greater data volumes and processing capabilities; and (Gray, 2009) who anticipated a revolution of scientific exploration based on data-intensive and high-performance computing resources.

The release by NASA and the USGS of a new Landsat Data Distribution Policy (National Geospatial Advisory Committee, 2012) which enables the free download of the whole available data collection constitutes an example of a data-intensive source which demands new approaches for extract meaningful information. In this sense, (Potapov et al., 2012) demonstrated the feasibility to work with large Landsat collections developing a methodology which enable the quantification of forest cover loss through the analysis of a set of 8,881 images and a decision tree change detection model. Moreover, (Flood et al., 2013) proposed an operational scheme for process a standardized surface reflectance product for 45,000 Landsat TM/ETM+ and 2,500 SPOT HRG scenes, developing an innovative procedure for correct the atmosphere, bidirectional reflectance and topographic variability between scenes. However, in both cases, is unknown the computing strategies adopted for manage and process such large collection of images.

Nonetheless, other authors describe with detail the use of High Performance Computing and Geospatial data. For instance, (Wang et al., 2011) develop a prototype of a scientific Cloud computing project applied in remote sensing, which describes the requirements and organization of the resources needed; (Almeer, 2012; Beyene, 2011) investigated the MapReduce programming paradigm for process large collection of images; and (Christophe et al., 2010) describes some benefits of Graphical processing units (GPU) respect to Multicore Central Processing Units (CPU) on the processing time of different algorithms types, commonly used in remote sensing.

From all the references consulted, these two approaches were mainly found, in other words, separating the design of the remote sensing processing chains from the BD management strategies. For this reason, this research aims to couple them on two specific cases of large Geospatial data collections applied on Ecosystem monitoring, which involve the design of processing chains and BD management strategies.

4 METHODOLOGY

As is mention in the section 2, the empirical research will be applied in two cases, therefore each research objective has their own specific data sources, analysis methods, validation procedures and study areas (except for the third one which is mainly theoretical). The materials and methods are summarized in the next paragraphs (subsections 4.1 to 4.4):

- Data sources: multispectral and radar remote sensing products, aerial photography, ancillary cartography, climatic databases, GPS inventories, field recognition and surveys.

- Data analysis: literature research, parallel computing paradigm, image processing, machine learning, time series analysis and exploratory statistics
- Validation procedures: multi-scale accuracy assessment
- Study areas: selected sites of different tropical forest in Ecuador

The study areas considered for this research represent sites which the available Geospatial data achieves the minimum information requirements, furthermore with a good register of field data and fidelity needed for the validation procedures. Finally, on this review only the first objective materials and methods are described in detail due the advances achieved until now.

4.1 Geospatial Data

For categorize the Geospatial data of the first objective, two types are listed with their respective sources:

- Primary sources (all in raster format):
  1) A set of Landsat 4, 5, 7 and 8 images (±350 data sets collected) acquired for the multispectral sensors TM, ETM+ and OLI-TIRS (all with 7 spectral bands in the optical, infrared and thermal regions) over 3 scenes (each scene covers 33,300 km2); processed until the level L1T (which means a geometric correction but not a radiometric correction). This information covers a period of ± 30 years with time intervals of 0.5 to 2 years; and with a spatial resolution of 30 meters.
  2) A set of digital elevation models obtained from the Shuttle Radar Topography Mission for the respective Landsat scenes, with a correction of the data voids (however with a fair quality). The spatial resolution of this data is 30 meters
  3) A set of very high resolution images from the RapidEye satellites (5 meters of spatial resolution with a geometric correction) and from historical archives of aerial photography (this information is under request).
- Secondary sources (raster and vector format):
  4) Ancillary cartography, which describes the ecosystems and the different forest types in Ecuador; and other layers for describe the human and biophysical features of landscapes (roads, rivers, administrative boundaries, cities, soil types, climates, etc.)
  5) Climate databases from the WorldClim and metereological stations.
  6) GPS inventories of plant species and forest carbon stock measures on the field.

4.1.1 Open Issues

Due that the processing capabilities are restricted to a multicore computer; the raster data used for the analysis is reduced to a set of small study areas inside the image scenes. However, our interest is extend the processing capabilities using a cloud computing service for modify and upgrade the processing chain developed with the complete area of the images.

4.2 High Performance Computing

According to (Christophe et al., 2011) high performance computing is a natural solution to provide the computational power, which have several approaches like cluster, grid or cloud computing. Moreover, this approach not only refers to a connected groups of computers locally or geographically distributed; it refers as well to the parallel computing paradigm which is the simultaneous assign of tasks when is possible to divide a big processing problem into smaller ones. This can be done through the central processing units (CPU) or the graphical processing units (GPU) which nowadays computers have as hardware resources.

The design of the processing chain for the Landsat images applies the parallel processing paradigm through the use of the cores in a multicore computer and the division of the collection of images acquired. For this purpose, an automatic detection of the cores in the computer makes possible to obtain the factor needed for subdivide the complete list of images. This is done in the R language through the package “foreach” (Weston, 2015) which allows the management of the cores in a loop programming structure. The next R script shows this design:

```
   #list the data repository in the
   variable "load.data" which is the
   set of images
   load.data <- list.files(load.data,
full.names= T)

   #detect the number of cores
available in the computer
   ncores <- detectCores(all.tests =
TRUE, logical = TRUE)
```

```
   #subdivide the "load.data" list in
groups according to the cores
   load.data <-
   split(load.data,as.numeric(gl(length
   (load.data),ncores,length(load.data)
   )))

   #for each element "j" in the
   "load.data" subdivided list, take
   only a set (which #represent a set
   of 4 image directories when a
   computer have 4 processing cores)
   for (j in 1:length(load.data)){
    data.set <- load.data[[j]]


    #register cores for apply in
   parallel
       clusters <-makeCluster(ncores)
       registerDoParallel(clusters)

       #apply the parallel loop for
       each element "i" inside the
       variable #"data.set". Due that
       each core need to load an
       environment, the command
       # "package=raster" specify that
       is needed this package to
       execute the script

   foreach(i=1:length(data.set),.packag
   es="raster") %dopar% {

       #here comes the script which
       indicates the different
       operations that should #be done
       to each image folder, which is
       indexed by the "i" element
       inside #the variable "data.set"
       …
       #for close the parallel loop for
       the "i" element
       }

       #for stop the cores and prepare
       them for the next element of the
       loop
       stopCluster(cl)

   #finally, for close the loop of the
   subdivided list "load.data" another
   curly #brackets is needed
   }
```

This structure allowed the distribution and application of complex algorithms over set of images, instead individual images, decreasing the processing time and according to (Zecena et al., 2012) a more efficient energy consumption and algorithm processing.

4.2.1 Open Issues

This approach leads to a further experimentation in a cluster, grid or cloud computing environment as is mentioned before, however is needed a feedback for validate the data distribution between the cores when they are parallelized, as well the management of cores in a cloud computing service.

4.3 Image Processing

The image processing approach of this research follows five modules, each one composed of different sets of processing tasks which accomplish specific objectives. This is showed in the figure 1, which is a flowchart that summarizes the steps of the processing chain developed, however not yet finished.

The design of this approach is inspired by the work of (Flood et al., 2013; Hansen et al., 2007; Potapov et al., 2012) who developed processing chains for large collections of Landsat images and; agreed in almost all cases with the order of the next required steps: 1) georectification/resampling; 2) conversion to the top of atmosphere (TOA) reflectance/atmospheric correction; 3) cloud/shadow/water masking; 4) standardization of the reflectance; 5) topographic and bi-directional reflectance normalization; 6) radiometric validation; 7) index generation/image classification/change detection/time series analysis; and 8) accuracy assessment.

4.3.1 Open Issues

Some steps of the processing chain are missing due: a fail in the processing algorithms used or an absence of the algorithm in the R language packages repository. This involves new challenges and in some cases a change in the language used (as for example the coregister script which had to be programmed in Python). Due that our aim is produce and distributes a pure R application for process large collections of Landsat images, this can interrupt the sequence of steps involved. Therefore, new algorithms, libraries, software and package alternatives are being searched, but with the only requirement that they should be open source.

Respect to the accuracy assessment, the results of the classifications of the images, will allow the measurement of the precision through the use of very high resolution satellite images and aerial photography. This approach called multi-scale accuracy has been proven for validate MODIS

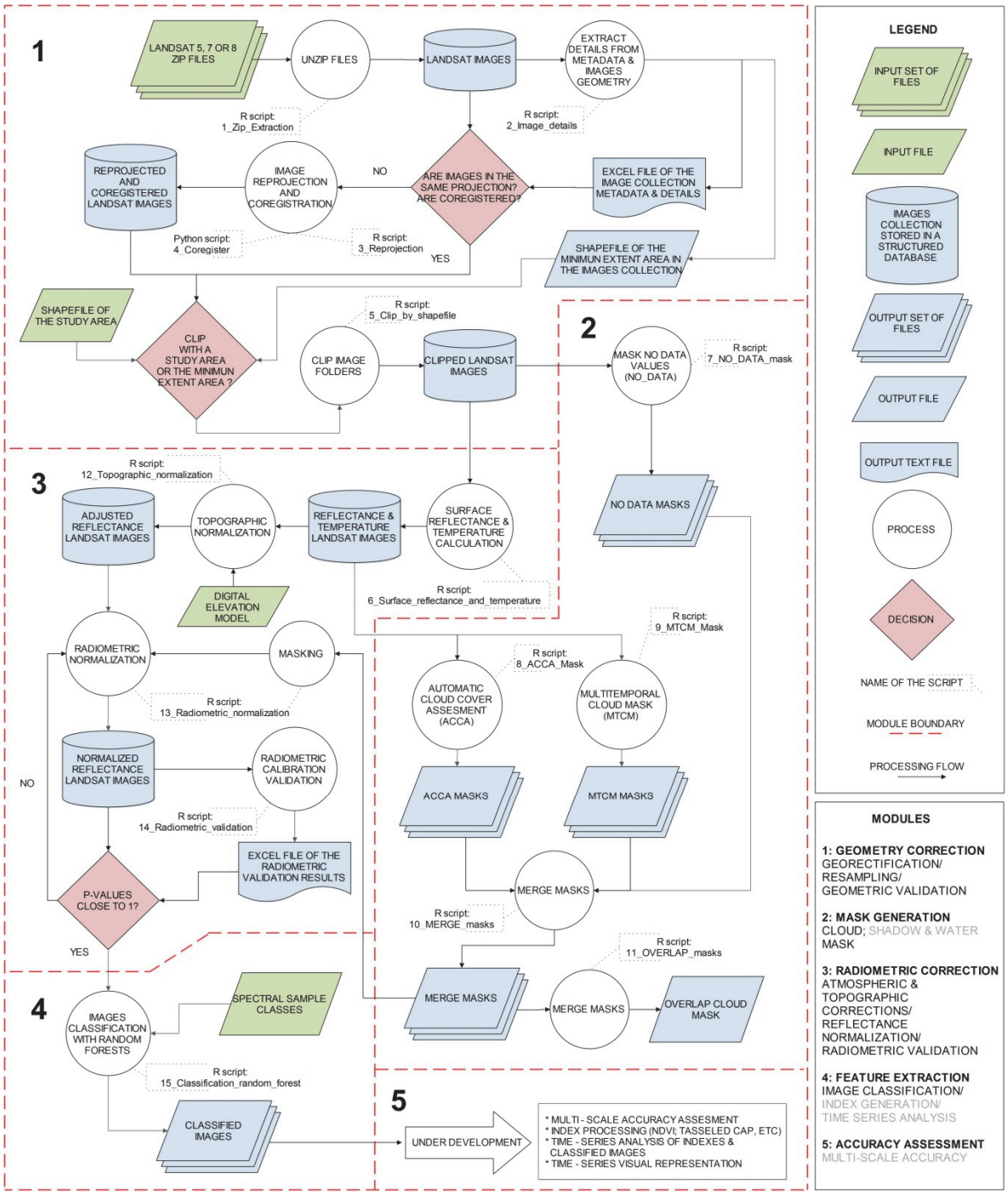

Figure 1: Flowchart of the processing chain under development for large collection of Landsat images.

(Morisette et al., 2012) and Landsat products (Goward et al., 2003); however with long time series a further literature research is needed for adopt a robust method.

4.4 Tropical Forests in Ecuador

Ecuador, despite its small size (only 283,560 km2), is one of the most diverse countries in the world (Sierra et al., 2002) and is counted 91 terrestrial ecosystems in its continental extension and 9 forest

types (MAE, 2013). However, with a deforestation rate of 0.68 – 1.7 % annual, which equals to 61,764.50 hectares per year (MAE, 2012) put it on the list of the countries with highest deforestation rate in the world (Tryse, 2008). Due of this, is needed better ecosystem monitoring methodologies, which can manage the lack of high quality remote sensing data and corresponding ground data sets (Avitabile et al. 2012).

4.4.1 Open Issues

Respect to the study areas of tropical forest in Ecuador, three study areas along an altitude gradient in the Amazon region are being considered for the first objective; principally for their accessibility and data availability. Moreover, due that they are integrated in a watershed; their results can be useful for the second objective.

5 EXPECTED OUTCOME

At the end of the first part of this research, our expectations are: 1) generate a first version of a open source processing chain designed to prepare time series analysis with large collections of Landsat images; 2) evaluate the regeneration time of different forest types in Ecuador; 3) contribute with some ideas about the links between Geospatial data and BD; and 4) demonstrate some benefits of the high performance computing approach in remote sensing and processing chains.

6 STAGE OF THE RESEARCH

This research started one year ago and its proposal was accepted six months ago. Since then, the processing chain has been under active development and in six additional months will be totally finished. In the other hand, the redaction of the scientific paper about this chapter started time before and will be ready, as well too, in six months. After that period, a field work of six months in Ecuador will be done for collect the necessary information of the second research objective and corroborate the results of the first research objective.

REFERENCES

Almeer, M. 2012. Cloud Hadoop Map Reduce For Remote Sensing Image Analysis. *Journal of Emerging Trends in Computing and Information Sciences* 3 (4).

Assunção, M., R. Calheiros, S. Bianchi, M. Netto, and R. Buyya. 2014. Big Data Computing and Clouds: Trends and Future Directions. *Journal of Parallel and Distributed Computing*.

Avitabile, V., A. Baccini, M. Friedl, and C. Schmullius. 2012. Capabilities and limitations of Landsat and land cover data for aboveground woody biomass estimation of Uganda. *Remote Sensing of Environment* 117:366-380.

Beyene, E. 2011. Distributed Processing Of Large Remote Sensing Images Using MapReduce A case of Edge Detection, Institute for Geoinformatics, Universität Münster, Münster - North-Rhine Westphalia - Germany.

Buckner, J., and M. Seligman. 2015. Package 'gputools': R-Project.

Chen, P., and C.-Y. Zhang. 2014. Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Information Sciences* 275 (2014) 314–347.

Christophe, E., J. Michel, and J. Inglada. 2010. Remote Sensing Processing: From Multicore to GPU. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 1.

Cox, M., and D. Ellsworth. 1997. Application-Controlled Demand Paging for Out-of-Core Visualization. Paper read at The 8th IEEE Visualization '97 Conference.

Denning, P. 1990. Saving All the Bits: Research Institute for Advanced Computer Science, 15.

European Commission. 2014. Communication from the Commission to the European Parliament, The Council, The European Economic and Social Committee and The Committee of the Regions. Brussels 2.7.2014.

