Benefits and challenges of a reference architecture for processing statistical data

Wahyudi, Agung; Matheus, Ricardo; Janssen, Marijn

DOI
10.1007/978-3-319-68557-1_41

Publication date
2017

Document Version
Final published version

Published in
Digital Nations – Smart Cities, Innovation, and Sustainability - 16th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, I3E 2017, Proceedings

Citation (APA)
Wahyudi, A., Matheus, R., & Janssen, M. (2017). Benefits and challenges of a reference architecture for processing statistical data. In *Digital Nations – Smart Cities, Innovation, and Sustainability - 16th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, I3E 2017, Proceedings* (Vol. 10595 LNCS, pp. 462-473). (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Vol. 10595 LNCS). Springer. https://doi.org/10.1007/978-3-319-68557-1_41

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Benefits and Challenges of a Reference Architecture for Processing Statistical Data

Agung Wahyudi(✉), Ricardo Matheus, and Marijn Janssen

Faculty of Technology, Policy and Management, Delft University of Technology, Jaffalaan 5, 2628 BX Delft, The Netherlands
{a.wahyudi,r.matheus,m.f.w.h.a.janssen}@tudelft.nl

Abstract. Organizations are looking for ways to gain advantage of big and open linked data (BOLD) by employing statistics, however, how these benefits can be created is often unclear. A reference architecture (RA) can capitalize experiences and facilitate the gaining of the benefits, but might encounter challenges when trying to gain the benefits of BOLD. The objective of the research to evaluate the benefits and challenges of building IT systems using a RA. We do this by investigating cases of the utilization of a RA for Linked Open Statistical Data (LOSD). Benefits of using the reference architecture include reducing project complexity, avoiding having to "reinvent the wheel", easing the analysis of a (complex) system, preserving knowledge (e.g. proven concepts and practices), mitigating multiple risks by reusing proven building blocks, and providing users a common understanding. Challenges encountered include the need for communication and learning the ins and outs of the RA, missing features, inflexibility to add new instances as well as integrating the RA with existing implementations, and the need for support for the RA from other stakeholders.

Keywords: Reference architecture · Open government · e-Government · Open Data · Big data · BOLD · Statistical data · LOSD · Data processing · Data cube

1 Introduction

Large amounts of data are available due to pervasiveness of data-generation and related technologies such as mobile computing, internet-of-things (IoT), and social media. This all results in big and open linked data (BOLD) in which some data is opened and the linking of data creates value [2].

Todays' massive data have been publicly available by government initiates to open data. The underlying motivations are to create transparency, enable participation and to stimulate innovation [3–7]. The data may represent government's spending, parliament meeting record, as well as Government's IoT such as GPS data from public trains and buses, weather data, and environment data. This extends the existing published statistical data, such as census data, demography data, education data, etc. Moreover, academia, businesses and individuals also start opening their data [8]. Research data, company's supply chain data, crowd-sourced data are examples of publicly available data from non-government parties. Open data refers to datasets that are published under

A.K. Kar et al. (Eds.): I3E 2017, LNCS 10595, pp. 462–473, 2017.
DOI: 10.1007/978-3-319-68557-1_41

an open license, access to and (third-party) use of the datasets is without any restrictions [9]. According to Janssen, et al. [4], the primary goal of open data initiatives is to minimize the constraints on and efforts of reusing data.

Combining a dataset with other datasets is easy if the dataset are published in a structured way and are linked to each another [10]. Data can be sourced from multiple providers, interlinked each other, and retrieved using semantic queries. Linked data principles has been adopted by a growing number of data providers (both public and private) over the years, leading to the development of a global data space (i.e. the Web of Data) that consists of billions of assertions across multiple sectors. According to the statistics provided by LOD stats, the Web of Data contains 149 billion RDF triples from 2973 datasets[1].

