

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/301641987>

Establishing Distributed Governance Infrastructures for Enacting Cross-Organization Collaborations

Chapter · January 2016

DOI: 10.1007/978-3-662-50539-7_3

CITATIONS

4

READS

193

1 author:



Alex Norta

Tallinn University of Technology

73 PUBLICATIONS 397 CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:



Collaboration enhancing sustainable conceptual model development and implementation for the SME-s in machinery domain [View project](#)



Authcoin [View project](#)

All content following this page was uploaded by [Alex Norta](#) on 19 July 2016.

The user has requested enhancement of the downloaded file.

Establishing Distributed Governance Infrastructures for Enacting Cross-Organization Collaborations

Alex Norta

Department of Informatics, Tallinn University of Technology,
Akadeemia tee 15A, 12816, Tallinn, Estonia
alex.norta.phd@ieee.org

Abstract. The emergence of blockchain 2.0 technology enables novel agile business networking collaborations for decentralized autonomous organizations (DAO). Smart contracts are enactable by service-oriented cloud computing and blockchain technology for governing DAOs that engage in business collaborations. The distributed governance infrastructure (DGI) for such governance involves an ecosystem of agents, policies, services, and so on. To the best of our knowledge, a formal investigation of the lifecycle for establishing such a DGI has not been explored so far. This paper fills the gap by establishing a formalized DGI intended for the enactment of smart contracts for electronic communities of DAOs. The evaluation of the DGI-lifecycle is performed by means of model checking and discussing what pre-existing systems and also by means of discussing what pre-existing solutions exist for an application-system implementation.

Keywords: decentralized autonomous organizations, e-governance, smart contract, open cloud ecosystem, service orientation

1 Introduction

Decentralized autonomous organizations (DAO)[4] engage in the formation of e-communities that are governed by smart contracts [32]. The latter are a computerized transaction protocol [31] for the execution of contract terms. Blockchain technology [14, 26] is suitable for achieving non-repudiation and fact-tracking of consensual agreements. By means of cryptographic digests resulting in hash value [24], blockchains are a distributed database for independently verifying the ownership of artefacts. Additionally, service-oriented cloud computing (SOCC) [35] accelerates the seamless, ad-hoc integration and coordination of information- and business-process flow [7] to orchestrate and choreograph [19] heterogeneous legacy-system infrastructures that are involved in DAO eCommunities for business collaborations.

While research results emerge for cross-organizational business collaboration [7, 19], a gap exists with respect to a formalized exploration of setting up

distributed governance infrastructure (DGI) for subsequently enacting DAO-collaboration lifecycles. This paper fills the gap by investigating the research question how to set up in a dependable way a decentralized governance infrastructure for enacting cross-organizational business collaborations? In this context, dependable [3] means the components that are part of the setup lifecycle are relied upon to perform exclusively and correctly the system task(s) under defined operational and environmental conditions over a defined period of time. Based on this main research question, we deduce the following sub-questions to establish a separation of concerns. What collaboration concepts are used in the design approach? What is the lifecycle for decentralized governance infrastructure establishment? What are the formal system properties that guide an application implementation? Note that the assumption of the collaboration enactment is derived from the smart contract that comprises the machine-readable code for dynamic DGI-establishment.

The remainder of the paper is structured as follows. Section 2 provides additional information relevant for understanding the business-collaboration context with respect to smart contracting [18, 21] and the embedded cross-organizational collaboration model [7]. Section 3 gives the design approach together with the used collaboration concepts to answer the first sub-question. Section 4 shows the formalized lifecycle for establishing a DGI that answers the second sub-question. Section 5 answers the third sub-question and lists the results from model checking that yield deeper insight for implementing a sound DGI-establishment lifecycle with transactionality provisions. Section 6 discusses related work and finally, Section 7 concludes this manuscript by summarizing the research work, giving the contributions achieved and showing directions for future work.

2 Conceptual Collaboration Context

For comprehending the DGI-establishment lifecycle in the sequel, the following frameworks are relevant. As contracts are the foundation of business collaboration, we show in Section 2.1 concepts and properties for smart contracting and give a corresponding peer-to-peer (P2P) collaboration model of DAOs in Section 2.2. Finally, Section 2.3 conceptually depicts a setup-lifecycle for a smart contract that is the precursor for the DGI-establishment lifecycle.