Flood, N., T. Danaher, T. Gill, and S. Gillingham. 2013. An Operational Scheme for Deriving Standardised Surface Reflectance from Landsat TM/ETM+ and SPOT HRG Imagery for Eastern Australia. *Remote Sensing* 5:83-109.

Goward, S., P. Davis, D. Fleming, L. Miller, and J. Townshend. 2003. Empirical comparison of Landsat 7 and IKONOS multispectral measurements for selected Earth Observation System (EOS) validation sites. *Remote Sensing of Environment* 88 (2003) (80 – 99).

Gray, J. 2009. Jim Gray on eScience: A Transformed Scientific Method. In *The Fourth Paradigm Data Intensive Scientific Discovery*. Washington - EEUU: Microsoft Research, xxii.

Hansen, M., and T. Loveland. 2012. A review of large area monitoring of land cover change using Landsat data. *Remote Sensing of Environment* 122 (2012) 66–74.

Hansen, M., D. Roy, E. Lindquist, B. Adusei, C. Justice, and A. Altstatt. 2007. A method for integrating MODIS and Landsat data for systematic monitoring of forest cover and change in the Congo Basin. *Remote Sensing of Environment (2008)* 112 2495–2513.

Laney, D. 2001. 3D Data Management: ControLling Data Volume, Velocity, and Variety. *Application Delivery Strategies META Group Inc.*

MAE. 2012. Línea Base de Deforestación del Ecuador

Continental edited by Subsecretaría de Patrimonio Natural. Quito - Ecuador: Ministerio del Ambiente (MAE).

MAE. 2013. Metodología para la representación Cartográfica de los Ecosistemas del Ecuador Continental, edited by Subsecretaría de Patrimonio Natural. Quito - Ecuador: Ministerio del Ambiente del Ecuador (MAE).

Morisette, J., J. Privette, and C. Justice. 2002. A framework for the validation of MODIS Land products. *Remote Sensing of Environment* (2002) 83:77 – 96.

National Geospatial Advisory Committee. 2012. Statement on Landsat Data Use and Charges.

Percivall, G. 2013. Big Processing of Geospatial Data: Open Geospatial Consortium.

Potapov, P., S. Turubanova, M. Hansen, B. Adusei, M. Broich, A. Altstatt, L. Mane, and C. O. Justice. 2012. Quantifying forest cover loss in Democratic Republic of the Congo, 2000–2010, with Landsat ETM+ data. *Remote Sensing of Environment* 122 (2012) (106–116).

Sierra, R., F. Campos, and J. Chamberlin. 2002. Assesing biodiversity conservation priorities: ecoszstem risk and representativeness in continental Ecuador. *Landscape and Urban Planning* 59 (2002):95-110.

Sultan, N. 2009. Cloud computing for education: A new dawn? *International Journal of Information Management* 30 (2010) 109–116.

Toffler, A. 1970. *Future Shock*. United States: Random House.

Tryse, D. 2008. David' s Google Earth files:Disappearing Forests of the World: Google Earth.

Wang, L., M. Kunze, J. Tao, and G. v. Laszewski. 2011. Towards building a cloud for scientific applications. *Advances in Engineering Software* 42 (2011):714–722.

Weston, S. 2015. Package 'foreach': Revolution Analytics R-Project.

Wyborn, L. 2013. It's not just about big data for the Earth and Environmental Sciences: it's now about High Performance Data (HPD) In *Big Data*: Geoscience Australia.

Zecena, I., Z. Zong, R. Ge, T. Jin, Z. Chen, and M. Qiu. 2012. Energy Consumption Analysis of Parallel Sorting Algorithms Running on Multicore Systems. Paper read at Green Computing Conference (IGCC), at San Jose, CA.

Sporadic Cloud Computing over a Virtualization Layer
A New Paradigm to Support Mobile Multi-hop Ad-hoc Networks

Esteban F. Ordóñez-Morales[1], Yolanda Blanco-Fernández[2] and Martín López-Nores[2]
[1]*Grupo de Investigación e Innovación en Ingenierías, Universidad Politécnica Salesiana, Cuenca, Ecuador*
[2]*AtlantTIC Research Center, Department of Telematics Engineering, University of Vigo, Vigo, Spain*
eordonez@ups.edu.ec, {yolanda, mlnores}@det.uvigo.es

In our doctoral proposal we deploy Sporadic Ad-hoc Networks (SANs) over the devices of a group of always-on users who happen to meet in a place. The goal is to develop tailor-made services that exploit the possible similarities among the preferences of the users and the technological capabilities of their terminals to establish direct and hop-by-hop ad-hoc communications. In order to overcome the intrinsic limitations of mobile devices, we explore the new concept of Sporadic Cloud Computing (SCC) that is aimed at providing each terminal with additional resources by exploiting the (computational, networking, storing...) capabilities of the rest of devices connected to the SAN. In order to abstract the complexity stemmed from the mobility scenarios, SCC works with an enhanced Virtualization Layer that deals with a few static virtual nodes instead of a higher number of mobile real nodes. This allows to turn our SANs into reliable and stable communication environments to promote interactions among potentially like-minded strangers in a great diversity of mobility scenarios, involving both pedestrians and cars in vehicular environments.

1 RESEARCH PROBLEM

After the irruption of the Web 2.0 and the smartphone revolution, most of the on-move users carry with them handheld devices for actively interacting daily with their friends/followers/followees/... in the context of the virtual world of the Internet. Sociologist have already advised about the negative effects derived from some of these behaviours, which might lead the users to immersing themselves in a virtual communication burble with their contacts, by giving up interacting face-to-face with nearby individuals (Kuss and Griffiths, 2011). In the same line, other experts have analyzed the consequences of the so-called FOMO (Fear Of Missing Out) effect which denotes the fear of users of losing events, news and situations that happen in their social networks, thus not taking an eye off their devices (Przybylski et al., 2013). In order to fight these situations, we are working in the development of the SPORANGIUM (SPORAdic networks in the Next-Generation Information services for Users on the Move) platform which deployes Sporadic Ad-hoc Networks (SANs) among always-on users who happen to be in a place, by establishing multi-hop ad-hoc connections over their respective mobile terminals. The goal is to promote more direct interactions among strangers who happen to meet in spaces like cinemas, stadiums, museums, concert halls, etc., who might have potentially common interests (cinema, sport, art, music, etc.) that would be convenient to explore.

For that purpose, SPORANGIUM must orchestrate activities and tailor-made services that bring together the particular context of the users, their potentially common preferences and the capabilities of their devices for establishing ad-hoc connections. However, these services far exceed traditional mobile devices capabilities, which suffer from computational limitations, as well as battery restrictions and processing time. To face this situation, the new Mobile Cloud Computing (MCC) paradigm has arisen that takes inspiration from the well-known Cloud Computing (CC), which is based on delivering computing as a service whereby shared resources, software and information are provided as a utility over a network (typically the Internet). In MCC the goal is to enable to process a large amount of data on demand anytime from anywhere, so that mobile devices connect to the Internet to use an environment that integrates diverse platforms and technologies (Dinh et al., 2013). Specifically, MCC promotes to move the computing power and data storage away from mobile devices and into the cloud, bringing multiple service models (IaaS, PaaS, SaaS...) and mobile computing to a wide range of on-move always-on users.

Recently, the MCC paradigm has been exported to vehicular communication environments where some researchers have proposed the so-called Vehicular Cloud Computing (VCC). The goal here is to exploit

both the physical data center units that are in charge of performing the data computation and storage (like in MCC) and the on-board resources of the own vehicles (Olariu et al., 2013). In our proposal of PhD work, we want to extend the VCC paradigm to support mobility scenarios beyond the vehicular environment, by involving both pedestrians and vehicles on the road. Specifically, we explore a new paradigm – named *Sporadic Cloud Computing* (SCC)– aimed to allow the users' devices to exploit both the (computing, storing, networking, sensing...) resources available in the rest of terminals connected to the SANs, and those provided from external data centers. In the deployment of our SCC paradigm in diverse mobile communication environments, one of the main research challenges has to do with the high mobility of the nodes connected to the ad-hoc network (e.g. cars in a vehicular ad-hoc network), and therefore, with its frequenlty changing topology. This causes that communication fails often as these nodes move fast and are out of the range of the ad-hoc network, which also hampers the routing tasks when forwarding information (Gerla, 2012).

2 OUTLINE OF OBJECTIVES

Our doctoral proposal is aimed at designing, developing and validating the mechanisms necessary to:

1. turn our SANs networks into reliable and stable communication environments with good performance in terms of overhead, packet delivery ratios and scalability, covering vehicular, pedestrian and mixed environments, and
2. deploy enhanced "X"aaS service models (e.g. CaaS, NaaS, STaaS, SEaaS...) through our SCC paradigm, so that the devices connected to the SAN can collaborate and share their respective Computing, Networking, SToring and SEnsing capabilities in the deployment of advanced communication services.

As introduced before, the main research challenges derived from both objectives has to do with (i) the frequent topology changes happened in certain communication environments (e.g. in a vehicular ad-hoc network due to the fast movements of the cars), and (ii) the fact that the capabilities available in the SANs vary on the time, as the location of the users' devices change.

To deal with the high mobility of the nodes connected to our SANs, our proposal takes advantage of the improvements proposed in the realm of Mobile Ad-hoc Networks (MANETs), which have been envisaged to face the problems derived from (i) the wireless transmission mediums, (ii) the high variability of the network topology due to unpredictable nodes' movements, and (iii) the existence of severe restrictions in terms of processing capabilities, memory and battery consumption. In particular, we start from the work presented in (Dolev et al., 2004) where the authors described a *virtualization layer* named VNLayer (*Virtual Node Layer*). Specifically, the VNLayer is a cluster-based approach where the mobile nodes collaboratively create an infrastructure of static *virtual nodes* to ease the routing problem and the maintenance of persistent state information in the area covered by an ad-hoc wireless network of mobile devices (as our SANs), notwithstanding the mobility of the (real) physical nodes. Actually, the VNLayer resides between the link layer and the Internet Layer, so that the virtual nodes can be addressed as if they were static server devices. This helps to mask the uncertainty that arises from the MANETs' varying topology and from the fact that the physical devices can fail unpredictably. Consequently, it is easier for developers to work at the nodes' upper layers, since they can deploy applications on mobile devices and virtual servers with greater ease and efficiency. Besides, virtualisation creates a level of hierarchy in the otherwise flat MANETs, which brings in opportunities to re-design MANET protocols to operate more efficiently and reliably.

Since the virtualization layer by Dolev et al. has been developed to handle communications in MANETs, the first objective of our doctoral proposal consists of **extending and adapting the working of the VNLayer to the restrictions and peculiarites of more demanding mobility scenarios**, including, for instance, communication environments where pedestrians and occupants of vehicles are involved. To this aim, we need to envisage refinements in the VNLayer (resulting in our VNLayer+) to take into account, for instance, the comparatively faster movements of vehicles, the freedom of movement of the pedestrians, as well as the fact that these nodes are not subject to the strict energy, space and computing capabilities restrictions of MANETs. These restrictions must be considered to turn our SANs into reliable communication environments, covering a wide diversity of application scenarios beyond the generic MANETs explored in Dolev et al.'s approach.

Our SCC paradigm must develop **transport-layer coordination mechanisms among the devices that are connected to the SANs in order to enable an efficient sharing and allocation of their available resources by working over the virtualization layer**,

which is the second objetive of the doctoral proposal. This fact causes that, differently from the traditional approaches envisaged in CC, MCC and VCC, our "X"aaS service models need to deal with the (static) virtual nodes of the VNLayer+, which are emulated/supported by the devices of the users on the move. Specifically, while the SAN is established among the users' terminals, the messages to request resources from the ad-hoc network (or to advertise resources that are left to other devices' disposal) are managed by virtual nodes. The tandem SCC-VNLayer+ contributes to (i) fight/alleviate the communication errors and data loss noticeable in mobile ad-hoc networks (Dinh et al., 2013), and (ii) to orchestrate advanced applications to improve the experience of the users, by taking advantage of the reliable data exchange over the SAN (thanks to the VNLayer+) and the availability of additional resources in each terminal (thanks to the service models of SCC).

The possible applications to be deployed in the realm of our SANs cover a wide spectrum, ranging from the orchestation of activities bounded to an event where a group of like-minded users happen to meet (e.g. in a museum, theater or stadium), to the provision of both improved applications for intervehicular communication (e.g. optimization of traffic flows, chats among drivers, proactive organization of ride-sharing opportunities or selective distribution of personalized advertising in nearby places), and refinements in the context of the *smart cities* through the planification of people mobility and urban games, among others.

3 STATE OF THE ART

In this section, we review related works in the two main research fields of our doctoral proposal: the use of virtualization in MANETs and the exploitation of resource sharing among devices in mobile communication environments.

3.1 Virtualization in Mobile Ad-hoc Networks

The VNLayer was presented in (Dolev et al., 2004) as a set of procedures to turn ad-hoc networks of mobile devices into more predictable environments for communications. The main idea is to engage the mobile physical nodes (PNs) in collaboration to emulate virtual nodes (VNs) that remain in known grid locations, as shown in Figure 1 (where back circles and white squares denote PNs and VNs, respectively).

The VNLayer divides the geographical area of an ad-hoc network into square regions, whose size is chosen so that every PN in a region can reliably send and receive data from every other physical node in that region and neighboring ones. Each VN (one per region) is emulated by the PNs located in the corresponding region, so that when all the physical nodes leave this region, the virtual node stops to work. In each region, one PN is chosen as the *leader* in the region and becomes the primary responsible for packet reception, buffering and forwarding. Meanwhile, a subset of non-leader nodes are designated as *backups* to maintain information consistent with the leader's version (specifically, replicas of the virtualization-related state information and the routing tables tackled by the routing protocols working on the virtualization layer). This way, the VNs can maintain persistent state and be fault tolerant even when individual PNs fail or leave the region.

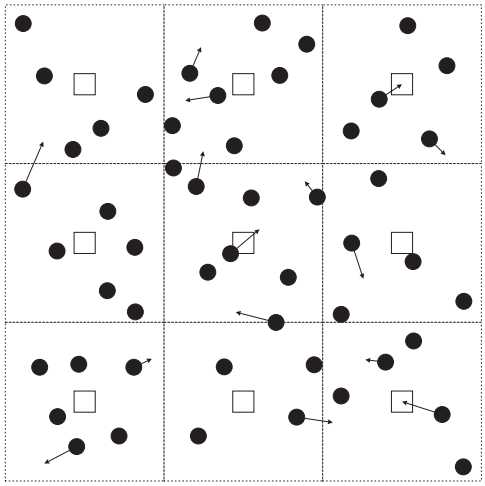

Figure 1: Static virtual nodes (white squares) overlaying the mobile nodes of a MANET (black circles).

An exhaustive analysis of the VNLayer allows to detect certain sources of inefficiency in its functioning, which are mainly related to:

- *The Procedure used to Identify New Backup Nodes*: The approach adopted by Dolev et al. in order to designate backup nodes among non-leaders is a probabilistic one and it is driven by a Coin Tosser Function, according to which the greater the number of nodes in the region, the lower the probability that a PN will choose as backup. This avoids having many backups in dense MANETs, and thus reduces the overhead due to state synchronisations, i.e. to the exchange of messages aimed at ensuring that the replicas of the state information from the upper layers (e.g. routing tables) kept in the backup nodes are con-

sistent with the leader's version.