The combination of big data, open data and the linking of data results in *linked open statistical data* (LOSD). A number of studies argue that organizations gain various benefits from LOSD, including improving economic growth, creating innovation, assisting to develop new or crafting better products and services [11–13]. The interest using LOSD is considerably growing [14], and a number of new business models for LOSD adoption is introduced [15–17].

The use of LOSD encounters a number of hurdles [18]. Gantz and Reinsel [19] found that even two thirds of businesses across North America and Europe failed to create value from their data. According to LaValle et al. [20], those challenges is not caused by the data only, but also by the IT systems capturing and processing the data, and the people who conduct operation on the data. Data users need to tackle issues such as metadata availability, connectivity between datasets, data quality, data ownership, privacy constraint, interoperability between applications, data standardization, and so on [21].

A reference architecture (RA) which serves as a guide to develop IT system has been developed to support the implementation of LOSD. A RA describes the highest level of abstraction and does not convey the design for an actual system or even a detailed diagram of the interconnection, but rather provides architectural guidance [22]. In this way a RA can support a smoother implementation.

The OpenCube Toolkit (OCT) serves as an instance of a reference architecture of IT system development for processing LOSD. OCT was built upon an underlying data processing lifecycle. Each process in the lifecycle is performed by certain applications. Those involved applications are then built and bundled in an integrated platform, i.e. Information Workbench[2].

A RA can help IT system developers to manage the complexities, and also deliver a number of benefits such as knowledge management, common understanding, risk mitigation, easing the analysis of systems, increasing reusability and connectivity, and reducing errors and mistakes [22, 23]. However, possible drawbacks are overhead projects and stifling creative and innovative solutions to problems [24]. Hence, the experiences with RA provide mixed outcomes.

[1] http://stats.lod2.eu/.

[2] https://github.com/opencube-toolkit/.

The objective of this paper is to evaluate the benefits and challenges of building IT system using a RA. This paper is organized as follows. First, we describe the research background. Thereafter the research approach is presented. This is followed by the presentation of the RA. In Sect. 4, we describe the cases of developing IT system for processing LOSD using the RA. Using the cases, we discussed the benefit and challenges of using an instance of RA (i.e. OCT) that will be covered in Sect. 5. Finally, conclusions are drawn.

2 Research Approach

We aim at investigating the benefits and challenges of building IT system using a RA. First, challenges and benefits of RAs were derive from literature. The findings were then used to investigate cases using OCT for developing LOSD applications (Fig. 1).

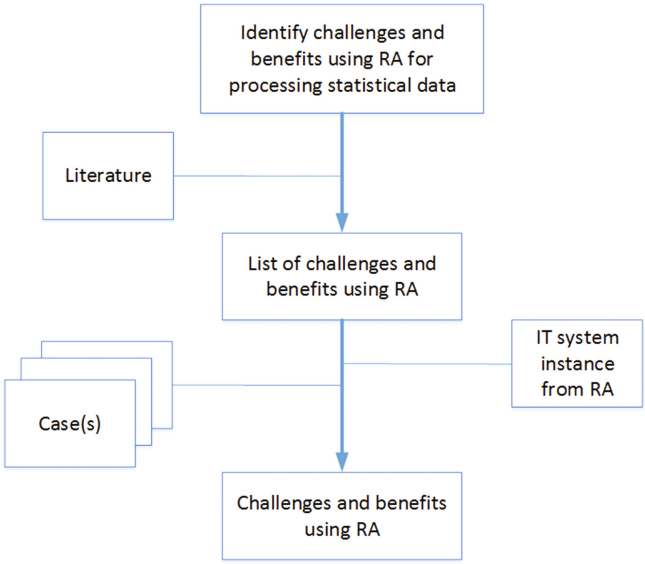

Fig. 1. Research approach in this study

OCT provided by OpenCube Consortium was used as the primary RA. Its use was investigated by analyzing eleven cases from an assignment given to students from Delft University of Technology (TU Delft), The Netherlands. The assignment was to create an IT system for combining LOSD that takes seven weeks to complete. Reports included mistakes, challenges and issues. We conducted content analysis to the groups' reports to identify benefits and challenges of using RA for building IT systems. We identified, coded and analyzed the benefits and challenges using NViVo. They were grouped based on the ICT architecture layers, i.e. business, business process, application, information, and infrastructure.