2.1 Smart contract

The depiction in Figure 1(a) shows the top-level structure of the smart contracting language eSourcing Markup Language (eSML) for which citation [21] gives full details and examples. The bold typed eSML-definitions extend and modify the Electronic Contracting Markup Language (ECML) [2] foundation.

The structure of a smart contract uses the interrogatives *Who* for defining the contracting parties together with their resources and data-use, *Where* to specify the business- and legal context, and *What* for define exchanged business values. Consensus establishment assumes the *What*-interrogative comprises

process views that are cross-organizational matched We refer to [21] for more information about the smart-contracting ontology.

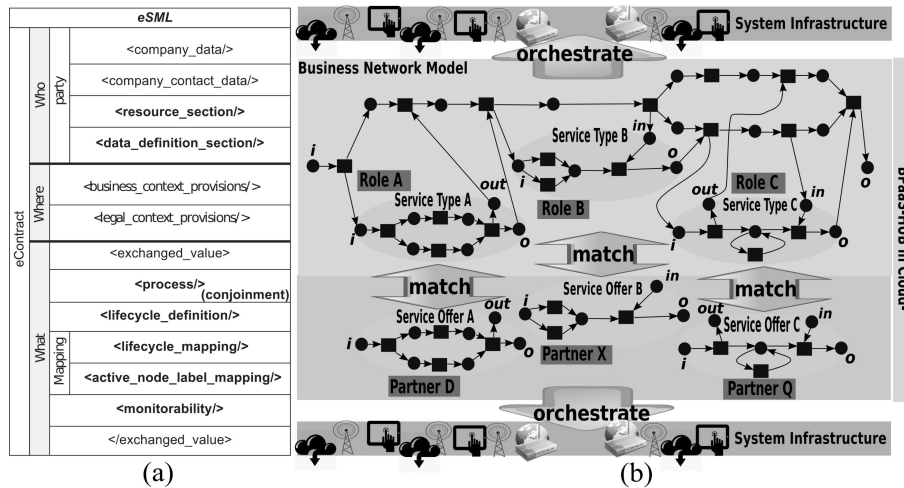


Fig. 1. P2P-collaboration using the eSourcing framework with (a) showing a smart-contracting template [21] and (b) showing a corresponding collaboration model [15].

2.2 P2P-collaboration model

The depiction in Figure 1(b) conceptually shows a DAO collaboration configuration such as a cluster of small- and medium sized enterprises in a cluster for automotive production. The blueprint for forming an eCommunity is a business-network model (BNM) [29]. The latter specifies choreographies for a business scenario and additionally contains template contracts that are service types with assigned roles. The BNMs are located in a collaboration hub for business processes as a service (BPaaS-HUB) [20] that are subset process views [7]. The process views enable a fast and semi-automatical discovery of collaboration parties for learning about their identity, services, and reputation. On the external layer of Figure 1(b), service offers match with service types contained in the BNM. A collaborating partner must also match [7] with the partner roles of a respective service type.

2.3 Conceptual setup top-level

The lifecycle of Figure 2 commences with breeding collaboration inceptions that produce BNMs comprising service types and roles. The BNMs that emerge from the breeding ecosystem exist permanently for repeated use in the subsequent

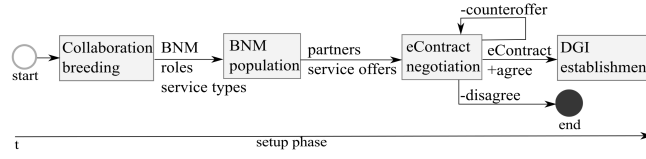


Fig. 2. Conceptual contracting setup-lifecycle for eCommunity establishment.

populating stage. The validation of BNMs matches the available inserted service offers of potential collaboration partners against service types.

The *populate*-phase in Figure 2, yields a proto-contract for a *negotiate* step that involves the collaborating partners. The negotiation phase has three different outcome options. An agreement of all partners establishes the eContract for subsequent rollout of a distributed governance infrastructure; a counter-offer from a partner that results in a new contract negotiation; finally, a disagreement of a partner results in a complete termination of the setup phase. Note that the setup-lifecycle is formalized and we refer the reader to [18] for further details.