- *The Procedure adopted in the Leader Election*: This process suffers from two main problems: (i) it involves a great number of messages to be exchanged –which contributes to increase the duration of the virtual nodes' downtimes–, and (ii) the selection process does not prioritize the backup nodes that have an updated version of the outgoing leader node (which would be the best candidates to become a new leader). In particular, this procedure could designate as new leader either a node that was not acting as a backup or a non-synchronised backup node, even in cases that there were synchronised backups in the region. This causes that the state information from the upper layers would be lost unnecessarily and new synchronisations would be triggered immediately, thus increasing the overhead.
- *The Procedures adopted Once a Region Becomes Empty*: The VNLayer does not preserve information about the states of the VNs corresponding to regions that become empty after the withdrawal of all the PNs located in them. This degrades the performance of the virtualization layer, thus causing unnecessary delays in the recovery of the VNs.

Besides the above limitations (which have been identified in MANET scenarios), the VNLayer requires additional refinements to improve its performance in more restrictive and demanding mobile communication environments, such as the pedestrian/vehicular/ mixed scenarios we want to explore in our doctoral proposal.

3.2 Resource Sharing among Mobile Devices

The idea of taking advantage of the resources available in the handheld devices that are located around an on-move user is not new at all. Specifically, the capabilities that have gained more momentum are those related to the networking resources. In this regard, the so-called *spontaneous networking* has arisen in the last years, where wireless mobile nodes opportunistically exploit multi-hop ad-hoc paths toward peers to share content and available resources in an impromptu way. The idea is to take benefit from the available bandwidth in many handheld devices (which is often underutilized) to be shared with other peers in current vicinity, thus better exploiting the increasing availability of computing/memory/bandwidth-related capabilities at portable wireless terminals. In this line, we found the approach proposed in (Bellavista and Giannelli, 2010) where a middleware named RAMP (*Real Ad-hoc Multi-hop Peer-to-peer*) is described. In particular, RAMP combines network-layer solutions and application-layer approaches to support Internet connectivity sharing in spontaneous networks. Specifically, RAMP creates multi-hop paths toward border nodes (i.e., nodes directly connected to the traditional Internet and offering part of their underutilized connectivity to nearby peers), so that each node can use the path currently deemed as the most suitable, e.g., because it provides largest bandwidth or requires lowest power comsumption. This approach leads to a significant routing overhead when exploiting different multi-hop heterogenous paths traversing the same node.

At the application layer we found other approaches that go beyond the solutions proposed in RAMP, which have been designed for vehicular ad-hoc networks taking inspiration from BitTorrent-style P2P file sharing systems (Nandan et al., 2005; Lee et al., 2007; Chen and Chan, 2009; Lee et al., 2006; Eriksson et al., 2008). The goal is not only to exploit the connectivity of one terminal from the rest of devices, but to collaboratively download different chunks of the same content during periods of connectedness. All these application-layer protocols are aimed at enabling (collaborative) downloads of contents that typically are appealing to all (or most of) the vehicles connected to the VANET. The contributions that we are pursuing in SCC are located in a lower layer, where the goal is to aggregate the connections of several nodes in a transparent way, without conditioning besides the usage of the downloaded contents by those nodes (covering, e.g., scenarios where the accessed information is useful for just one node in the SAN). To this aim, we must envisage tranport-layer solutions that deal with multiple connections and multi-hop communications in diverse mobile ad-hoc networks, by working on the top of our enhanced virtualization layer, which, to the best of our knowledge, is approach completely novel in literature.

Beyond the networking capabilities, in the vehicular environment it is also possible to find new service models that allow vehicles to share to each other onboard storage facilities (STaaS: Storage as a Service), computing power (CaaS: Computing as a Service), services (about traffic information, driver safety or weather and road conditions) that are assembled from the information collected by other vehicles (COaaS: Collaboration as a Service), and advanced functionalities related to provision of entertainment as a service on the road (ENaaS: ENtertainment as a Service) or taking of photos and recording of videos in particular places and at specific times (PicWaaS: Pictures on a Wheel as a Service), as described in (Arif

et al., 2012). These services can be deployed over diverse vehicular clouds, ranging from static clouds (which aggregate the capabilities of parked cars) and semi-static clouds (involving vehicles stopped for a moment because of a traffic jam) to mobile clouds (the most common option where a large amount of vehicles travel on the road). The most sophisticated approaches have been designed in static and semi-static clouds, while the challenges derived from the frequently changing topologies of mobile clouds have not received the same attention (Gerla et al., 2014). The goal of our proposal is to handle the mobility of the nodes connected to our SANs (both pedestrians and vehicles), by exploiting the virtualization refinements and the mechanisms envisaged in the SCC to face the communication errors and data loss noticeable in highly dynamic ad-hoc communication environments (Arif et al., 2012).

4 METHODOLOGY AND STAGE OF THE RESEARCH

After an in-depth review of the state-of-the-art, the first step to tackle our research problem has been the development of a simulator (whose high-level design is sketched in Section 4.1), which is aimed at validating the procedures of the VNLayer+ (Section 4.2) and the "X"aaS service models of the SCC paradigm (Section 4.3). While the simulator has been totally implemented, our ongoing work is focused on the foundations of the VNLayer+ and the specific mechanisms of some "X"aaS models of the SCC.

4.1 A SAN Simulator

Covering vehicular, pedestrian and mixed environments requires that our simulator (i) models the mobility requiremens of each application scenario, and (ii) deals with the communications among the mobile nodes. To this aim, we have revised diverse pedestrian and vehicular mobility models defined in literature (Sharma and Singh, 2013), with the goal of selecting the ones that represent realistically the behaviours of the moving nodes that are connected to our SANs. Regarding the pedestrians, as depicted in Figure 2, we have chosen three different models to generate diverse types of mobility traces, referred both to individuals and groups.

- The *Random Walk Mobility Model* allows the nodes to move randomly without restrictions (e.g. a pedestrian walking a street in a city), so that the destination, speed and direction are all chosen independently of other nodes.
- The *Nomadic Community Mobility Model* considers groups of nodes that collectively move from one point to another. This model is especially appropriate for scenarios where a group of pedestrians move together, but each individual could also roam around a particular location individually (e.g. a group of tourists who visit together the historical centre of a city). By adjusting the corresponding parameters, it is possible to control how far each node can roam from each reference point, thus resulting into very realistic movements.
- Finally, we adopt the *Reference Point Group Mobility Model* where each group is composed of a number of members and one leader, so that the movements of the leader determine the mobility behaviour of the entire set (e.g in a mobility scenario where a group of students visit a museum, being guided by an expert).

As depicted in Figure 2, the above mobility models have been integrated via the existing simulation tool MobiSim[1], whose modular architecture allows to easily add extra models and trace formats. Regarding the generation of vehicular mobility traces, we have resorted to SUMO[2], due to the possibility of adding new mobility models and submitting realistic (vehicular) mobility traces in NS-2 format.

Also, we have decided to adopt NS-3 because this simulator greatly improves NS-2 in terms of efficiency, memory management and kernel architecture, besides making easier the integration of third-group software and the definition of new mobility models by using C++. Certainly, the models implemented in NS-3 are too simple in order to fulfill a wide diversity of mobility requirements. However, our simulator overcomes this limitation thanks to the (pedestrian and vehicular) mobility behaviours modeled by the external simulators MobiSim and SUMO. As these behaviours are modeled as NS-2 traces, NS-3 uses a NS2MobilityHelper module to convert them to NS-3 mobility events. As seen in Figure 2, NS-3 supports protocols such as UDP, TCP, IP and multiple routing protocols for mobile ad-hoc networks. Considering our virtualization mechanisms requires to include two additional modules aimed at implementing the virtual layer level (VNLayer+) and a virtualized routing protocol grounded on it (VNRouting), thus greatly improving the performance of the ad-hoc network. Lastly, the lowest level hosts diverse versions of the IEEE 802.11 protocol (such as IEEE 802.11p specifically developed for vehicular networks).

[1] http://www.masoudmoshref.com/old/myworks/documentpages

[2] http://sumo.sourceforge.net/

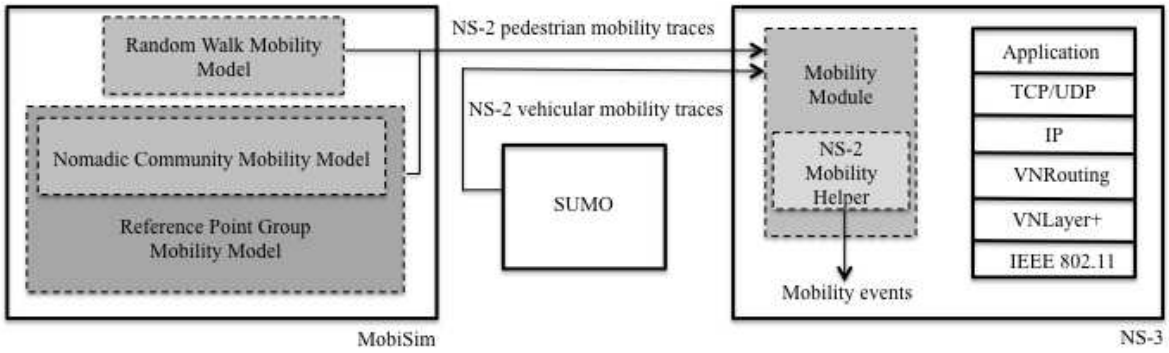

Figure 2: High-level design of our SAN simulator.

In conclusion, the exploitation of synergies among MobiSim, SUMO and NS-3 makes it possible to easily define multiple and diverse mobility scenarios where we will explore the potential of the SANs contributed in our proposal.

4.2 The VNLayer+

As depicted in Figure 3, the VNLayer+ divides the geographical area of the SAN into regions (denoted as cubes), so that a virtual node is located in each region. Analogously to what we commented for the VNLayer by Dolev et al., each virtual node is supported by several moving nodes (both pedestrians and vehicles in our scenarios), so that the virtual node stops to work after all the physical nodes leave the region. Also, there exist in each region leader and backup nodes in order to maintain replicas of the virtualization-related state information and the routing tables tackled by the routing protocols working on the VNLayer+.

We have focused on the envisage of procedures aimed to face the inefficiency sources identified in the traditional VNLayer (recall Section 3.1). Next, we sketch the ideas considered in our refinements, whose details can be found in (Bravo-Torres et al., 2015):

- First, we have developed a new leader election procedure to prevent from the slow reaction of the VNLayer to leader withdrawls, which impinged heavily on the communications in scenarios of high mobility, since the VNs were down during a non-negligible portion of the average time that the vehicles would remain in the respective region. Briefly, the leader election procedure is driven by different types of events (message receipts, timeouts and region changes), which take each PN to multiple states. Our approach consists of removing some of these states and reorganizing the transitions between the remaining ones in order to speed up the discovery of the new leader.

  Besides, we are also interested in sophisticating the election of VN leaders so that the role is dynamically transferred to the physical node that is most likely to remain longest within the corresponding region (as inferred from information coming from either the link layer or the applications layer).

- The backup designation procedure proposed in the VNLayer needs improvements too. In particular, our goal is to ensure that the number of backup nodes in a region stays, whenever possible, within a given minimum (to guarantee the resilience of the virtual nodes) and a given maximum (to avoid excessive synchronisation overhead). To this aim, our idea is that the leader reports the number of backups in the region, so that other non-leaders can learn whether they should offer themselves to further support the VN. This way, becoming a backup is no longer a fortuitous and autonomous decision as in Dolev et al.'s approach, but rather an informed and supportive one.

- Last, we have also developed procedures to improve the management of empty regions designed in the VNLayer. In this regard, our approach is based on defining a new table (named *B_table*) whose entries contain the physical addresses of the backup nodes along with its state (synchronised or not). Specifically, *B_table* replicas are stored in all the nodes of a region, which is the key to avoid losing state information from the upper layers when a newcomer assumes leadership shortly after the previous leader has left. In this scenario, the upstart (the newcomer) does not have the state information of the virtual VN of the region, but the synchronised backup do. Combining the *B_table* and the information from the backup node, the upstart can start operating just as well as the former leader in a very short time. These mechanisms are complemented with other procedures aimed to keep the structure of our SANs stable, thus enabling that the communica-

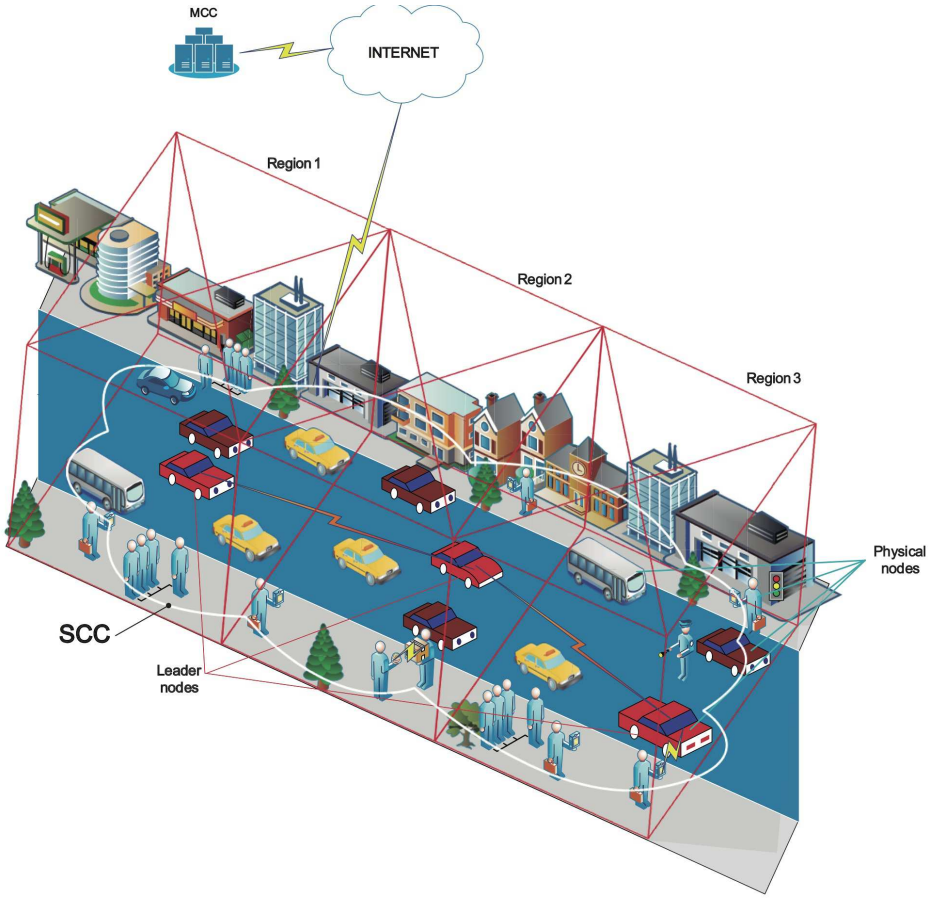

Figure 3: Sporadic Ad-hoc Network (SAN) deployed in a generic mobility scenario involving pedestrians and vehicles.

tions and services over the ad-hoc networks are not interrupted as users move. For that purpose, when a virtual node is about to become inactive (because there are very few devices in that region), the leader sends its buffered information to the leader node of a neighbor region. This information is stored until the original virtual node is operative again (i.e. until new terminals enter that region). At this moment, the just-restored virtual node requests the information and processes it by resorting to the capabilities provided by the new terminals that are located in its region, thus preventing from losing information as nodes move.