3 OpenCube Toolkit (OCT) Reference Architecture

The OpenCube Toolkit (OCT) is open source software developed by Open Cube Project[3]. The project aimed at developing software tools that facilitate (a) producing high-quality LOSD and (b) reusing distributed LOSDs in data analytics and visualizations. As a reference OCT takes a data processing lifecycle as the foundation. The OCT projects describe three main processes, i.e. Create, Expand, and Exploit. In the *creation phase*, the data users ingest raw data, pre-process the data, and then convert the data to linked data format in the data cubes forms. Data cube is a way to describe multi-dimensional variables contained in the data. For example, a 4-dimensional data cube may contains income, population, age, and year of observation from a certain country.

Three activities are defined in the *expansion phase*, i.e. (1) Discover and pre-process raw data; (2) Define structure & create cube; and (3) Publish cube. The outcome of this phase is a linked data cube. The cube can be expanded using new data. For this two activities need to be executed; (1) identify compatible cubes and (2) expand cube. Expansion of the cube could be caused by aggregating different cubes to accomplish a certain objective.

The last phase is the *exploitation phase* in which data users process, analyze and visualize the data, communicate the result, and/or make decision from the result. Therefore, three activities are defined in this phase, namely (1) discover and explore cube, (2) analyze cube, and (3) communicate results.

The components of OCT were selected and/or developed based on the proposed data processing lifecycle. There are number of open source components corresponding to certain process. In the *creation phase*, the goal is to transform raw data to linked data so that the proposed RA applications include data converting software such as JSON-stat2qb, Grafter, D2RQ, TARQL, and R2RML. The applications were developed by the members of OCT consortium. Most of them are used in the integrated platform, but some are stand-alone such as Grafter. TARQL creates RDF data cubes from legacy tabular data, such as CSV/TSV files. D2RQ produces RDF data cubes from relational databases. JSON-stat2qb converts JSON-stat files into RDF data cubes. R2RML transforms tabular data to linked data cubes.

The objective in the *expansion phase* is to expand the linked data cube. The corresponding applications proposed in the RA are the OpenCube Compatibility Explorer, OpenCube Aggregator, and OpenCube Expander. Given an initial cube in the RDF store, the main role of the OpenCube Compatibility Explorer is to search into the Linked Data Web and identify cubes that are relevant to expand the initial cube, and create typed links between the local cube and the compatible ones. The role of OpenCube Aggregator is twofold. First, given an initial cube with n dimensions the aggregator creates $(2n - 1)$ new cubes taking into account all the possible combinations of the n dimensions. Second, given an initial cube and a hierarchy of a dimension, the aggregator creates new observations for all the attributes of the hierarchy. OpenCube Expander creates a new expanded cube by merging two compatible cubes.

[3] http://www.opencube-toolkit.eu.

Data users create value from the data in *Exploitation* phase. OCT RA proposes a number of accessing, processing, analytics, visualization applications such as Data Catalogue Management, SPARQL console, OpenCube Browser, DataCube Grid View, Spreadsheet Builder, OpenCube OLAP Browser, R Statistical Analysis, Choropleth Map View, OpenCube Map View, and Interactive Chart Visualization. Data catalogue management serves as user interface (UI) templates for managing metadata on RDF data cubes and supporting search and discovery. OpenCube Browser is a table-based visualization of RDF data cubes. Data users could perform OLAP operations (e.g. pivot, drill-down, and roll-up) on top of multiple linked data cubes using OpenCube OLAP Browser. R statistical analysis enables execution of R data analysis scripts from the OpenCube Toolkit, visualization of results or their integration as RDF triples. Interactive chart serves as visualization widgets, i.e. visualization of the RDF data cube slices with charts. OpenCube MapView is map-based visualizations of linked data cubes with a geo-spatial dimension.