3 Design Approach and Used Concepts

The DAO-collaboration lifecycle¹, we formalise with Coloured Petri Nets (CPN) [10], a language for design, specification, simulation and verification of systems. CPN has a graphical representation with a set of modules, each containing a network of places, transitions, arcs and tokens. The modules interact through well-defined interfaces. We use CPN Tools² for designing, simulating, performance testing and verifying the models in this paper.

In Table 1, the data elements of the overall DAO-lifecycle are declared for all refinement levels and they correspond to token colours [17] in the CPN-models below. The table shows hierarchic module-refinement availability mentioned in the left column (1 for the top level and 6 for the most detailed refinement). Token colours are present for all lower but not for any higher CPN-refinement-hierarchy levels. The fourth column explains the purpose of a token colour for a lifecycle. The integer-type tokens mostly represent an identification number and string-type tokens are either eContract negotiation outcomes, or eContract proposals. Boolean-type tokens represent decision points in the lifecycle.

4 Decentralized Governance Infrastructure Establishment

The DGI-establishment models below comprises of three parts. First, Section 4.1 addresses how the eContract is distributed to respective collaborating parties. As a model-refinement elaboration of the governance distribution, Section

¹ Full CPN-model: <http://tinyurl.com/ofae8gn>

² <http://cpntools.org/>

level	service	data property	description	type
1	eCommunity lifecycle	sO	service offer that fits a service type	integer
		sOs	service offer source for communication channel establishment	
		sOt	service offer target for communication channel establishment	
		pA	partner of an eCommunity	
		rO	role a partner can fill	
		eC	eCommunity identification	
		eCo	eContract based on which partners of an eCommunity transact	boolean
		n,r,k,p,l,q,s	counter variables	
		assigned	service offer assigned to a service type	string
		processed	partner prepared for eContract counteroffer re-distribution	
		decision	for negotiated contract proposal (agree disagree counter)	
		outcome	like decision, but input for eCommunity continuation or termination	
2	create	bNM	business network model that get populated with service types and roles	integer
		m	counter variable	
3	populate	sT	service type that populates a bNM	integer
		ch	channel of communication between services	
4	interoperability checking	rOt	role source for communication channel establishment	integer
		rOs	role target for communication channel establishment	
4	contract extraction	spec	specification of extracted eContract	string
4	agreement finalizing	result	whether all eCommunity partners agree on an eContract proposal or not	boolean
		distributed	contract distributed to partner	
4	disagreeing	z	counter variable	integer
		eCo_new	new eContract from a counteroffer to be negotiated	
2	perform	bnma	business network model agent	integer
		sE_l	local service of a respective eCommunity member	
		mO	monitor for observing policy adherence of eCommunity partners	
		sE	electronic service that is enacted	
		lp	local policy extracted from a local contract copy	
		lpnr	counter of local policies	
		s,x	counter variables	
		IC	local contract for respective eCommunity partners extracted from the eContract that coordinates the first	
		insert	service inserting to local contract	
		extracted	instances of contract copies for negotiation	
5	governance distribution	errorID	error identity	integer
		error	error for synchronizing main contract and local copies	
5	prepare	eP	published endpoint for allowing services to communicate	integer
6	preparation error	prepErr	preparation error in the context of assigning an electronic service to a service offer	integer
		assignErr	assignment error in the actual assignment of an electronic service to a service offer	
4	operate	sEr	service error related to concrete electronic service, e.g., deadlock	integer
		tc	termination criteria, either full for eCommunity or partial for disruptive partner change that rolls back to a negotiation stage	

Table 1. Data properties of the DGI-establishment lifecycle [17].

4.2 shows how policies are extracted from the local eContracts. Finally, Section 4.3 focuses on equipping the DGI with a technical enactment foundation. The sections present on the one hand CPN-models and on the other hand, discusses what pre-existing research and technology exists for an application system implementation.

4.1 Governance distribution

The start of this module depicted in Figure 3, constitutes choosing an agreed upon eContract that first leads to the distribution of local contract copies and monitors for respective eCommunity-partners. From every local contract follows an extraction of local policies to which an eCommunity partner must adhere to.

The assigned monitors observe if policy violations occur. Every local contract has a business network model agent (BNMA) attached that utilizes the monitors to see if eCommunity-partners adhere to local policies and also policies for behaviour-control of the entire eCommunity. Before the governance distribution completes, an error may be thrown if a synchronization check between the eContract and the local contracts fail.