4.3 The SCC Paradigm

The devices connected to the SANS require coordination mechanisms to orchestrate the sharing and allocation of their available resources (and even of extra resources available from the Internet). To this aim, the virtual nodes of the VNLayer+ must establish communications to each other in order for the users' devices (i) to request resources to other terminals and (ii) to advertise those that they are willing to provide to the SAN. We have designed a transport-layer approach aimed at sharing and allocating resources, on which the multiple "X"aaS services of our SCC paradigm will be grounded. Each of these service models will require particular refinements (on which we are focusing our ongoing research work) that will be develop on the common substratrum described in this section.

For our descriptions, we assume that the geographic area where the SAN has been deployed is divided into regions (recall Figure 3), whose leaders and backup nodes have been selected by the mechanisms of the VNLayer+ mentioned in Section 4.2. In this scenario, we suppose that a terminal connected to the SAN (hereafter, application node) needs extra resources for running an application and asks for them to the sporadic cloud. This process is organized as follows, as depicted in Figure 4:

- Firstly, the application node broadcasts a Resource Discovery Message (`MRDiscovery`) in its region.

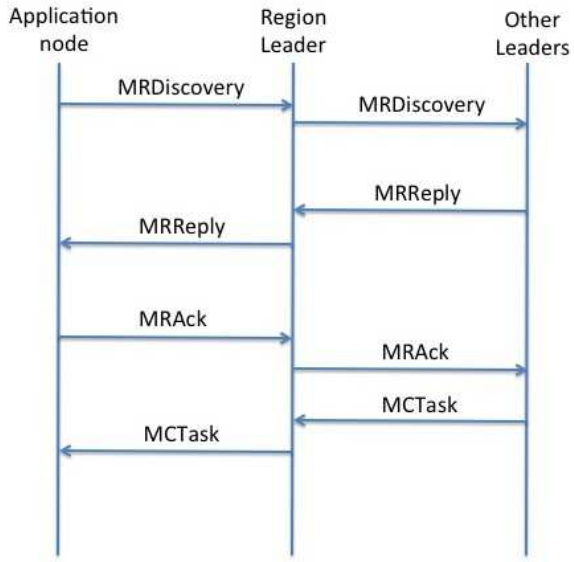

Figure 4: Messages exchanged among the application node and the leaders of the regions identified within a SAN.

- Upon the reception of the `MRDiscovery` message, the leader node of the region sends it to the leaders of its adjacent regions. This process is repeated by the remaining leaders until reaching the last region.
- In each region, the leader includes in the `MRDiscovery` message information about the resources available in the devices supporting that virtual node. The leader of the last region aggregates all the responses and sends this information back to the application node via a Resource Reply Message (`MRReply`) through the leaders of the intermediate regions.
- After receiving the `MRReply` message, the application node distributes tasks among the virtual nodes, considering the availability of resources reported by the leaders of their respective regions. The corresponding task assignment is notified to each leader via a Resource Acknowledgement Message (`MRAck`). This allocation process changes as per the specific "X"aaS service deployed, which at the same time depends on the kind of resources required for the application node (computing, storing, networking...).
- Since virtual nodes might need to cooperate when it comes to getting the information required by the application node, each leader records the tasks to be done by the rest of leaders. This information is also reported to the backup nodes of each leader in order to ensure a correct synchronization among them, so that a backup can take the place of the current leader when this node leaves the region.
- After receiving the `MRAck` message, each leader distributes its task assigment among the physical nodes of the region, by considering the capabilities that these devices put at disposal of the SAN at this moment. Once the terminals have finished their job (which depends obviously on the capabilities shared in each "X"aaS service), the leader of this region is notified. This node finally informs to the application node by sending a Completed Task Message (`MCTask`).
- This way, the cooperation among the virtual nodes enables to get the information required by the application node. Depending on the application to be runned in this node, our approach allows to replicate this information in special virtual nodes of the SAN, so that other application nodes can also access it. Specifically, to this aim, we resort to very stable virtual nodes which are supported by a huge amount of connected devices that are located, for example, in crowded squares or avenues and intersections with a lot of vehicular traffic. In order to access these contents, each node just needs to send a Content Request Message (`MCRequest`) to the leader of its region to trigger the process. The information is finally received via a Content Response Message (`MCResponse`).

Note that above descriptions are also valid for a scenario where multiple application nodes ask for resources simultaneously. In this case, the capabilities available in each virtual node will be considered when deciding about new requests, and, once the resources provided by the terminals are about to finish, the requests will be queued until the resources blocked by other application nodes are released.

As introduced before, having developed the common substratrum of SCC, we need to envisage the specific mechanisms required in each "X"aaS service. Currently, we are working on the N(etworking)aaS model with the goal of allowing the mobile nodes to collaboratively download contents from the Internet and to share them with the rest of members over the SAN. To this aim, we require transport-layer solutions that manage multiple connections over the multi-hop ad-hoc network in a transparent way. These mechanisms should lead to significant improvements in terms of download time, thanks to the simultaneous exploitation of the Internet connections provided from the terminals of several individuals (e.g. if we have 3 terminals with a bandwidth of 1 Mbps each one to download a content of 3 MB, the download would last 3 seconds using just one connection, and 1 second splitting the content in three chunks of 1 MB each one and aggregating the 3 connections).

5 EXPECTED OUTCOME

In our doctoral proposal, wireless mobile nodes opportunistically harness multi-hop ad-hoc paths to exploit the availability of a great amount of (often underutilized) resources that are provided by the terminals of nearby "always-on" users in the context of the SANs.

On the one hand, such SANs promote shared experiences among potentially like-minded users who happen to be close to each other, by relying on direct or hop-by-hop ad-hoc communications. On the other one, in the realm of these ad-hoc networks, our new SCC paradigm enables the deployment of multiple context-aware applications and "X"aaS service models whose novelty is grounded on two features. First, the fact of working with static virtual nodes instead of mobile real nodes, which makes easier the routing tasks and ensures reliable and stable communication environments. Second, the possibility of bringing together in a transparent way (i) the resources provided by each terminal that is connected to the SAN and (ii) the capabilities of the traditional MCC (if available) in diverse ad-hoc mobility scenarios, working at the transport layer and on the top of the virtualization mechanisms.

To put it from another angle, the resource sharing and allocation pursued in our SCC allows to improve the experience of each individual thanks to the capabilities contributed by the rest of members of the SAN, notwithstanding their mobility. This way, the users might, for instance, enjoy connectivity of the Internet and even reduce download times by aggregating the 3G/4G connections of other terminals, use extra storage space (in these devices or even in the cloud) and also exploit additional computational resources provided by more powerful terminals in order to run, for example, (demanding) personalization strategies. Such strategies would allow to provide the SAN users with contents of their interest which might have been proactively selected as per their preferences (and even collected, enhanced and shared by other individuals) within tailor-made sporadic communities.

ACKNOWLEDGEMENT

This work is being funded by the Ministerio de Educación, Cultura y Deporte (Gobierno de España) research project TIN2013-42774-R.

REFERENCES

Arif, S., Olariu, S., Wang, J., Yan, G., and Khalil, I. (2012). Datacenters at the airport: reasoning about time-dependent parking lot occupancy. *IEEE Transactions on Parallel and Distributed Systems*, 23:2067–2080.

Bellavista, P. and Giannelli, C. (2010). Internet connectivity sharing in multi-path spontaneous networks: Comparing and integrating network- and application-layer approaches. In *MOBILWARE'10, 3rd International ICST Conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications*, pages 84–99.

Bravo-Torres, J. F., López-Nores, M., Blanco-Fernández, Y., Pazos-Arias, J. J., and Ordóñez-Morales, E. F. (2015). Vanetlayer: A virtualization layer supporting access to web contents from within vehicular networks. *Journal of Computational Science*, In press.

Chen, B. B. and Chan, M. C. (2009). MobTorrent: A framework for mobile internet access from vehicles. In *INFOCOM'09, 24TH IEEE Conference on Computer Communications*, pages 1404–1412.

Dinh, H. T., Lee, C., Niyato, D., and Wang, P. (2013). A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless Communications and Mobile Computing*, 13(18):1587–1611.

Dolev, S., Gilbert, S., Lynch, N. A., E., E. S., Shvartsman, A. A., and Welch, J. L. (2004). Virtual mobile nodes for mobile ad hoc networks. *Distributed Computing*, 3274:230–244.

Eriksson, J., Balakrishnan, H., and Madden, S. (2008). Cabernet: Vehicular content delivery using WiFi. In *MobiCom'08, 14th ACM International Conference on Mobile Computing and Networking*, pages 199–210.

Gerla, M. (2012). Vehicular cloud computing. In *MED-HOC-NET'12, 11th Annual Mediterranean Ad-Hoc Networking Workshop*. IEEE Press.

Gerla, M., Wu, C., Pau, G., and Zhu, X. (2014). Content distribution in VANETs. *Vehicular Communications*, 1(1):23–12.

Kuss, D. J. and Griffiths, M. D. (2011). Excessive online social networking: Can adolescents become addicted to facebook. *Education and Heath*, 29(4):68–71.

Lee, K., Lee, S. H., Cheung, R., Lee, U., and Gerla, M. (2007). First experience with CarTorrent in a real vehicular ad-hoc network testbed. In *2007 Mobile Networking for Vehicular Environments*, pages 109–114.

Lee, U., Park, J. S., Yeh, J., Pau, G., and Gerla, M. (2006). Code torrent: Content distribution using network coding in VANET. In *DIWANS'06, Workshop on Dependability Issues in Wireless Ad Hoc Networks and Sensor Networks 2006 (part of Mobicom 2006)*.

Nandan, A., Das, S., Pau, G., Gerla, M., and Sanadidi, M. (2005). Cooperative downloading in vehicular ad-hoc networks. In *WONS'05, 2nd International Conference on Wireless On-Demand Network Systems and Services*, pages 32–41.

Olariu, S., Hristov, T., and Yan, G. (2013). The next paradigm shift: from vehicular networks to vehicular

clouds. In Basagni, S., Giordano, S., and Stojmenovic, I., editors, *Mobile ad-hoc networking: cutting edge directions*. John Wiley & Sons.

Przybylski, A. K., Murayama, K., DeHaan, C. R., and Gladwell, V. (2013). Motivational, emotional, and behavioral correlates of fear of missing out. *Computers in Human Behavior*, 29(4):1841–1848.

Sharma, A. and Singh, E. J. (2013). Mobility models for manet: Mathematical perspective. *International Journal of Advanced Research in Engineering and Applied Sciences*, 2(5):59–68.

Towards Domain Model Optimized Deployment and Execution of Scientific Applications in Cloud Environments

Fabian Glaser
Institute of Computer Science, University of Göttingen,
Goldschmidtstraße 7, 37077 Göttingen, Germany
fabian.glaser@cs.uni-goettingen.de

Existing solutions for automatic scaling of applications in the cloud focus on the requirements of web services. A number of application servers is deployed, a load balancer is utilized to distribute the requests to these application servers, and new application servers are launched and configured when the requests exceed a certain capacity. However, the requirements for scaling scientific applications in a cloud are different. Often, these applications are used by a single scientist and the computational load is defined by the complexity of the model to be computed rather than by the number of users. In this paper, we present an alternative approach to scale scientific applications in the cloud. Hereby, the deployment scaling is driven by a domain model defined by the scientist.

1 RESEARCH PROBLEM

Todays scientific research and simulations largely depend on computing resources. While grid computing offered the possibility to share and combine large computing resources already in the past, cloud computing offers more flexibility and automatic scaling features when deploying distributed applications. Large-scale international projects, e.g., Helix Nebula[1] and the EGI Federated Cloud[2] exist, that leverage the usage of inter-connected clouds for the scientific community. While cloud computing has penetrated many business domains, the adoption in the scientific community has not been that enthusiastic. Bunch at al. (Bunch et al., 2012) identify three major reasons for this observation:

1. Cloud systems are diverse and code written for one cloud platform can not easily be ported to another platform ("vendor lock-in").
2. Current cloud systems are designed for the execution of applications from the web service domain.
3. Using cloud computing requires user expertise, rendering it inaccessible for non-computer scientists.

The first issue has been addressed recently, by introducing techniques known from *Model-Driven Development* (MDD) to the design of cloud applications (see e.g. MODAClouds (Ardagna et al., 2012)) and also by the proliferation of standards like the *Topology Orchestration and Specification for Cloud Applications* (TOSCA) (OASIS, 2013) for Cloud Orchestration and the *Open Cloud Computing Interface* (OCCI) (Nyren et al., 2011). The two later points remain as open issues. To address these issues, we introduce a framework that uses information from the domain of the scientist to appropriately scale scientific applications in cloud environments. By leveraging the information encoded in the problem definition to scale the infrastructure, we enable scientists to focus on their domain rather than being distracted by complicated cloud internals.

The remainder of this paper is structured as follows. Section 2 summarizes the research objectives of this project. Section 3 introduces three applications from different scientific domains that motivate our research, while Section 4 summarizes the current state of the art in modeling and deploying (scientific) applications in cloud environments. In Section 5, we discuss the requirements and restrictions that are defined by our motivating applications on cloud deployments and in Section 6, we introduce a framework that aims to provide a foundation to solve the research questions defined in Section 2. We comment on drawbacks of the suggested solution in Section 7. Section 8 defines the expected outcome and finally in Section 9, we summarize the current state of our research.

[1]Helix Nebula: http://www.helix-nebula.eu

[2]EGI Federated Cloud: https://www.egi.eu/infrastructure/cloud/

2 OUTLINE OF OBJECTIVES

To be able to fully leverage the benefits of cloud computing in science, we identify the following research questions:

RQ1: How can we shield the individual scientist who is willing to deploy his application in a cloud from complicated cloud internals?

RQ2: Which parameters from the domain model of the scientist have an influence on the required scale of the cloud infrastructure?

RQ3: How can these parameters be utilized to scale the cloud infrastructure conveniently?

To answer these questions, we propose a framework that uses models to deploy applications in cloud environments and utilizes information from the domain of the scientist to appropriately scale the deployed infrastructure.

3 MOTIVATING EXAMPLE APPLICATIONS

In the scope of our project, we target three different scientific application that are candidates for being executed in a cloud environment. These applications utilize different software stacks, which will be introduced in the following.

3.1 Monte-Carlo Simulation and Data Analysis in High Energy Physics

Experimental particle collision data collected by the experiments at the *Large Hadron Collider* (LHC)[3] at CERN is produced at a data rate of $\sim$15 PB/year. To establish a ground truth, particle collisions data in the same order of magnitude is generated with help of Monte-Carlo methods according to the standard model and its variations. Currently, this data is stored and analyzed with the help of a multi-tiered grid[4], which spawns the whole globe. However, due to the update of the LHC in 2013, the data rate will increase dramatically. Hence, the high energy physics community is searching for new computing models and extending the resources with cloud resources is a viable option. Production of Monte-Carlo data by simulation and also analysis of data can be easily parallelized since each simulated event (a particle beam crossing) can be simulated and analyzed independently. Cloud computing is a valuable solution for scientist conducting analysis on smaller filtered datasets (up to a few TB in size). The parallel ROOT facility (PROOF) (Ganis et al., 2008) is a commonly used tool for conducting the analysis on computing clusters built from commodity hardware. It is both parallelized (using multiple threads on multiple kernels) and distributed (master/worker architecture).