The software building blocks are integrated and bundled in a single platform, namely Information Workbench Community Edition platform. This is an open source application that serves as an architectural backbone of the toolkit. Information Workbench provides the SDK for building customized applications and realizing generic low-level functionalities such as shared data access, logging and monitoring (Fig. 2).

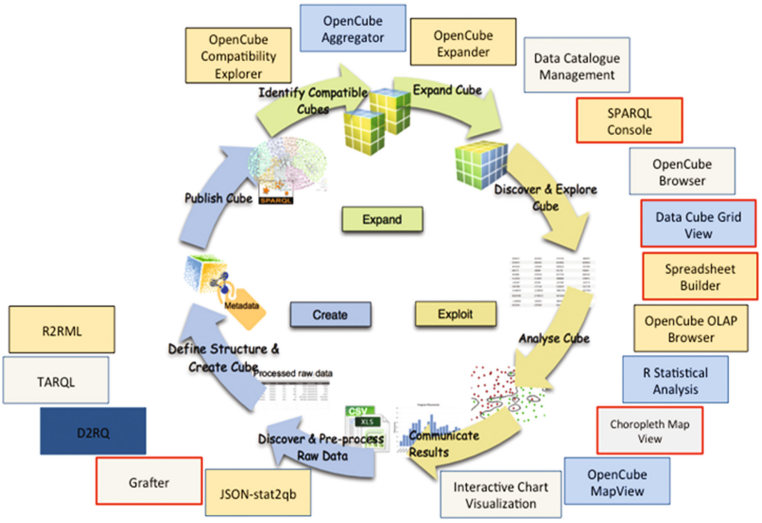

Fig. 2. Open cube toolkit processes and systems components RA [25]

OCT meets the attributes of a RA because (1) it comprises a prescriptive architecture that is built based on data processing lifecycle and includes the corresponding system elements (i.e. applications and infrastructure), and (2) it serves as a guidance for implementations (principles, guidelines, or technical positions).

4 IT Architecture for Processing LOSD Using OCT

Our objective is to investigate the experiences of the use of the RA for building a concrete IT system for processing LOSD. For that purpose, we exploit OCT as a reference architecture for combining LOSD. An assignment solving a business problem using LOSD was given to a number of Master students from Delft University of Technology (TU Delft), The Netherlands. There were eleven cases created by eleven groups that consist of 3–4 persons each, as listed in the Table 1.

Table 1. LOSD use cases from Master students of Delft University of Technology

Group	Project
1	Not-so-funda: A Linked Open Data analysis of house prices and education in Utrecht
2	Location Analysis for the Automotive Industry after the Brexit in the EU: Designing a decision-making process for reallocating assembly plants of Nissan, Toyota and Honda within European Union
3	Matching human capital supply and demand in Europe
4	OpenUN: An architectural design for measuring a Sustainable Development Goal
5	E-Doctor Platform: Healthcare services for integrating immigrants in the Netherlands
6	Linking the data - Where to invest?: A research in a Linked open data architecture on investment regions within the municipality of Amsterdam
7	Amsterdam parking app
8	Raising Awareness About GHGs Emission Among EU Citizens with the Use of Open Data
9	Primary School Recommendation System
10	Attractiveness of countries' living situations
11	The European Gender Inequality Indicators

5 Benefits and Challenges of the Reference Architecture

The benefits and challenges faced by the groups were analyzed. The benefits as found in the literature were used to evaluate the assignment and the results are shown in Table 2. The benefits are categorized using architecture layers [26] as shown in the left column in the table.

In the *business process* layer, the majority of the groups mentioned OCT helped them to reduce project complexity due to the availability of pre-defined data processing lifecycle as part of OCT. They did not need to reinvent the processes but were able to directly fit the processes to their objectives. Some customization of the data processing lifecycle probably took place, but the effort was much less than building the processes from scratch. This finding confirms the benefit mentioned in the literature, i.e. RA is supposed to help IT architects to reduce complexity [22].