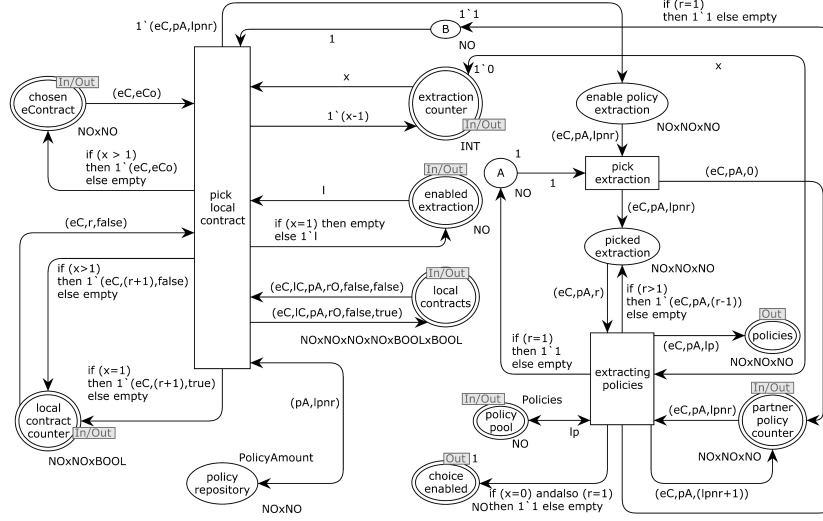


Fig. 4. Extracting from local contract copies a set of required policies (*extraction*) [17].

[1] is the eXchangable Markup Language XML [22] in the form of generated business-rules documents.

4.3 Service-level preparation

The technical realization of a DGI involves the preparation of concrete local electronic services that fill the respective service offers. The model in Figure 5 catches and reports exceptions during service-level preparation and service assignment. For each electronic service, the *prepare-service* depicted in Figure 5 shows the establishment of communication endpoints precede a final check for the operability of services.

With respect to communication endpoints, in [13] a new programming model supports a compositionality of choreographies based on partial choreographies. The latter mix global descriptions such as on a BNM level with communications among collaborating parties. The essential element for this compositionality of choreographies are process views that are the communication endpoints. Accordingly, citation [7] shows that process views reveal only public, relevant parts of services to partner organizations as abstractions of internal services that are private business process based on different projection relations between them. Several matching relations between the respective process views ensure no structural conflicts such as deadlocks, arise with the internal services that comprise additional activities. For using legacy systems to be part of a larger service-based application system, citation [5] proposes loosely coupled services facilitate the establishment of highly dynamic business relations with means of service-oriented computing [6, 25].

During the preparation of an established DGI, errors may occur that the sub-component named *preparation error* of Figure 5 catches. Error options are

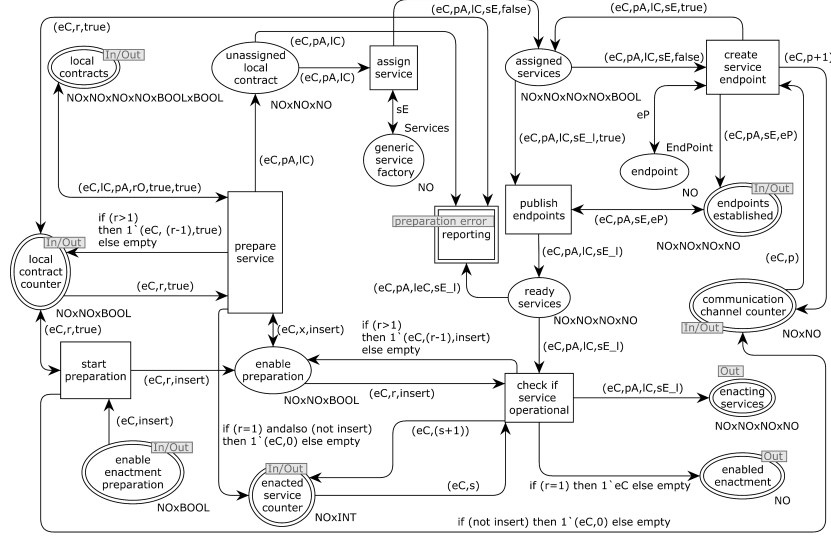


Fig. 5. Electronic service choosing and communication endpoint creation (*prepare*) [17].

for the technical preparation of an electronic service, a failure of soundness check [34] and an assignment error between an electronic service and a local contract copy. Further refinements of Figure 5 are future work.