3.2 Modeling and Optimization of Public Transport Networks

LinTim[5] is a software framework, developed by the optimization working group at the University of Göttingen. It aims to support the different planning stages in public transport networks. Hereby, planning and optimization of the networks is broken up into five stages which are network design, line planning, timetabling, vehicle scheduling, and delay management. Each stage involves modeling the problem as an optimization problem which is then either solved by a heuristic or by third-party solvers. LinTim supports the solvers Xpress[6], Gurobi[7], and Cplex[8]. It is used to steer the execution of the solving steps and feeding the output data from one step into the other. LinTim itself is implemented to run on a single-core machine. The external mathematical solvers are optimized for multi-core machines. The runtime of a solving step is largely influenced by the complexity of the optimization problem.

3.3 Material Science

OpenFOAM[9] is a software toolbox, which was primarily created for numerically solving systems of partial differential equations (PDEs) of continuum mechanics problems on a predefined geometry and domain. Its inter-process communication is based on Open MPI[10]. Thereby, solving the system of PDEs, typically involves the following steps: definition of the PDEs, definition of the geometry of the domain, definition of a mesh that covers that domain and decomposition of the domain for parallel computation. Different solvers can be utilized to then numerically

[3]The Large Hadron Collider: http://home.web.cern.ch/topics/large-hadron-collider

[4]The Wordwide LHC Computing Grid: http://wlcg.web.cern.ch/

[5]LinTim: http://lintim.math.uni-goettingen.de

[6]FICO Express Optimization: http://www.fico.com/en/products/fico-xpress-optimization-suite

[7]Gurobi: http://www.gurobi.com/

[8]Cplex: http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/

[9]OpenFOAM: http://www.openfoam.org/

[10]OpenMPI: http://www.open-mpi.org/

solve the PDEs, including finite volume methods and finite elements methods.

4 STATE OF THE ART

In the following, we shortly sketch the state of the art in modeling, deployment, configuration management, and automatic scaling of cloud applications.

4.1 Cloud Application Modeling

MODAClouds (Ardagna et al., 2012) targets cloud-provider independent development of cloud applications. Thereby, it supports the design, implementation and deployment of software with a cloud-provider independent modeling approach. The application model undergoes different modeling refinement steps, starting with a *Cloud-Enabled Computation Independent Model* (CIM), which identifies the basic components of a system, to a *Cloud-Provider Independent Model* (CPIM), which incorporates general cloud concepts like elements from *Software-as-a-Service* (SaaS), *Platform-as-a-Service* (PaaS), and *Infrastructure-as-a-Service* (IaaS) and finalizing with the *Cloud-Provider Specific Model* (CPSM), which includes the information on how to deploy the application on a specific cloud.

4.2 Automated Scaling in the Cloud

In general, there is a differentiation between *vertical scaling*, i.e., the resizing of virtual machines or containers and *horizontal scaling*, i.e., the multiplication of virtual machines or containers (compare, e.g., (Rajan et al., 2013)). Horizontal scaling of web-services is commonly implemented with the help of a load balancer, which distributes the load on the existing application servers. If an application needs to be scaled, is triggered by so called *user-defined rules*, which define actions, e.g., the launch of an additional application server in case a certain condition is met, e.g., the number of user requests exceeds a certain threshold.

4.3 Cloud Deployments and Cloud Orchestration

Cloud orchestration frameworks, such as Amazon CloudFormation[11], OpenStack Heat[12], and Cloudify[13] utilize reusable templates that model the infrastructure and the component structure required by a cloud application and enable their orchestrated and reproducible launch. TOSCA (OASIS, 2013) aims to provide a standardized template format for orchestration. Cloudify is set to fully adopt the standard in future versions and there are ongoing implementation efforts to build a translator to convert TOSCA to the Heat Orchestration Template (HOT) format. Scalability is supported by defining *policies* that trigger a user-defined action, when a certain condition is met. Cloudinit.d (Bresnahan et al., 2011) is a tool to support the orchestrated launch, configuration and monitoring of services in an IaaS cloud. Thereby, it differentiates between different *run-levels* of the required services, where the levels define the dependencies of services to each other: services on the same run-level have no dependencies, while there might be dependencies between services running in different run-levels. The deployment and configuration of the Virtual Machines (VMs) is steered by three user defined scripts, which is submitted to the VMs at launch time: The *startup script*, which is run at start-up to install the necessary software packages and the required configuration, the *test script*, which is used to test the system after configuration and the *termination script*, which runs, when a service is shut down. In the startup script the user is free to also use Configuration Management tools like Puppet[14], Chef[15] or Ansible[16] (see Section 4.4). To offer scientific software frameworks in the cloud, Bunch et al. (Bunch et al., 2012) introduce a domain-specific language called *Neptune*. In contrast to Cloudinit.d, Neptune is specialized to steer the deployment of specific software frameworks on top of the PaaS framework AppScale[17]. Thereby, it supports the utilization of different *High Performance Computing* (HPC) software packets for distributed computation that are often used in science, including MPI, X10 and MapReduce.

4.4 Configuration Management

While configuration management tools like Puppet, Chef, and Ansible are not cloud deployment specific, they are often used to manage large scale deployments on cloud platforms. For this purpose, they use declarative text-based languages to define the de-

[11] Amazon CloudFormation http://aws.amazon.com/cloudformation/

[12] OpenStack Heat: https://wiki.openstack.org/wiki/

[13] Cloudify: http://getcloudify.org/

[14] Puppet: http://puppetlabs.com/

[15] Chef: https://www.chef.io/chef/

[16] Ansible: http://www.ansible.com/home

[17] AppScale: http://www.appscale.com

sired configuration of a (distributed) set of physical or virtual hosts. This includes, but is not limited to installed software, permissions, and network configurations. These descriptions are then used to automatically enforce and keep the hosts in the desired state. Hereby operating-system specific details, like the utilized packet manager are transparent to the user. Wettinger et al. (Wettinger et al., 2013) define different approaches to integrate Configuration Management with Cloud Orchestration.

5 DISCUSSION

While the applications represent heterogeneous approaches from different scientific domains, they share common key characteristics, which are given below:

C1. A fixed set of existing frameworks is used to evaluate or execute input models from the distinct scientific domain.

C2. These frameworks are highly specialized and encode a high amount of domain knowledge.

C3. The frameworks are not built for being executed on a scalable computing infrastructure.

C4. The utilized frameworks are responsible for distributing the computational load.

C5. The computational load to be handled largely depends on the input model provided to the framework.

When moving scientific applications to cloud environments, these characteristics have to be taken into account. While cloud computing offers great flexibility, when creating computing infrastructures, it poses the question on how these infrastructures need to be scaled to fit the computational demand. The characteristics of the frameworks described above restrict the solution space for this problem: C1 defines the software configuration of the cloud infrastructure, C2 renders it often impossible to switch to a framework which is optimized for being executed in a cloud environment, C3 does not allow the cloud environment to scale driven by the framework, and C4 restricts the infrastructure deployment to a certain architecture. While C1-C4 enforce a certain structure on the deployed compute infrastructure, C5 defines the computational load on this infrastructure and so it should present a main source for defining its scale. In addition to the requirements introduced by the characteristics defined above, another set of requirements is defined by the scientist. This might include limitations on the overall runtime in case a certain deadline needs to be met or a certain number of backups of the output data needed to be kept for reliability. Given the characteristics above, we hence identify three main sources that define the requirements on an optimal deployment selection:

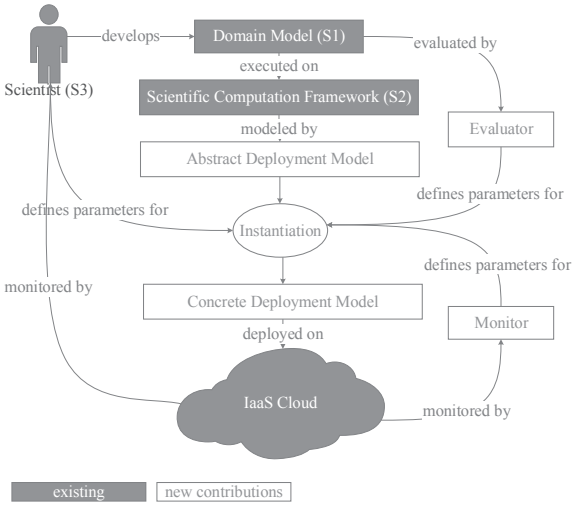

Figure 1: A framework for adapting scientific application deployments according to domain model demands.

S1. the domain model to be computed,

S2. the scientific computation framework, which is utilized,

S3. the scientist.

A framework for automatic scalability of scientific cloud applications needs to be able to leverage requirements from all three sources. In the next section, we will introduce a framework that allows to deploy and scale the application according to these observations.

6 METHODOLOGY

Figure 1 depicts the overall framework which is used to address the research questions defined in Section 2. To leverage the full flexibility of cloud computing, the framework builds on IaaS. The components shaded in grey are already existing and the research in this project is focused on the white components.
In the following, we discuss the different components and their interaction. Thereby, we exemplify the steps with help of the example application from material science defined in Section 3.3.

6.1 Domain Models

Domain models appear in very different formats. Often only a limited set of parameters defined in the domain model have an impact on the computational de-

mand e.g., the selection of a certain solving strategy might require a certain amount of RAM in the infrastructure.
In OpenFOAM the domain model is defined with the help of a dictionary file that defines the geometry, a file that defines the boundary and initial conditions of the system of PDEs, and a properties file that defines the physical properties such as the system of PDEs to be solved. An additional file is used to decompose the domain into subdomains for parallel execution. The number of subdomains thereby should match the number of processors available in the infrastructure. While in traditional compute infrastructures, this is determined by the number of available cores, in cloud environments it should be defined according to the domain model.

6.2 Scientific Computation Frameworks

In our example, OpenFOAM represents the scientific computation framework (SCF). The SCF is the most restrictive component for the cloud deployment. It encodes how the computational load is distributed on the underlying infrastructure. As described in Section 3, OpenFOAM is parallelized using Open MPI, hence it requires an MPI cluster to run and distribute the computational load.

6.3 Abstract/Concrete Deployment Model

The deployment of the SCF is modeled with help of standardized modeling languages. Using models for describing a cloud application has the advantage that the description of the application deployment becomes cloud-provider agnostic and hence avoids vendor lock-in. It enhances comprehensibility by increased abstraction, and it fosters re-usability, since a cloud-provider agnostic model can be (semi-) automatically transformed to suit the requirements of a certain cloud-provider. By using models, we address **RQ1**.
The aforementioned TOSCA standard allows to define input parameters for application models. These parameters can include the virtual machine type to use or the number of instances of a certain type to launch. We call a model with unset parameters *abstract* and a model with instantiated parameters *concrete*. The transformation from an abstract model into a concrete model is called *instantiation*. For each domain model it must be evaluated which domain model parameters have an influence on the performance in the cloud, and these parameters then need to be mapped to and reflected by the input parameters of the abstract deployment model.
The optimal setting for the parameters needed for the instantiation is determined by three sources: the scientist, an *evaluator* and a *monitor*.

6.4 Evaluator/Monitor

To find suitable parameters settings for the requirements of a specific domain model, different strategies are possible. Hereby, we distinguish between *static evaluation*, whereby the applications is not executed in the cloud, and *dynamic evaluation*, whereby the applications are executed and monitored. Static evaluations are implemented with help of a domain specific model *evaluator*. This evaluator transforms values of parameters of interest of the domain model into suitable settings of parameters for the concrete deployment model. The evaluator is domain specific and the parameters of interest in the domain model need to be carefully selected (**RQ2**). Static evaluations are done before the application is deployed on the cloud infrastructure. The concept of the evaluator addresses **RQ3**. Dynamic evaluations can be done by manual observation of the application execution in the cloud, or automatically with help of a *monitor*. According to the outcome of the deployment evaluation, the parameters that have been used for the initial deployment are readjusted and a new instantiation of the abstract deployment model is initiated. Hereby, either a new concrete deployment model is created and deployed or the existing concrete deployment model is readjusted. The second approach is similar to the Models@Runtime approach, suggested by Ferry et al. (Ferry et al., 2014).

6.5 Deployment

The deployments of the applications are fully automated to avoid manual interaction with the cloud and enable transparent application deployment for the scientist. A cloud orchestration framework is used for the orchestrated launch of the infrastructure and a configuration management tool is utilized to automatically configure the launched infrastructure.

7 LIMITATIONS OF OUR APPROACH

While the proposed framework offers the methodology to adapt application deployments to the needs of a specific domain model, it has one limitation: Domain models have very heterogeneous formats and the relation between a domain model and the deployment

model has to be defined manually for each scientific computation framework to enable automatic scaling. Hence a domain specific *evaluator* needs to be implemented for each domain. Once this mapping is done, our framework enables the scientist to focus on the development of the domain model for his problem definition and is freed from deploying the applications accordingly.

8 EXPECTED OUTCOME

We expect the following outcome of this research project:

- Contributions to the state of the art in modeling of scientific applications for the cloud,
- a novel method to leverage domain model information of the scientist to scale scientific applications in the cloud,
- a prototypical implementation of the proposed framework to demonstrate its feasibility.

9 STATE OF THE RESEARCH

In this paper, we proposed a framework for automatically scaling scientific applications in a cloud. We argue, that the traditional way of scaling applications in cloud environments does not suit the frameworks for scientific computation, since it does not take the scientific domain into account. By evaluating the domain models defined by the scientist and mapping certain key characteristics of this model to the deployment model, we are able to shield the scientist from complex cloud internals.
In the first phase of this project, we were setting up an infrastructure, to support the different steps defined by our framework. We deployed the example application defined in Section 3 in a prototypical IaaS cloud based on OpenStack[18]. A modified version of Cloudify was used for the orchestrated deployment of the applications, whereby the application models are based on Cloudify's current support for TOSCA. The configuration of the cloud applications was automated with help of the configuration management tool Ansible. Unfortunately, it became clear that current implementations of the TOSCA language are very limited when it comes to defining and launching scalable components. If TOSCA is able to properly support the scalability demands, defined in our framework is currently under evaluation.

[18]OpenStack: https://www.openstack.org/

ACKNOWLEDGEMENTS

I would like to thank my supervisor Jens Grabowski for his support and fruitful comments. This work is partially funded by the Joint Centre of Simulation Technology (SWZ) of the University of Göttingen and the Technical University of Clausthal (project 11.4.1).

REFERENCES

Ardagna, D., Di Nitto, E., Mohagheghi, P., Mosser, S., Ballagny, C., D'Andria, F., Casale, G., Matthews, P., Nechifor, C.-S., Petcu, D., et al. (2012). Modaclouds: A model-driven approach for the design and execution of applications on multiple clouds. In *Modeling in Software Engineering (MISE), 2012 ICSE Workshop on*, pages 50–56. IEEE.

Bresnahan, J., Freeman, T., LaBissoniere, D., and Keahey, K. (2011). Managing appliance launches in infrastructure clouds. In *Proceedings of the 2011 TeraGrid Conference: Extreme Digital Discovery*, page 12. ACM.

Bunch, C., Drawert, B., Chohan, N., Krintz, C., Petzold, L., and Shams, K. (2012). Language and runtime support for automatic configuration and deployment of scientific computing software over cloud fabrics. *Journal of Grid Computing*, 10(1):23–46.