In the *application* layer, several groups noted the benefit originating from reusing the building blocks in OCT. The blocks were designed to support the data processing

Table 2. Benefits of using OCT as a reference architecture

Architecture layer	Benefit	Mentioned by group
Overall architecture	• Not having to start from scratch • More efficient development (less time) • Decomposing the complex problem into smaller parts • Providing a common knowledge (and improving understandability)	#2–#4, #7–#9, #11
Business process	• Using the process of data lifecycles	#1,#2, #4, #5–#11
Application	• Use of proven interconnected building blocks • Knowledge transfer of building blocks • Reduce risks of failure	#2, #5, #7, #9
Information	• Variety of involved information is pre-defined as a template • Templates are knowledge repository	#2, #4, #7, #11
Infrastructure	• Effective on implementing the system (hardware and software)	#1–#4, #6, #8, #10, #11

lifecycle. The interrelation (i.e. between the business process and the related applications) eases the architecture's users to understand and breakdown the system. This finding confirm the benefit stated by Gong [23], that a RA should ease the analysis of a (complex) system. The building blocks were also proven to do the specified job and they are interoperable with each other. The groups found the building blocks were very helpful and replicable for the functions they needed to accomplish their objectives. This confirms the findings of Cloutier et al. [22], that a RA should preserve knowledge (e.g. proven concepts and practices) that can be reused and replicated for future projects. Reusing proven building blocks will also reduce failure risk that is a benefit from a RA [22].

In the *information* layer, a number of pre-defined information were found useful for several groups. Using these as templates, they did not need to design types of information to be used, stored, and archived. The templates act as a knowledge repository for the information architects.

Most of the groups found that OCT helped them to execute the systems implementation project better. Using the hardware and software components that are proven to work and interoperate, the implementation project became effective which means the amount of available resources such as investment and labor were properly utilized. Consequently, risk from the architecture project such as delay and the resulting overrun project cost could be properly mitigated, as Cloutier et al. [22] mentions.

As illustrated in the OCT case, a RA provides IT architects the common language to speak about the business process and the corresponding applications, information, and infrastructure. For example, OCT users interpret the meaning of expand process as the updating process for any current data cubes with a recent corresponding incoming data, not other definitions. This confirms common understanding advantage from using a RA as described by Cloutier et al. [22].

We also identified a number of challenges from the groups' report. Those challenges create hurdles and impediments of using the RA. We listed the identified challenges in Table 3.

Table 3. Challenges of using OCT as a reference architecture

Architecture layer	Challenge	Mentioned by group
Overall architecture	–	–
Business process	1. Using building blocks from OCT is not straightforward due to lack of documentation (e.g. uploading CSV files, converting CSV to RDF) 2. It's not clear how to create data pipelines in OCT (i.e. placing output of a building block as inputs of the others) 3. No clue how to automate the process (e.g. processing streaming of data, visualizing real-time output) 4. OCT does not provide assessment of data quality support 5. Lack of community involvement	1. #1–#11 2. #2, #3, #6, #11 3. #3, #10 4. #1, #2 5. #10
Application	1. Users find it difficult to use the menu and interface in the Information Workbench because they are not intuitive 2. Certain dependencies are required (e.g. Oracle Java 8); OCT does not work with updated version of the dependencies 3. Very often applications outside OCT are utilized due to OCT limitation (e.g. OpenRefine, Google Fusion) 4. Data visualization using OCT is challenging because the installed R packages are limited by default while OCT users are impossible to install packages 5. Only support R for visualization; Difficult to connect other visualization applications to OCT	6. #2,#5,#7, #10 7. #3, #4, #7, #10 8. #2, #4, #9 9. #1, #2, #6, #8, #11 10. #7, #11
Information	1. OCT does not provide mechanism to store the data in the different machine (e.g. data center, data lake) from the one where OCT is installed 2. Which linked data vocabularies that OCT supports is not documented clearly 3. SPARQL queries is challenging to use 4. Since linked data is not human-readable, it's difficult to understand the benefit	11. #1, #3, #10 12. #3, #7, #8, #11 13. #2 14. #1, #5
Infrastructure	1. OCT could be installed only in Unix-based environment 2. No clue how to implement OCT in a cluster of computers	15. #2, #6, #8 16. #10, #11