Next, we discuss the enactment the specifics of an established DGI.

5 Model-Checking Results

We used CPN Tools [10] for correctness and performance checking, especially on aspects relevant for system developers: reachability of CPN-modules end states in manual, or fully automated simulation token games as state explosion means full computational verification is challenging for this size of models; detection of loops as a potential source of livelocks that prevent desired termination reachability; loops require specific attention with respect to effectiveness of exit conditions, such as elements of business-level policy control; performance peaks during runtime either for the design of sufficient resources or for restricting the load with business-level policy control; full system utilisation for ensuring that each part of the modelled system actually is used in some scenario; and consistent termination, i.e., consistent home markings that ensure simple testing of a real system.

The model-checking results in Table 2 focus on CPN-modules where the generated state-space is computationally feasible to verify. Loops exist in the enact-module, while not in remaining modules. Performance peaks in Table 2 represent places in the startup-lifecycle that are potential performance bottlenecks. Peaks exist in all listed modules and we give in the corresponding column the labels of occurrence transitions. While no module listed in Table 2 has any home marking, the model-checking results for dead markings show they are all

module	module property				
	loops	performance peaks	liveness	home marking	dead marking
governance distribution and extraction	no	extraction of local policies	D*/NL	no	multiple
prepare	no	assign service, create service endpoint, publish endpoint, check if service operational	D*/NL	no	multiple
enact	yes	enact service, report start error	D*/NL	no	multiple

Table 2. Model-checking results for the DGI-establishment lifecycle [17].

multiple. D^* means the model-checking results show the dead markings result from intentional disabling of marking paths for the purpose of focusing in specific marking paths under investigation. The latter means for practitioners the testing of implementations is more demanding as many test cases are required. Finally, pertaining to utilisation tests, Table 2 shows no unused subsets exist in the models. We refer to [17] for full details about the model-checking results.

6 Related Work

It is necessary to cross-organizationally establish collaboration frameworks in a way that does not force companies into disclosing an undesirable amount of business internals [7]. Different research efforts address this issue. In [9], the authors investigate tool support for cross-organizational collaboration design. Similarly in [30], research results present an integrated specification language and a user interface for collaborating government organizations to specify events of common interest, policies, constraints and regulations in the form of different types of knowledge rules, manual and automated services, and sharable workflow processes.

In [16], a framework facilitates the understanding of major cross-organizational collaboration challenges. For example, supporting process-level collaboration, and protection of shared IP and data with various enterprise-level and regulatory policies, including flexible and policy-aware process collaboration among people from different enterprises. Also [23] points out the need for agility of business operations in a collaborative services ecosystem of partners and providers by proposing Work-as-a-Service that a collaboration hub facilitates. A cross-organizational architecture in [8] specifies the features and their composition at a higher level that abstracts the internal implementation mechanisms of the organizations involved.

In [11], the author proposes a very general mathematical model for a governance system of large and heterogeneous distributed systems that assumes the use of policies and so-called law-governed interaction protocols. However, the model does not allow for behaviour simulation comparable to the CPN models in this paper and also the refinement level makes it more challenging to assess a technical feasibility study.

e-Government is a focus in [33] that states computing clouds may lead to considerable cost savings. Still, the research work is not as generally applicable

independent of e-Government as the models in this paper that are also suitable for cyberphysical system governance.

Finally, citation [27] discusses the advantages and disadvantages of using distributed governance systems for the internet of things. The authors show that distributed governance poses additional security- and privacy challenges. The assumption in this paper of using process views that hide business secrets internally in larger local services, solves many security- and privacy challenges.

7 Conclusion

The focus of the paper is establishing a decentralized governance infrastructure for enacting cross-organizational business-process aware collaborations. For that, the notion of smart contracts and an affiliated collaboration model is the foundation for deducing a collaboration lifecycle. Assuming an eContract already exists, the lifecycle for establishing and enacting a decentralized autonomous organization we present in combination with additional pre-existing literature discussions that underlines the feasibility of the approach. Finally, the model-checking results allow for a deeper collaboration-system understanding that supports an application-system implementation.