Ferry, N., Brataas, G., Rossini, A., Chauvel, F., and Solberg, A. (2014). Towards Bridging the Gap Between Scalability and Elasticity. In *Proceedings of the 4th International Conference on Cloud Computing and Services Science*, pages 746–751.

Ganis, G., Iwaszkiewicz, J., and Rademakers, F. (2008). Data Analysis with PROOF. In *Proceedings of XII International Workshop on Advanced Computing and Analysis Techniques in Physics Research*, number PoS(ACAT08)007 in Proceedings of Science (PoS).

Nyren, R., Edmonds, A., Papaspyrou, A., and Metsch, T. (2011). Open Cloud Computing Interface - Core. [Available online: http://ogf.org/documents/GDF.183.pdf].

OASIS (2013). Topology and Orchestration Specification for Cloud Applications (TOSCA) 1.0. [Available online; http://docs.oasis-open.org/tosca/TOSCA/v1.0/os/TOSCA-v1.0-os.html fetched on 03/12/2015].

Rajan, D., Thrasher, A., Abdul-Wahid, B., Izaguirre, J. A., Emrich, S. J., and Thain, D. (2013). Case Studies in Designing Elastic Applications. In *Cluster, Cloud and Grid Computing (CCGrid), 2013 13th IEEE/ACM International Symposium on*, pages 466–473. IEEE Computer Society.

Wettinger, J., Behrendt, M., Binz, T., Breitenbücher, U., Breiter, G., Leymann, F., Moser, S., Schwertle, I., Spatzier, T., et al. (2013). Integrating Configuration Management with Model-driven Cloud Management based on TOSCA. In *Proceedings of the 3rd International Conference on Cloud Computing and Services Science*, pages 437–446.

Secure Data Integration Systems

Fatimah Y. Akeel[1,2], Gary B. Wills[1], Andrew M. Gravell[1]
[1]*Electronics and Computer Science, University of Southampton, Southampton, U.K.*
[2]*King Saud University, Riyadh, Saudi Arabia*
{fya1g12, gbw, amg}@ecs.soton.ac.uk

Abstract: With the web witnessing an immense shift towards publishing data, integrating data from diverse sources that have heterogeneous security and privacy levels and varying in trust becomes even more challenging. In a Data Integration System (DIS) that integrates confidential data in critical domains to contain a problem and make faster and reliable decisions, there is a need to integrate multiple data sources while maintaining the security levels and privacy requirements of each data source before and during the integration. This situation becomes even more challenging when using cloud services and third parties in achieving any part of the integration. Therefore, such systems face a threat of data leakage that compromises data confidentiality and privacy. The lack of literature addressing security in DIS encourages this research to provide a data leakage prevention framework that focuses on the level prior to the actual data integration, which is the analysis and early design of the system. As a result, we constructed SecureDIS, an architectural framework that consists of several components containing guidelines to build secure DIS. The framework was confirmed by 16 experts in the field and it is currently being prepared to be applied on a real-life data integration context such as the cloud context.

1 RESEARCH PROBLEM

Integrating personal or sensitive data sources originating from different organisations that vary in security and privacy requirements is a challenging task. The main reason for this is that the integration occurs at two different levels, one is the data level and the other is the level of the security and privacy requirements that belong to each data source. The latter raises concerns of conflict between security and privacy requirements (Yau and Chen 2008). In addition, there is an issue of difficulty in maintaining those requirements throughout the complete integration process. To further aggravate the situation, the entities providing the information, or participating in the integration, can vary in their levels of trustworthiness. Hence, the integration process should not be focused on the data level only. It should address the level of combining security and privacy requirements and consider trustworthiness.

Achieving data integration without considering maintaining the combination of the Security, Privacy, and Trust (SPT) aspects of the entities participating in the integration process leads to different threats. One important threat is *unauthorised access* to data, caused by weakness, mis-configuration, or inappropriate access controls models (Braghin et al. 2003; Watson 2007; Pistoia et al. 2007). Another example is the wide spectrum of inference attacks occurring from failure to address privacy (Fung et al. 2012; Boyens et al. 2004; Whang and Garcia-Molina 2012; Clifton et al. 2004). Yet another example is the untrustworthy behaviour caused by entities involved in the integration process, such as initiating transitive trust with other entities without the consent of the DIS (Fung et al. 2012). These threats are combined under a generic threat called *Data Leakage,* which is defined as the disclosure of confidential information to unauthorised entities intentionally or unintentionally (CWE 2013).

As mentioned, the failure to combine the SPT together in a system may allow data leakage to occur. Additionally, the mis-design of the SPT features of the system can lead, eventually, to data leakage. Therefore, data leakage threats can be controlled if the systems are designed to address SPT from the start and consider the combination of the SPT of each source and entity.

Current approaches found in the literature to secure a DIS in general, are either focused on a specific component of the system, such as the integration approach, or focused on a specific attribute over the whole system, such as privacy. However, there is a need for an approach that considers all the components of the system at the same time as to considering several attributes that contribute to the overall security of the system.

The data integration literature found covers the aspect of privacy in a specific component of the DIS, addressing privacy-preserving data integration and data mining techniques extensively. However, in terms of security requirements, there is a separate body of literature that is concerned with combining the security polices of entities collaborating together, and not particularly in data integration context(Cruz et al. 2008). Very limited literature has been found that discusses these two levels together, such as the work by Haddad, Hacid & Laurini (2012) and to the best of my knowledge, no literature has addressed the combination of SPT in a DIS context. In any case, many approaches simply assume that the entities collaborating or integrating are trustworthy.

This lack of literature encourages this research to investigate the security in data integration contexts, and to focus on the level prior to the actual data integration, which is the analysis and early design of the system.

2 OUTLINE OF OBJECTIVES

The main objective of this study is to find a solution that allows integration, collaboration, and data sharing between different organisations while maintaining individual security policies of the participating entities. We argue that maintaining the confidentiality and protecting privacy while considering trust in each entity in the DIS with a middle layer (e.g. cloud) will reduce the threat of data leakage.

Our approach focuses on guiding the design of DIS to include confidentiality, privacy, and trust aspects. This can be achieved by the following stages:

- Identifying the architectural components of DIS with a middle layer.
- Identifying the possible data leakage locations within the DIS architecture.
- Identifying the confidentiality, privacy and trust weaknesses that cause data leakage threats in DIS components.
- Creating a framework that contains the DIS architectural components.
- Creating an initial set of guidelines that aim to reduce possible data leakage threats.
- Linking the guidelines with the appropriate framework components.
- Confirming the framework and its guidelines with experts in the field.
- Using the framework on a cloud-computing context to assess its usefulness in reducing threats of data leakage.

3 STATE OF THE ART

This section provides the scope of the topics covered by this study and defines the key concepts. In addition, it discusses the themes of the reviewed literature and provides a critique relevant to the scope of this study.

3.1 Scope and Definitions

Data integration systems are usually complex (Russom 2008) and have different variations and forms. Therefore, to manage this study, the scope is focused on DIS that use a middle layer to manage the interaction between data sources and data consumers and achieve integration. The data sources used in such systems originate from different organisations and hence they vary in security and privacy requirements and trust levels.
The important aspects of the scope are defined as follows:

Security: is usually defined as the combination of confidentiality, integrity and availability (ISO 2014). The attribute of concern in this study is confidentiality; other attributes are assumed to be implemented. Confidentiality is achieved by limiting access to data to aauthorised individuals, entities and processes.

Privacy: is concerned with protecting personal information (Gollmann 2006) and determining when, how and what type of information can be exposed to others (Jawad et al. 2013). The attribute of concern is anonymity, to ensure that personal information is not exposed.

Trust: is the belief that an entity will behave in a predictable manner by following a security policy (Ross et al. 2014).

Data leakage: is disclosing private information intentionally or unintentionally to unauthorised parties (CWE 2013).

3.2 Securing DIS Components

Few works in the published literature have suggested securing DIS as a whole by considering privacy and trust. However, trust is still an issue in distributed systems. Prakash & Darbari (2012) discuss several security approaches that aim to enforce trust, such as the use of trust models. They also discussed risk management as a method to evaluate trust. Van Den

Braak et al. (2012) propose a framework that aims to support data sharing among different public organisations. It preserves privacy through sharing data based on a need-to-know principle, where data is provided only when required for a specific process. The authors propose the notion of a Trusted Third Party (TTP). The TTP is responsible for integrating and sharing data between different government organisations. The proposed framework contains two parts: the first part is data integration techniques to achieve privacy, while the second part provides guidelines on data sharing that ensure security and trust. Nevertheless, the guidelines provided mainly focus on the *Integration Location* and *Data and Data sources* components of the DIS rather than the system as a whole. However, the integration covered was across government organisations, and thus, still under the same security policies; therefore, the risks of violating the integrated security policy were not present as one of the security and privacy challenges that the approach overcomes. In addition, in this approach, trust is assumed, and it lacks guidance on the need to establish trust or evaluate it in relation to other entities. Therefore, when security and privacy challenges are discussed in this work, they do not include trust threats because it is assumed in the first place, and those challenges are not particularly addressed as data leakage threats. Finally, although the proposed guidelines are reasonable to prevent data leakage threats, they are not linked to any specific phase of the software development and therefore it is not clear when to apply the guidelines to the SDLC.

The approach proposed by Clifton et al. (2004) is a privacy framework for data integration. The purpose of the framework is to provide an insight into the privacy challenges in data integration. It provides several research directions, one of them emphasizes the need for a privacy framework that considers users privacy views to expose and hide sensitive attributes, privacy policies implementing these views and a purpose statement specifying which data is allowed to be accessed and integrated. The solutions discussed to preserve privacy in data integration consider the following components: *data and data sources*, *integration approach*, *data consumers* and *security policy* only. The integration location and the management of the process of the integration are not addressed within the framework. In addition, it is not presented with any link to software development. It should be mentioned that the authors have addressed data leakage mainly through discussing the difficulty of preventing multiple query attacks. The heterogeneity in the security and privacy policies are only briefly addressed and there is no specific focus on them in terms of their relation to the *Integration Approach*. Trust, on the other hand, is not addressed at all as one of the challenges within the framework.

The work of Bhowmick et al. (2006) is very similar to that of Clifton et al. (2004); however, it proposes a more detailed architecture and a framework for privacy-preserving data integration and sharing deployable DIS. It includes security and policy considerations by suggesting adding a security policy component to the system. It also provides several suggestions on preserving privacy in different DIS components. The architecture covers most of the DIS architectural components except the management. However, the level of detail provided in terms of integrating the various security policies of the data sources and the integration location is not sufficient. In addition, the suggestions provided are not listed in the form of technical or practical security guidelines. It also does not present the framework from a clear specific development phase.

A policy integration method that combines the authorization policies of data sources integrating and sharing data is proposed by Haddad, Hacid, & Laurini (2012).The method focuses on creating a global security policy that consists of local security policies of the participating data sources in the integration process. This global policy is enforced within the mediator level i.e. *Integration Location* component. According to this approach, queries will not be processed unless they are authorised by a source. Therefore, the access to sources will be preserved. One of the limitations of this approach is that it covers the security policies generated by *Data Sources* and the *Integration Location*; but it does not consider the actual data *Integration Approach* and the *Data Consumers,* nor *Management*. In addition, it does not consider the trustworthiness of the entities participating in the integration process. Furthermore, it does not explicitly specify the software development phase in which the approach can be used.

Another work by Jurczyk and Xiong (2008) focuses on privacy preserving data integration. It proposes several protocols for data anonymization in addition, to a general architecture for data integration. Hu and Yang (2011) propose a privacy protection model for DIS by using semantic approaches. The works reviewed in this section are primarily focused on privacy, which can be related to security, and little or no attention to trust is given

in these works. In addition, the approaches provided are applicable to relational databases and do not consider other data formats.

Generally, the literature provides practical and applicable solutions expected to solve problems in a specific DIS component. However, the security of the whole system is discussed only in a limited way in the literature. Studies usually assume that the provided data is secure and comes from trustworthy entities. However, from a security perspective, data sources are considered an important and effective element to guarantee the security of a DIS. Data sources can therefore fall under the data-centric security category within the information security field, and having security-meta data would help in distinguishing secure sources from unsecure ones.

In organisations that employ data integration to integrate private data, there is a need to manage data access and authorization. A carefully selected access control model that enforces security policies is essential. Therefore, organisations need to create well-defined security policy that enforces data security, privacy and protection. To ensure the security of a DIS, the combination of the individual security policies of each data source needs to be carefully considered. There are many studies of security policies and access controls that cover policy integration in different contexts; however, there is an evident lack in considering trust. One possible reason is that organisations usually integrate data coming from their own data sources, which are assumed to be trustworthy.

In DIS, the resulted integrated data are normally requested using queries by data consumers, which can be humans or services. Data consumers are an important component of any DIS, as they can be a source of security and privacy violation. Many attacks on information systems, including DIS, are caused by data consumers, such as SQL-Injections to gain access to data that they are not authorised to. In the DIS context specifically, inference attacks and attribute correlations that lead to data leakage threats are usually carried out by data consumers. In addition, the consumer use of access control models that are not well evaluated leads to unauthorised data access. Therefore, it is important to consider data consumers from a threat point of view and plan to build the system in a way that prevents such attacks. The literature on DIS threats and attacks focuses on privacy attacks conducted by data consumers; other attacks are not discussed as they do not differ fundamentally from any other web-based applications attacks.

3.3 Integration Borders

This theme relates to the differences between integrating data within or outside an organisation and the effect this has on the security, privacy and trust of the data integration process.

3.3.1 Integrating Data within the Organisation

There are several domains where data is requested and integrated within the border of a single organisation or within a range of similar organisations belonging to the same sector. Enterprise Information Integration (EII) (Halevy et al. 2006), healthcare (Bhowmick et al. 2006) and, scientific research (Ray et al. 2009) are all examples of where this type of integration can occur.

There is a large volume of published studies on data integration that takes healthcare domains as a context, such as the works by Boyens, Krishnan & Padman(2004),Tian, Song & Huh (2011), and Eze, Kuziemsky, Peyton, *et al.*(2010). There are also several studies concerned with security and privacy issues in healthcare in general, for example the one by Meingast, Roosta & Sastry(2006). This large body of existing work makes healthcare approaches on security and privacy useful to survey. Healthcare is a unique sector, with characteristics that are not found in other sectors (Avison and Young 2007).

This means legislation and policies exist that strive to protect this kind of domain to maintain data integrity and confidentiality. In the UK, healthcare organisations have to comply with the Data Protection Act (Philip Coppel Qc 2012),whereas, in the United States healthcare organisations follow the HIPPA act (U.S. HHS 1996).

In many healthcare organisations, a DIS is systematically monitored for compliance with legislation and policy as well as other criteria (Eze et al. 2010). Reviewing 20 of 30 Health On the Net Foundation Code of Conduct (HONcode) accredited American online healthcare appointment websites for compliance with basic principles of security and privacy, Hong, Patrick, & Gillis (2008) found that only 8 of the 20 websites are secure and 12 of the 30 were not showing privacy notices to patients on their websites. They found that there is a gap between the ideal security and privacy requirements and the reality in applying them. They therefore, propose several steps to overcome this gap and make it possible for healthcare organisations to be compliant with legislation and security guidelines.

There are several requirements needed in the

healthcare domain. One is to balance security and interoperability between healthcare actors and organisations. Dawson, Qian, & Samarati (2000) suggest an approach that allows multilevel secure data sources to integrate and provide the results to external applications while maintaining their security levels. In addition to interoperability, Armellin et al. (2010) propose a system that publishes healthcare documents that provide interoperability and complies with privacy laws.