In the *business process* layer, all groups reported that understanding the RA was somewhat difficult due to a lack of documentation. This hindered them to use the OCT better. After effortful try-and-error activities that stuck the progress, many of them finally used other applications beyond OCT, such as OpenRefine, Perl, R, Python, awk,

Tableau, etc. They have gone a number of unsuccessful trials of building their IT system using the menu in the Information Workbench. There was also no guideline how to automate the process, such as scheduling of retrieving raw data from the data sources, processing streaming of data, or visualizing real-time data. Some groups also noted that data quality was difficult to be assessed using the Information Workbench. Incorporating multiple datasets mean that the data users should take variety of data quality into account. Therefore, some additional applications beyond OCT were used to assess and improve data quality. The use of OCT was also difficult because there was very few example of successful OCT implementation. We hardly found community involvement for OCT improvement such as forum, user groups, mailing lists, etc.

In the *application* layer, the groups found it's difficult to use the menu and interface in the Information Workbench because they are too simple and not intuitive enough. Dependencies of OCT applications were also too rigid, for example OCT works only with Oracle Java 8. Very often applications outside OCT are utilized due to OCT limitation (e.g. Open-Refine, Google Fusion). Data visualization using OCT is challenging because the installed R packages are limited by default while OCT users are impossible to install packages. Only support R for visualization; Difficult to connect other visualization applications to OCT.

There are also a number of challenges found in the *information* layer. First, OCT does not provide mechanism to export and store the data to other machines (e.g. data center or data lake). Second, which linked data vocabularies that OCT supports is not documented clearly. Currently there are many varieties of linked data vocabularies with which data creators could confuse. Third, SPARQL syntax is quite different from standard SQL/PL. Some groups found it's quite challenging to understand and use SPARQL. Fourth, since linked data is not human-readable, it's difficult to understand the benefit. Some groups questioned the need to convert the raw data to linked data. They preferred to exploit the raw data directly without having spent additional effort to publish linked data.

The groups mentioned several challenges in the *infrastructure* layer such as OCT could be installed only in Unix-based environment and no clue how to implement OCT in a cluster of computers. As the data size and number of users grows, the most common approach is to deploy a cluster of regular hardware. Building an OCT instance in a parallel environment was not described in the documentation and currently OCT does not support cluster implementation.

From the OCT cases, we derived challenges coping with a RA in general. First, proper documentation is needed to fully exploit the RA. It means that a RA needs the optimum amount of documentation. Too few guidelines will cause the RA difficult to concretize and implement. Issues mentioned in the cases, i.e. difficult to use the RA components and confusing what standards to be followed (e.g. LOD vocabularies) reflect the consequences of lack of documentation. Proper documentation is also required to introduce new or unpopular technologies adopted by the RA, for example linked data principles and SPARQL syntax in our cases. On the other hand, too much information in the documentation will lead the high level users such as business managers and customers troublesome to get the helicopter view.

Second challenge is that missing important features will make the RA irrelevant. Those important features should exist in every RA because they constitute the

functionalities a RA must have. We noted several missing important features from OCT cases, i.e.: (1) process automation that is mandatory for a RA in data processing; (2) intuitive and sufficient user interface that strictly important for helping the users to master the RA; (3) proper authority that ensures the user to fit the tools with the jobs (e.g. users unable to install R packages in the R statistical analysis in OCT, meanwhile the packages are required to accomplish the data objective).