For the lifecycle itself, we choose a formal approach using Coloured Petri Nets that has a graphical notation and also tool support for design, simulation and model checking. We also list the concepts that are embedded in specific parts of the collaboration lifecycle. The latter for decentralized governance infrastructure establishment commences with copying the agreed upon eContract to each respective collaborating party. Next, from each local eContract copy a set of local policies is extracted and monitors and business-network-model agents are assigned to each party. A configuration of local services together with their communication endpoints follows in the lifecycle. When checking the models, the considered properties are loops, performance peaks, liveness, home- and dead markings as they reveal valuable insight for application-system realizations.

For future work, we plan to apply the lifecycle for establishing distributed governance infrastructures in projects for cyberphysical system governance. Additionally, we explore blockchain technology for realizing non-repudiation in process-aware collaboration missions. Furthermore, blockchain technology also promises to enable novel approaches for an effective management of trust, reputation, privacy and security in cross-organizational cyberphysical system collaborations.

References

1. S. Ali, B. Soh, and J. Lai. Rule extraction methodology by using xml for business rules documentation. In *Industrial Informatics, 2005. INDIN '05. 2005 3rd IEEE International Conference on*, pages 357–361, Aug 2005.
2. S. Angelov. *Foundations of B2B Electronic Contracting*. Dissertation, Technology University Eindhoven, Faculty of Technology Management, Information Systems Department, 2006.

3. A. Avizienis, J.C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *Dependable and Secure Computing, IEEE Transactions on*, 1(1):11–33, 2004.
4. V. Butterin. A next-generation smart contract and decentralized application platform, 2014.
5. . Di Nitto, C. Ghezzi, A. Metzger, M. Papazoglou, and K. Pohl. A journey to highly dynamic, self-adaptive service-based applications. *Automated Software Engineering*, 15(3-4):313–341, 2008.
6. Hicker G. Huemer C. Erven, H. and M. Zaptletal. The web services-businessactivity-initiator (ws-ba-i) protocol: an extension to the web services-businessactivity specification. In *In 2007 IEEE International Conference on Web Services (ICWS 2007)*, page 216–224, 2007.
7. R. Eshuis, A. Norta, O. Kopp, and E. Pitkanen. Service outsourcing with process views. *IEEE Transactions on Services Computing*, 99(PrePrints):1, 2013.
8. C. Hahn, J. Recker, and J. Mendling. An exploratory study of it-enabled collaborative process modeling. In M. zur Muehlen and J. Su, editors, *Business Process Management Workshops*, volume 66 of *Lecture Notes in Business Information Processing*, pages 61–72. Springer Berlin Heidelberg, 2011.
9. Christopher Hahn, Jan Recker, and Jan Mendling. An exploratory study of it-enabled collaborative process modeling. In M. zur Muehlen and J. Su, editors, *Business Process Management Workshops*, volume 66 of *Lecture Notes in Business Information Processing*, pages 61–72. Springer Berlin Heidelberg, 2011.
10. K. Jensen, L. Michael, K.L. Wells, K. Jensen, and L.M. Kristensen. Coloured petri nets and cpn tools for modelling and validation of concurrent systems. In *International Journal on Software Tools for Technology Transfer*, page 2007, 2007.
11. N. Minsky. Decentralized governance of distributed systems via interaction control. In A. Artikis, R. Craven, C. Kesim, B. Nihan, Babak. Sadighi, and K. Stathis, editors, *Logic Programs, Norms and Action*, volume 7360 of *Lecture Notes in Computer Science*, pages 374–400. Springer Berlin Heidelberg, 2012.
12. C. Molina-Jimenez, S. Shrivastava, and M. Strano. A model for checking contractual compliance of business interactions. *Services Computing, IEEE Transactions on*, 5(2):276–289, April 2012.
13. F. Montesi and N. Yoshida. Compositional choreographies. In P.R. D’Argenio and H. Melgratti, editors, *CONCUR 2013–Concurrency Theory*, volume 8052 of *Lecture Notes in Computer Science*, pages 425–439. Springer Berlin Heidelberg, 2013.
14. S. Nakamoto. Bitcoin: A peer-to-peer electronic cash system. *Consulted*, 1(2012):28, 2008.
15. N.C. Narendra, A. Norta, M. Mahunnah, and F. Maggi. Modelling sound conflict management for virtual-enterprise collaboration. In *Services Computing (SCC), 2014 IEEE International Conference on*, pages 813–820, June 2014.
16. H.R.M. Nezhad, C. Bartolini, J. Erbes, and S. Graupner. A process- and policy-aware cross enterprise collaboration framework for multisourced services. In *SRII Global Conference (SRII), 2012 Annual*, pages 488–493, July 2012.
17. A. Norta. *Safeguarding Trusted eBusiness Transactions of Lifecycles for Cross-Enterprise Collaboration*. <http://tinyurl.com/lghxtrx>, 2012.
18. A. Norta. Creation of Smart-Contracting Collaborations for Decentralized Autonomous Organizations (forthcoming). In *14th International Conference on Perspectives in Business Informatics Research (BIR’15), Tallinn, Estonia*, August 2015.