Another requirement is aiming to preserve privacy while integrating healthcare data. For example, building automatic data mashups that are aware of privacy concerns (Barhamgi, Benslimane, Ghedira, Tbahriti, et al. 2011; Barhamgi, Benslimane, Ghedira and Gancarski 2011). In addition, access controls used in healthcare systems can be extended to adapt to privacy requirements (Hung 2005).

3.3.2 Integrating Data outside the Organisation

Integrating data outside of the organisation means that the integration location or part of it is outside the organisation boundaries, for example, when data sources are handled by cloud services and/or third parties. In addition, the data sources may reside outside the organisations boundaries. The following subsections discuss each of these cases.

3.3.2.1 Using the Cloud as an Integration Location

Clouds suffer from many security, privacy and trust issues and therefore they need to comply with regulatory laws for data protection (Takabi et al. 2010; Youssef and Alageel 2012). In addition, to prevent unauthorised access, they need to deal with the heterogeneity of security components and multi-tenancy (Takabi et al. 2010). The fact that the clouds are not under an organisation's physical control elevates the problem of managing data security (Reeve 2013), especially when there are no standard rules and regulations to deploying the cloud (Saeed et al. 2014). These aspects should be considered in any data integration security model, to emphasize the importance of investigating the location of integration.

3.3.2.2 Using Third Parties

Third parties are used in data integration applications for different purposes. On one hand, organisations may want to outsource the data to a third party to analyse it and find out aggregation statistics (Xiong et al. 2007). Alternatively, an organisation may require an entity to handle access control to personal integrated data, such as the approach described by Van Den Braak et al. (2012) which uses a trusted third party to handle access controls to government data in the public sector. The proposed approach uses privacy preserving data integration and collaboration.

Harris, Khan, Paul, & Thuraisingham (2007) argue that, in general, data integration applications that handle critical data, such as emergency response and healthcare awareness, need to share their data with different organisation s to make effective decisions. The authors discuss standard-based approaches to secure data across organisations covering different types of data and different types of domains. Their work emphasizes the need to enforce security policies and create standards or guidelines to govern application in critical domains.

3.3.3 Accessing Data outside the Organisation's Boundaries

There are cases when there is a need to integrate data from public data sources with an organisation's private data sources. This integration leads to different challenges, such as the lack of a form of privacy measurement, i.e. measuring the amount of privacy lost when data is exposed (Pon and Critchlow 2005). Another work, by Yau & Yin(2008) proposes a repository for data integration across data sharing services by collecting data based on user's requirements. If the repository is compromised, only the result of the integration is revealed and the original data will remain intact. The work by Mohammed, Fung, & Debbabi (2011) proposes two algorithms to overcome the challenges of revealing data coming from different data providers, using game theory.

3.3.4 Discussion

Integrating data within the organisation could be considered safe to an extent. The reason is that many of the entities involved in the integration process are within the organisation and are under the same security measures. The data sources are well known and the integration location is within the organisation's boundary. Therefore, the concerns about security are controllable, to some extent.

However, integrating data outside the organisation can be very critical. The reason is that many of the entities involved in the data integration process use different security models and have different privacy requirements. These entities can be

data providers or integration locations, which may not be secure enough.

Integrating data coming from outside the organisation raises issues on security policies. One aspect to consider in integrating security policies, especially when integrating data outside the organisation boundary, is to include the organisation's own policies with the external policies and the government legislation related to data protection, to ensure the security of the overall system.

Some organisations need to use trusted third parties to handle their data. One possible reason is that an organisation may have to respond to a significant number of requests and cannot respond in a timely manner. Another reason would be that an organisation may have lack of technical skills or infrastructure to handle the data. It therefore, transfers this responsibility to a third party to take over consumers' requests. As a result, releasing data to trusted third parties is critical, as the organisation may not monitor or track private data processing and movement exacerbating security concerns. In the real world, companies rely on legal agreements of data disclosure. Few use technical enforcement of data movement. However, the literature lacks coverage of this specific aspect in the data integration context. One study, by Van Den Braak, Choenni, Meijer, *et al.* (2012), proposes the use of trusted third parties; however, security concerns still arise.

3.4 Covering Security, Privacy and Trust

In studies that aim to secure DIS, the focus on the Security, Privacy and Trust (SPT) aspects varies. Some studies focus on SPT as separate aspects; other studies combine two of the SPT aspects. However, only a very limited number of studies have focused on SPT combination in a DIS context. The following section investigates these studies and how they focus on the aspects of SPT.

Secure data integration, mining, and sharing are addressed in the literature as approaches to SPT *separately*. In terms of ***security in DIS***, several recent studies, including the one by Haddad, Hacid, & Laurini (2012) and Begum, Thakur, & Patra (2010), have focused on security policy integration and conflict reconciliation and their uses to answer users' queries. Other studies have proposed extensions and improvement to RBAC to adapt to the integration context, such as the work by Lamb, Power, Walker, *et al.*(2006).

However, ***privacy in DIS*** has the lion's share of research. Privacy-preserving techniques are well established in the literature, spanning a range of different topics from privacy in peer-to-peer DIS to anonymization techniques. Bhowmick et al.(2006) propose a privacy preserving DIS framework that emphasizes the need to consider the balance between privacy and data sharing. This perspective has been later addressed by many studies: the work by Pasierb et al. (2011) presents different approaches to privacy-preserving data integration in e-healthcare systems, while the same concept applies to web services and data mashups by Barhamgi, Benslimane, Ghedira, *et al.* (2011b, 2011a).

Finally, in terms of ***trust in DIS***, there are many distributed trust models that can be adopted in DIS and can be used to determine the level of trustworthiness of either data providers or third parties. The work by Treglia & Park(2009) suggests a trust framework for intelligence information sharing between agencies. Other approaches focus on computational trust using either policy-based trust or reputation-based trust (Artz and Gil 2007)which can also be applied to DIS. Other studies acknowledge the need for a *combination between areas*, for example, the work by (Hung 2005) combines security and privacy by extending the RBAC model with privacy-based extensions.

Security combined with Privacy and Trust (SPT) has recently gained attention from the research community. Systems security is complicated and influenced by many other areas and therefore it cannot be addressed solely. Morton & Sasse (2012) supports this argument by proposing an integrated SPT framework to create an effective privacy practice in any information system. Considering human factors, another work in progress (Flechais et al. 2013) also takes this holistic perspective. It discusses authentication taking into account SPT in banking context in Saudi Arabia. Other studies such as that of Manan, Mubarak & Isa (2011) emphasize the need for such a research direction. In addition, a recent work on federated identity and access management created an access control solution that encompasses SPT, considered together (Khattak et al. 2012).

This concept is applied in a limited data integration context where Van Den Braak, Choenni, Meijer, *et al.* (2012) provide a framework for sharing and integrating data among public organisations in which ensuring security and privacy is achieved by using a trusted third party to manage access controls to private data.

However, to the best of my knowledge there is no single approach to preventing data leakage in DIS

that considers SPT aspects together, although they are very important to protect confidentiality and allow sharing data in a secure fashion.

3.5 Discussion of Data Leakage in DIS

Many vulnerabilities in software are caused by flawed design (McGraw 2004). Therefore, data leakage as a generic security threat can be caused by weaknesses in DIS design. The following subsections discuss these issues from security, privacy, and trust perspectives.

3.5.1 Design Issues Related to Security

Due to the heterogeneity and distribution of DIS components, security threats to DIS can arise from any component of the system. Threats can start from data sources that may not have adequate security and privacy meta-data, which define their level of sensitivity, and therefore, the DIS may face difficulties in maintaining the data sources' policies throughout the whole system. Moreover, different trust and security concerns may also be faced in the middle layers such as the cloud, where the data is mined, pre-processed, integrated, and prepared for presentation, in addition to many different tasks. Finally, data consumers where the data is accessed and queries are answered may pose threats as well.

Security in DIS is important, as it is to any information system. Having appropriate organisational security objectives and conducting early security analysis according to well-defined security requirements help in shaping any DIS towards providing a better security. Security is a comprehensive feature that includes many attributes; however, since many approaches proposed in a DIS context mainly focus on creating global security policies, enforcing security policies using access controls, and managing access to data this study will focus on the confidentiality as an attribute security. Some examples of how confidentiality in a DIS is applied: 1) Authorization to access data sources and the results of the integration is utilized by access control; 2) Data disclosure should not be disclosed equally to all user roles but should be limited to users with appropriate roles; 3) Considering security and privacy policies associated with data sources.

However, mis-configured access control models or selecting inappropriate ones can be a major design flaw that causes systems to be unsecure. It is important to adopt a suitable access control model to guarantee authorization to access data and guarantee its confidentiality.

Another issue that increases the possibility of data leakage is lack of knowledge of DIS users. Users' ignorance about legal issues in data management allows unauthorised access to occur(Batty et al. 2010). This can include developers and data managers as well. Therefore, a proper training is required to base the DIS design on updated knowledge of data management legal issues.

3.5.2 Design Issues Related to Privacy

Several design flaws that violate privacy cause data leakage. Firstly, the data sources used in the DIS may be originally predictable and very easy to link, or it may miss very important security meta-data that defines its level of sensitivity. Therefore, the DIS becomes vulnerable to different privacy attacks. A good design needs to create data sources that are very hard to link, or, in case of external data sources, it uses techniques that refine the data to minimize the ability to link information.

Secondly, although anonymization techniques are a popular solution to privacy, they are not always sufficient, for two reasons. On one hand, it is not the Personal Identifiable Information (PII) only that a DIS needs to worry about; it should worry about all the other attributes that cause inference attacks, such as Quasi Identifier (QID). On the other hand, a DIS needs to have customised anonymity levels that are suitable for the users accessing the data, as failing to manage anonymity discloses private data (Meingast et al. 2006).

Thirdly, systems that allow multiple consecutive queries to data sources (Clifton et al. 2004), especially sensitive data sources, are prone to privacy violation, as users can use the results for inference attacks, such as inferring conditional information from non-confidential data or from statistical aggregates, as mentioned by Zhang, Zeng, Wang, *et al.*(2011). Therefore, there is a need to manage the number of queries to private data sources, along with the users' roles and authorization level.

Considering these issues in DIS design will result in minimizing threats such as inference attacks, attribute correlations, and insiders attacks.

3.5.3 Design Issues Related to Trust

Designing systems that use external data sources coming from different organisations entails trusting the entities to provide accurate and reliable information. Using cloud services that process and integrate data, in addition to providing services to query and analyse the data (Carey et al. 2012) is

very risky. Clouds are security and privacy critical and they still require more effort to increase their reliability (Saeed et al. 2014). The risks arise from using multi-tenancy public clouds that share physical infrastructure with untrustworthy users can lead to different attacks, such as cross-virtual machine attacks (Ristenpart et al. 2009).

Trusting third parties to handle data in any form needs to be carefully considered. Computational trust does not differ conceptually from human trust. A trust in entities may be gained by their reputation or by certain actions they perform to obtain trust. Data leakage can occur from transitive trust where a trusted entity reveals sensitive data to other entities (Fung et al. 2012). Third party rights on the data need to be determined (Meingast et al. 2006), and many critical questions need to be answered during the design of DIS systems, such as: Do third parties have authority over the data similar to that of the data owner?

Data transfer to clouds and third parties needs to be based on trust (Saeed et al. 2014); hence trust should be added to the design process, through considering the entities that the system deals with and conducting risk assessment and risk mitigation. A DIS should be designed considering the collaboration among other entities and enforcing trust models between these entities to guarantee security.

In a DIS context, trust is an issue to consider for data sources and integration locations as well as third parties involved in the DIS. It also extends to data consumers, where granting roles to users depends on their level of trustworthiness. Therefore, trust is an important aspect in a DIS that cannot be neglected; however, it needs to be balanced with other properties to achieve secure and reliable systems.

SPT issues are exacerbated in a DIS context due to the complex and distributed nature of a DIS, especially across organisations. These issues, which are summarized and categorized in Table 1**,** contribute to the threat of data leakage in DIS. The full threat analysis is discussed in our previous work (Akeel et al. 2014).

Table 1: Summary of DIS Threats Causing Data Leakage.

Category	Threats and Concerns
Security	Unauthorised access caused by:
	1) Inappropriate access control model
	2) Access control weakness
	3) Mi-configured access control model
	Ignorance of legal issues on data management
	Inapplicable confidentiality on merged data
	Confidentiality violations due to inconsistent regulatory laws
	Leaking data to the platform
Privacy	Inference attack: attribute linkage
	Inference attack: linking data with QID
	Inference attack: interval disclosure
	Inference attack: gathering information about queries
	Inference attack: use of non-confidential information and statistical aggregates, namely record linkage
	Inference attack: consecutive queries attack
	Insiders attack: data providers inferring data
Trust	Using clouds to process data
	Third parties rights on data
	Third parties and transitive trust

4 METHODOLOGY

The results of reviewing the literature and realising the research gap and linking that to the study main objective, the following research questions are proposed:

RQ1: What is an Appropriate Framework to Reduce Data Leakage Threats in a DIS with a Middle Layer?

Focusing on the DIS architecture with a middle layer, this research question is mainly concerned with finding an appropriate framework that helps in reducing data leakage threats. The proposed framework aims to consist of the basic architectural component of a DIS with a middle layer.

RQ2: In the Proposed Framework, What Are the Confidentiality, Privacy and Trust Guidelines Used to Reduce Data Leakage Threats?

Understanding data leakage threats and locations in the context of a DIS helps in suggesting an appropriate set of confidentiality, privacy and trust guidelines that aim to reduce those threats. These guidelines can be included in the framework under each architectural component.

RQ3: How Can the Proposed Framework and Guidelines Be Used to Reduce the Threats of Data Leakage in a Real-life Scenario?

Using the framework and its guidelines in a real scenario will help in evaluating the framework from the practicality, applicability to context and usefulness to reduce data leakage threats.

Based on the previously mentioned research questions, Table 2 summarises the research activities relevant to answer each research question.

Table 2: Research Activities.

Research Activities	Purpose	Research Question
Literature review about DIS + Expert reviews 1	Confirming the components of SecureDIS	RQ1
Literature review about data leakage threats + Expert reviews 1 and 2	Identifying data leakage threats and their locations within the DIS architecture	RQ1
Literature review about reducing data leakage	Create the guidelines that aim to reduce data leakage threats	RQ1
The synthesis and analysis of the results of each step of the previous research activities	A proposed framework with architectural components containing a set of guidelines (SecureDIS)	RQ1
Experts reviews 2	Confirming the guidelines proposed by SecureDIS	RQ2
Expert reviews 2	To find out whether the proposed guidelines are comprehensive	RQ2
Expert reviews 2	To know whether the proposed guidelines are practical	RQ2
The synthesis and analysis of the results of each step of the previous research activities	The confirmed framework and guidelines (SecureDIS)	RQ2
A case study analysis	To find out the appropriate context to use the guidelines in	RQ3
A case study analysis + study of data leakage threats	To customise the guidelines to the appropriate context and apply them	RQ3
A case study + metrics	To measure reduction in data leakage	RQ3
The synthesis and analysis of the results of each step of the previous research activities	Confirming/refuting the applicability of the guidelines in a real-life application	RQ3

5 SecureDIS: A FRAMEWORK FOR SECURE DATA INTEGRATION

This research conjectures that considering SPT together in designing a DIS will reduce data leakage threats in these systems. Hence, this section presents a Secure Data Integration System (SecureDIS), an architectural framework that consists of several components, each of which includes a set of guidelines designed specifically to prevent data leakage. The following subsections discuss how SecureDIS was created and evaluated.