The need for proper documentation for full exploitation of RA, missing important features from a RA that makes it irrelevant, inflexibility to add a new instance as well as integrate it in existing implementation, and RA still island without future support and collaboration among stakeholders.

Every user has different data objectives with different kind of problems (e.g. issues with data quality, privacy, etc.), initial conditions (e.g. having legacy system), and constraints (e.g. budget, time, management approval, etc.). Consequently, there should be many customizations in implementation of a RA. Systems customization could be also resulted due to adoption of emerging technologies, such as cloud computing, parallel processing, in-memory analytics, etc. Therefore, a RA should be flexible to add a new instance (e.g. a process, application, information, or an infrastructure component) as well as to integrate the instance in existing implementation. From our cases, some groups require features beyond OCT capability such as data quality assessment, data wrangling, web service, storing the data in a location besides OCT machine, and implementing in a cluster. As we observed, these available features from OCT were not feasible to perform the task. Although the features could be deployed in the machine where OCT resides, but integrating it within OCT environment was troublesome.

The last challenge is that OCT is still a stand-alone without future support and collaboration between users and developers, among users, and among developers for massive use. The collaboration is stimulated and incubated in an ecosystem. Good collaboration will result in proven components of RA, richness of RA implementation cases, and crowd-solutions for many architectural problems. From our cases, after the groups found the documentation of OCT was not helpful, they tried to search relevant cases and find the answers for their questions in the Internet. However, those were neither useful because useful knowledge was hardly available on the internet.

6 Conclusion

The objective of the research presented in this paper is to evaluate the benefits and challenges of using a reference architecture for building IT systems. The OpenCube Toolkit was used as a reference architecture for developing Linked Open Statistical Data applications. We investigated the experiences by observing the development in eleven cases. A range of benefits using OCT as a reference architecture were identified. The RA helps to (1) reduce project complexity and need not “reinvent the wheel”, (2) eases the analysis of a (complex) system, (3) preserves knowledge (e.g. proven concepts and practices) that can be reused and replicated for future projects, (4) mitigates multiple risks such as failure risk, delay and the resulting overrun project cost by reusing proven building blocks, and (5) provides common understanding.

Implementing IT system using OCT seems to be initially straightforward, but in a reality a number of challenges needs to cope with, i.e. (1) the need for proper documentation for full exploitation of RA, (2) missing important features from a RA that makes it irrelevant, (3) inflexibility to add a new instance as well as integrate it in existing implementation, and (4) RA is a blueprint that could only be widely used with support and collaboration among stakeholders. Although generalization of the results is difficult, our findings suggest when developing a RA the users should have clear guidelines on how to use the RA and what the limitations of its use are.

Acknowledgement. Part of this work is funded by the European Commission within the H2020 Programme in the context of the project OpenGovIntelligence (www.opengovintelligence.eu) under grant agreement No. 693849.

We would like to thank PT. Telekomunikasi Indonesia, Tbk. for their support during the study as well as the students who participated in the practical work.