19. A. Norta, P. Grefen, and N.C. Narendra. A reference architecture for managing dynamic inter-organizational business processes. *Data & Knowledge Engineering*, 91(0):52 – 89, 2014.
20. A. Norta and L. Kutvonen. A cloud hub for brokering business processes as a service: A "rendezvous" platform that supports semi-automated background checked partner discovery for cross-enterprise collaboration. In *SRII Global Conference (SRII), 2012 Annual*, pages 293–302, July 2012.
21. A. Norta, L. Ma, Y. Duan, A. Rull, M. Kõlvart, and K. Taveter. econtractual choreography-language properties towards cross-organizational business collaboration. *Journal of Internet Services and Applications*, 6(1):8, 2015.
22. OASIS. *eXtensible Markup Language (SOAP) 1.1*. <http://www.xml.org/>, 2006.
23. D. Oppenheim, S. Bagheri, K. Ratakonda, and Y.M. Chee. Agility of enterprise operations across distributed organizations: A model of cross enterprise collaboration. In *SRII Global Conference (SRII), 2011 Annual*, pages 154–162, March 2011.
24. B.S. Panikkar, S. Nair, P. Brody, and V. Pureswaran. Adept: An iot practitioner perspective, 2014.
25. M.P. Papazoglou and D. Georgakopoulos. Service-oriented computing. *Communications of the ACM*, 46(10):24–28, 2003.
26. T. Patron. *The Bitcoin Revolution: An Internet of Money*. Travis Patron.
27. R. Rodrigo, J. Zhou, and J. Lopez. On the features and challenges of security and privacy in distributed internet of things. *Computer Networks*, 57(10):2266 – 2279, 2013. Towards a Science of Cyber Security Security and Identity Architecture for the Future Internet.
28. F. Rosenberg and S. Dustdar. Towards a distributed service-oriented business rules system. In *Web Services, 2005. ECOWS 2005. Third IEEE European Conference on*, pages 11 pp.–, Nov 2005.
29. T. Ruokolainen, S. Ruohomaa, and L. Kutvonen. Solving service ecosystem governance. In *Enterprise Distributed Object Computing Conference Workshops (EDOCW), 2011 15th IEEE International*, pages 18–25. IEEE, 2011.
30. S.Y.W. Su, Xuelian Xiao, J. DePree, H.W. Beck, C. Thomas, A. Coggeshall, and R. Bostock. Interoperation of organizational data, rules, processes and services for achieving inter-organizational coordination and collaboration. In *System Sciences (HICSS), 2011 44th Hawaii International Conference on*, pages 1–10, Jan 2011.
31. M. Swan. Blockchain thinking: The brain as a dac (decentralized autonomous organization). In *Texas Bitcoin Conference*, pages 27–29, 2015.
32. N. Szabo. Formalizing and securing relationships on public networks. *First Monday*, 2(9), 1997.
33. A. Tripathi and B. Parihar. E-governance challenges and cloud benefits. In *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on*, volume 1, pages 351–354, June 2011.
34. W.M.P. van der Aalst, K.M. van Hee, A.H.M. ter Hofstede, N. Sidorova, H.M.W. Verbeek, M. Voorhoeve, and M.T. Wynn. Soundness of workflow nets: classification, decidability, and analysis. *Formal Aspects of Computing*, 23(3):333–363, 2011.
35. Y. Wei and M.B. Blake. Service-oriented computing and cloud computing: Challenges and opportunities. *Internet Computing, IEEE*, 14(6):72–75, 2010.
36. H. Weigand, W.J. van den Heuvel, and M. Hiel. Business policy compliance in service-oriented systems. *Information Systems*, 36(4):791 – 807, 2011. Selected Papers from the 2nd International Workshop on Similarity Search and Applications (SISAP) 2009.