5.1 SecureDIS Construction

There are two parts to SecureDIS framework components and guidelines.

5.1.1 Constructing Components

Analysis of the previous studies (Gusmini and Leida 2011; Dicelie et al. 2001; Nachouki and Quafafou 2011) that build DIS with middle layer architecture, together with understanding the implications of using cloud computing, remote servers, and third parties to achieve integration contributed significantly to the formulation of SecureDIS components. The initial components of SecureDIS are data sources, the integration location, the integration approach and the data consumers, as adapted from the previous studies.

5.1.2 Creating SecureDIS Guidelines

The outcome of the careful review and analyses of the literature created an understanding of how several research areas relate to each other and contribute to securing a DIS. A DIS encompasses different levels: a low level that includes technical details of achieving secure and privacy-preserving integration to a much higher level of managing such a system to a medium level of engineering and designing a secure DIS. Synthesizing the results of the analysis, a set of guidelines were created. These guidelines are categorised into confidentiality, privacy and trust requirements and some of the guidelines overlap between these different categories. After constructing DIS components, the guidelines were grouped under each component. Each guideline was inspected individually and in relation to other guidelines in different components. This process of refining the guidelines was iterative, as many guidelines were moved around components and grouped differently until the final version was reached. A further analysis was conducted to link each guideline to the data leakage threats found in the data integration context and some guidelines were eliminated as they were out of the scope. The initial version of SecureDIS before evaluation can be found in (Akeel et al. 2013).

5.2 SecureDIS Evaluation

To confirm the SecureDIS framework and its guidelines, two expert reviews were conducted, one for the components and another for the guidelines. The following subsections provide an overview of the results of the confirmation.

5.2.1 Confirming SecureDIS Components

Based on the results of the first expert reviews of SecureDIS components, two additional components are proposed: *Security Policy* and *System Security Management.* Security policy was separated from other components of the system due to its importance in governing the security, privacy and trust of the DIS, which is supported by the work of Bhowmick, Gruenwald, Iwaihara, *et al.*(2006). System Security Management is added to ensure security measures are in place and to manage them, which is needed in any information system. Figure 1 shows SecureDIS framework after the first expert reviews.

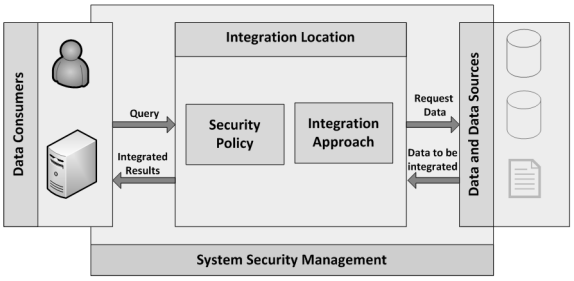

Figure 1: SecureDIS framework componenets.

5.2.2 Confirming SecureDIS Guidelines

To confirm and extend the proposed guidelines and answer the research questions, SecureDIS is planned to be reviewed by a second group of experts. The results will help in reshaping SecureDIS to an accepted version that can be useful to the system analysts and designers.

6 EXPECTED OUTCOME

The expected outcome of this study is a confirmed framework and set of guidelines, namely SecureDIS, which are comprehensive, practical and applicable to different contexts including cloud-computing environment. SecureDIS aims to help systems analysts and early designers in their analysis phase of building systems that considers security by design to prevent data leakage threats.

7 STAGE OF THE RESEARCH

The research questions RQ1 and RQ2 were answered. The current stage of the PhD is to customise and apply SecureDIS to a real-life context to assess its usefulness in preventing data leakage threats in DIS contexts. Possible case is an organisation using cloud services to integrate or store data coming from different resources. The results of the coming stage will help in answering RQ3.

REFERENCES

Akeel, F. et al. 2013. SecureDIS: a Framework for Secure Data Integration Systems. In: *The 8th International Conference for Internet Technology and Secured Transactions*. London, UK.

Akeel, F. Y. et al. 2014. Exposing Data Leakage in Data Integration Systems. *The 9th International Conference for Internet Technology and Secured Transactions (ICITST-2014)*, pp. 420–425.

Armellin, G. et al. 2010. Privacy preserving event driven integration for interoperating social and health systems. *Secure Data Management*, pp. 54–69.

Artz, D. and Gil, Y. 2007. A survey of trust in computer science and the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web* 5(2), pp. 58–71.

Avison, D. and Young, T. 2007. Time to rethink health care and ICT communications. *Communications of the ACM* (June 2007), pp. 69–74.

Barhamgi, M., Benslimane, D., Ghedira, C., Tbahriti, S.-E., et al. 2011. A Framework for Building Privacy-Conscious DaaS Service Mashups. In: *2011 IEEE International Conference on Web Services*. Washington DC, USA: IEEE, pp. 323–330.

Barhamgi, M., Benslimane, D., Ghedira, C. and Gancarski, A. L. 2011. Privacy-Preserving Data Mashup. In: *IEEE International Conference on Advanced Information Networking and Applications*. Biopolis, Singapore: IEEE, pp. 467–474.

Batty, M. et al. 2010. Data mash-ups and the future of mapping by. *JISC TechWatch*, pp. 1–45.

Begum, B. a. et al. 2010. Security policy integration and conflict reconciliation for data integration across data sharing services in ubiquitous computing environments. In: *International Conference on Computer and Communication Technology (ICCCT'10)*. Allahabad, India: IEEE, pp. 1–6.

Bhowmick, S. S. et al. 2006. PRIVATE-IYE: A Framework for Privacy Preserving Data Integration.

In: *Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06)*. Washington, DC, USA: IEEE.

Boyens, C. et al. 2004. On privacy-preserving access to distributed heterogeneous healthcare information. In: *Proceedings of the 37th Hawaii International Conference on System Sciences*. Big Island, Hawaii, USA, pp. 1–10.

Van Den Braak, S. W. et al. 2012. Trusted third parties for secure and privacy-preserving data integration and sharing in the public sector. In: *Proceedings of the 13th Annual International Conference on Digital Government Research - dg.o '12*. College Park, MD, USA: ACM Press, pp. 135 –144.

Braghin, C. et al. 2003. Information leakage detection in boundary ambients. *Electronic Notes in Theoretical Computer Science* (78), pp. 123–143.

Carey, M. J. et al. 2012. Data Services. *Communications of the ACM* 55(6), pp. 86–97.

Clifton, C. et al. 2004. Privacy-preserving data integration and sharing. In: *Proceedings of the 9th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery - DMKD '04*. Paris, France: ACM Press, p. 19.

Cruz, I. et al. 2008. A Secure Mediator for Integrating Multiple Level Access Control Policies. *Knowledge-Based Intelligent Information and Engineering Systems*, pp. 354–362.

CWE 2013. CWE-200: Information Leak (Information Exposure). [Online] Available at: http://cwe.mitre.org/data/definitions/200.html [Accessed: 2 August 2013].

Dawson, S. et al. 2000. Providing security and interoperation of heterogeneous systems. *Distributed and Parallel Databases* (8), pp. 119–145.

Dicelie, J. J. et al. 2001. Data integration system.

Eze, B. et al. 2010. Policy-based Data Integration for e-Health Monitoring Processes in a B2B Environment: Experiences from Canada. *Journal of theoretical and applied electronic commerce research* 5(1), pp. 56–70.

Flechais, I. et al. 2013. In the balance in Saudi Arabia: security, privacy and trust. In: *Extended Abstracts on Human Factors in Computing Systems CHI '13*. Paris, France, pp. 823–828.

Fung, B. C. M. et al. 2012. Service-Oriented Architecture for High-Dimensional Private Data Mashup. *IEEE Transactions on Services Computing* 5(3), pp. 373–386.

Gollmann, D. 2006. *Computer Security*. Second Edi. John Wiley & Sons.

Gusmini, A. and Leida, M. 2011. A patent: Data Integration System.

Haddad, M. et al. 2012. Data Integration in Presence of Authorization Policies. In: *IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*. Liverpool, UK: IEEE, pp. 92–99.

Halevy, A. et al. 2006. Data integration: the teenage years. In: *32nd International Conference on Very large data bases VLDB'06*. Seoul, Korea.

Harris, D. et al. 2007. Standards for secure data sharing across organizations. *Computer Standards & Interfaces* 29(1), pp. 86–96.

Hong, Y. et al. 2008. Protection of Patient's Privacy and Data Security in E-Health Services. In: *2008 International Conference on BioMedical Engineering and Informatics*. Sanya, China: IEEE, pp. 643–647.

Hu, Y. and Yang, J. 2011. A semantic privacy-preserving model for data sharing and integration. In: *Proceedings of the International Conference on Web Intelligence, Mining and Semantics - WIMS '11*. Sogndal, Norway: ACM Press.

Hung, P. 2005. Towards a privacy access control model for e-healthcare services. In: *Third Annual Conference on Privacy, Security and Trust*. Andrews, New Brunswick, Canada.

ISO 2014. ISO/IEC27000: Information technology — Security techniques — Information security management systems — Overview and vocabulary. *BSI Standards Publication*.

Jawad, M. et al. 2013. Supporting Data Privacy in P2P Systems. *Security and Privacy Preserving in Social Networks*, pp. 1–51.

Jurczyk, P. and Xiong, L. 2008. Towards privacy-preserving integration of distributed heterogeneous data. In: *Proceedings of the 2nd PhD workshop on Information and knowledge management*. Napa Valley, California, USA, pp. 65–72.

Khattak, Z. et al. 2012. Evaluation of Unified Security, Trust and Privacy Framework (UnifiedSTPF) for Federated Identity and Access Management (FIAM) Mode. *International Journal of Computer Applications* 54(6), pp. 12–19.

Lamb, P. et al. 2006. Role-based access control for data service integration. In: *Proceedings of the 3rd ACM workshop on Secure web services - SWS '06*. Alexandria, VA,USA: ACM Press, pp. 3–12.

Manan, J. A. et al. 2011. Security, Trust and Privacy–A New Direction for Pervasive Computing. In: *Proceedings of the 15th WSEAS international conference on Computers*. Stevens Point, Wisconsin, USA, pp. 56–60.

McGraw, G. 2004. Software security. *IEEE Security & Privacy Magazine*, pp. 80–83.

Meingast, M. et al. 2006. Security and privacy issues with health care information technology. In: *Proceedings of the 28th IEEE Annual International Conference of Engineering in Medicine and Biology Society*. New York, New York, USA, pp. 5453–5458.

Mohammed, N. et al. 2011. Anonymity meets game theory: secure data integration with malicious participants. *The VLDB Journal—The International Journal on Very Large Data Bases* 20(4), pp. 567–588.

Morton, A. and Sasse, M. 2012. Privacy is a process, not a PET: a theory for effective privacy practice. In: *Proceedings of the 2012 workshop on new security paradigms NSPW'12*. Bertinoro, Italy, pp. 87–104.

Nachouki, G. and Quafafou, M. 2011. MashUp web data sources and services based on semantic queries. *Information Systems* 36(2), pp. 151–173.

Pasierb, K. et al. 2011. Privacy-preserving data mining, sharing and publishing. *Journal of Medical Informatics & Technologies* 18, pp. 70–76.

Philip Coppel Qc 2012. The Data Protection Act 1998 & Personal Privacy. *Statute Law Society* 499(19 March 2012), pp. 1 – 31.

Pistoia, M. et al. 2007. When Role Models Have Flaws : Static Validation of Enterprise Security Policies Introduction : RBAC Systems. In: *29th International Conference on Software Engineering*. Minneapolis, MN, USA.

Pon, R. and Critchlow, T. 2005. Performance-oriented privacy-preserving data integration. *Data Integration in the Life Sciences*, pp. 240–256.

Prakash, V. and Darbari, M. 2012. A Review on Security Issues in Distributed Systems. *International Journal of Scientific & Engineering* 3(9), pp. 1–5.

Ray, S. S. et al. 2009. Combining multisource information through functional-annotation-based weighting: gene function prediction in yeast. *IEEE transactions on bio-medical engineering* 56(2), pp. 229–36.

Reeve, A. 2013. Cloud-Based Data Integration Adds Concerns about Latency and Security [Online] Available at: http://data-informed.com/cloud-based-data-integration-adds-concerns-about-latency-and-security/ [Accessed: 4 February 2014].

Ristenpart, T. et al. 2009. Hey, you, get off of my cloud: exploring information leakage in third-party compute clouds. In: *Proceedings of the 16th ACM conference on Computer and communications security*. Chicago, Illinois, USA.

Ross, R. et al. 2014. Systems security Eegineering an integrated approach to building trustworthy resilient systems. *NIST Special Publication (800-160)*, p. 121.

Russom, P. 2008. Data Integration Architecture: What It Does, Where It's Going, and Why You Should Care [Online] Available at: http://tdwi.org/articles/2008/05/27/data-integration-architecture-what-it-does-where-its-going-and-why-you-should-care.aspx.

Saeed, M. Y. et al. 2014. Insight into Security Challenges for Cloud Databases and Data Protection Techniques for Building Trust in Cloud Computing. *Journal of Basic and Applied Scientific Research* 4(1), pp. 54–59.

Takabi, H. et al. 2010. Security and privacy challenges in cloud computing environments. *IEEE Security & Privacy Magazine* (December), pp. 24–31.

Tian, Y. et al. 2011. Dynamic content-based cloud data integration system with privacy and cost concern. In: *Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference on - CEAS '11*. Perth, Western Australia, Australia: ACM Press, pp. 193–199.

Treglia, J. V. and Park, J.S. 2009. Towards trusted intelligence information sharing. In: *Proceedings of the ACM SIGKDD Workshop on CyberSecurity and Intelligence Informatics - CSI-KDD '09*. Paris, France: ACM Press, pp. 45–52.

U.S. HHS 1996. Health Insurance Portability and Accountability Act (HIPAA) [Online] Available at: http://www.hhs.gov/ocr/privacy/hipaa/understanding/summary/.

Watson, D. 2007. Web Application Attacks. *Network Security* (October), pp. 10–14.

Whang, S. and Garcia-Molina, H. 2012. A model for quantifying information leakage. *Secure Data Management*.

Xiong, L. et al. 2007. Preserving data privacy in outsourcing data aggregation services. *ACM Transactions on Internet Technology* 7(3), p. 28.

Yau, S. and Chen, Z. 2008. Security policy integration and conflict reconciliation for collaborations among organizations in ubiquitous computing environments. *Ubiquitous Intelligence and Computing*, pp. 3–19.

Yau, S. S. and Yin, Y. 2008. A Privacy Preserving Repository for Data Integration across Data Sharing Services. *IEEE Transactions on Services Computing* 1(3), pp. 130–140.

Youssef, A. and Alageel, M. 2012. A Framework for Secure Cloud Computing. *International Journal of Computer Science* 9(4), pp. 487–500.

Zhang, D. Y. et al. 2011. Modeling and evaluating information leakage caused by inferences in supply chains. *Computers in Industry* 62(3), pp. 351–363.

AUTHOR INDEX

Proceedings of CLOSER 2015 Doctoral Consortium

5th International Conference on Cloud Computing and Services Science

www.closer.scitevents.org

IN COOPERATION WITH:

acm In-Cooperation

sigmis

DELL