References

1. Mayer-Schönberger, V., Cukier, K.: Big Data: A Revolution That Will Transform How We Live, Work, and Think. Houghton Mifflin Harcourt, Boston (2013)
2. Janssen, M., Kuk, G.: Big and Open Linked Data (BOLD) in research, policy, and practice. J. Organ. Comput. Electron. Commer. **26**(1–2), 3–13 (2016)
3. Charalabidis, Y., Psarras, J.: Combination of interoperability registries with process and data management tools for governmental services transformation. In: Hawaii International Conference on System Sciences (HICSS-42), pp. 5–8, January 2009
4. Janssen, M., Charalabidis, Y., Zuiderwijk, A.: Benefits, adoption barriers and myths of open data and open government. Inf. Syst. Manag. **29**(4), 258–268 (2012)
5. Zhang, J., Dawes, S.S., Sarkis, J.: Exploring stakeholders' expectations of the benefits and barriers of e-government knowledge sharing. J. Enterp. Inf. Manag. **18**(5), 548–567 (2005)
6. Zuiderwijk, A., Jeffery, K., Janssen, M.: The potential of metadata for linked open data and its value for users and publishers. J. eDemocracy **4**(2), 222–244 (2012)
7. Manyika, J., Chui, M., Groves, P., Farrell, D., Van Kuiken, S., Doshi, E.A.: Open data: unlocking innovation and performance with liquid information. McKinsey&Company (2013)
8. Zuiderwijk, A., Janssen, M., van de Kaa, G., Poulis, K.: The wicked problem of commercial value creation in open data ecosystems: policy guidelines for governments. Inf. Polity **21**(3), 223–236 (2016)
9. Open Knowledge Foundation: Open Data Handbook Documentation (Release 1.0.0) (2012)
10. Heath, T., Bizer, C.: Linked Data: Evolving the Web into a Global Data Space, 1st edn., vol. 1, no. 1. (2011)
11. Janssen, K.: The influence of the PSI directive on open government data: an overview of recent developments. Gov. Inf. Q. **28**(4), 446–456 (2011)
12. Lundqvist, B.: Digital agenda: turning government data into gold: the regulation of public sector information–some comments on the compass-case (2012). SSRN 2148949
13. Martin, C.: Barriers to the open government data agenda: taking a multi-level perspective. Policy Internet **6**(3), 217–240 (2014)
14. GovLab: Open Data 500 US (2015). http://www.opendata500.com/us/

15. Ferro, E., Osella, M.: Eight business model archetypes for PSI Re-Use. In: Open Data on the Web Workshop, Google Campus, Shoreditch, London (2013)
16. Janssen, M., Zuiderwijk, A.: Infomediary business models for connecting open data providers and users. Soc. Sci. Comput. Rev. **32**(5), 694–711 (2014)
17. Magalhaes, G., Roseira, C., Manley, L.: Business models for open government data. In: Proceedings of the 8th International Conference on Theory and Practice of Electronic Governance, pp. 365–370 (2014)
18. Zuiderwijk, A., Janssen, M., Choenni, S., Meijer, R., Alibaks, R.S.: Socio-technical impediments of open data. Electron. J. e-Gov. **10**(2), 156–172 (2012)
19. Gantz, J., Reinsel, D.: Extracting value from Chaos State of the universe: an executive summary. IDC iView, pp. 1–12, June 2011
20. LaValle, S., Lesser, E., Shockley, R., Hopkins, M.S., Kruschwitz, N.: Big data, analytics and the path from insights to value. MIT Sloan Manag. Rev. **21** (2013)
21. Janssen, M., Estevez, E., Janowski, T.: Interoperability in Big, Open, and Linked Data—Organizational Maturity, Capabilities, and Data Portfolios. IEEE Computer Society (2014)
22. Cloutier, R., Muller, G., Verma, D., Nilchiani, R., Hole, E., Bone, M.: The concept of reference architectures. Syst. Eng. **13**(1), 14–27 (2010)
23. Gong, Y.: Engineering Flexible and Agile Services: A Reference Architecture for Administrative Processes. TU Delft, Delft University of Technology (2012)
24. Windley, P.J.: Digital Identity. O'Reilly Media, Inc., Sebastopol (2005)
25. Kalampokis, E., Roberts, B., Karamanou, A., Tambouris, E., Tarabanis, K.: Challenges on developing tools for exploiting linked open data cubes. In: CEUR Workshop Proceedings, vol. 1551 (2015)
26. Lankhorst, M.: Enterprise Architecture at Work: Modelling, Communication and Analysis. The Enterprise Engineering Series (2009)
27. Janssen, M.: Framing enterprise architecture: a meta-framework for analyzing architectural efforts in organizations. In: Coherency Management: Using Enterprise Architecture for Alignment, Agility, and Assurance, pp. 99–118 (2009)