A Semantic Rule-Based Approach for Software Privacy by Design

Fatemeh Zarrabi, Allan Brimicombe
University of East London, United Kingdom
shadi.zr@uel.ac.uk , a.brimicombe@uel.ac.uk

Abstract- Information system business is currently witnessing an increasing demand for system conformance with the international regime of GRC Governance, Risk and Compliance. Among different compliance approaches, data protection and privacy laws plays a key role. In this paper, we propose a compliance requirement analysis method from early stages of system modelling based on a semantically-rich model, where a mapping can be established from data protection and privacy requirements defined by laws and regulations to system business goals and contexts. The early consideration of requirements satisfies Privacy by Design, a key concept in General Data Protection Regulation 2012. The proposed semantic model consists of a number of ontologies each corresponding to a knowledge component within the developed framework of our approach. Each ontology is a thesaurus of concepts in the compliance related to system along with relationships and rules between these concepts that encompass the domain knowledge. The main contribution of the work presented in this paper is the ontology-based compliance framework that demonstrates how description-logic reasoning techniques can be used to simulate legal reasoning requirements employed by legal professions against the description of each ontology.

Keywords. Ontology, compliance, risk analysis, data protection, security, privacy by design, Requirement Engineering

1. INTRODUCTION

Data Protection Laws in national and international territories have been considered as the key tool in compliance to privacy. Data Protection Directive 1998 of European Union and its recent mandate, General Data Protection Regulation (GDPR) [1]- in 2012, has been taken very serious recently in privacy compliance of information systems. One of the most important aspects of compliance to privacy is considered in Article 25 of GDPR for implementation of a "privacy by design" approach as part of organisational IT-systems and processes. It requires that data protection is designed into the development of business processes for products and services, or to be said, a before-the-fact approach to compliance to data protection. This is in contrast with after-the-fact compliance approaches which are traditional solutions in which compliance is audited when the final product is working and running in application areas. They include bulk of existing software solutions that generate audit reports against hard-coded checks performed on the running system or supports documentation and reporting of internal controls such as Microsoft Office Solutions Accelerator for Sarbanes-Oxley [2]-, IBM Lotus workplace [3]-, SAP GRC (Governance, Risk and Compliance) Solution [4]-.The concept of privacy by design had been initially introduced by Dr Ann Cavoukian [5]-. Her team initially suggested to simply take a few 'PETs' (Privacy Enhancing Technologies) and add a good dose of security and privacy in form of user identity protector technologies (pseudo-identity) in design and implementation of information systems [6]-. But later Dr Ann Cavoukian acknowledged that PETs are not general answer to PbD and stands for a pro-active integration of technical privacy principles in a system's design and the recognition of privacy in a company's risk management processes. PbD also has been addressed by different information commissioners around the world such as ICO in the UK [7]-. In brief, **Privacy by Design** is an approach to system engineering, which takes privacy into account throughout the whole engineering process in which human values should be considered in a well-defined manner throughout the whole process. System engineering focuses on analysing and eliciting customer needs and required functionality of systems early in the development cycle. This process includes fully understanding of all stakeholders involved in the system. System modelling and simulation plays a key role in system analysing and popular tools such as UML are used in system engineering. However, privacy by design in software systems means making software under development to operate according to data protection law and any related policy and standard such as ISO 27000 and thus privacy plays an increased role in any software development process. To respond to the mentioned demands of the General Data Protection Regulation regarding to the Privacy by

Design and the urgent requirement to apply to the new regulation, this research has proposed a novel framework where separate components correspond to each element within Privacy by Design. This includes a separate component for GPDR along with all its concepts and their relationships such as its stakeholders and rules. A specific component also has been considered for system modelling in order to analyse and design the system context to which the GPDR should apply. In order to refine GDPR requirements with security and privacy standards and guidelines, separate components are also considered in our framework for the key standards and guidelines in this area. We have opted key international security standard of ISO 27000 series [8]- and ISO 29100 [9]-. We also have used guidelines provided by ICO. Privacy impact assessment has been taken as the other element of our proposed framework in order to assess privacy and security risks. Our literature review indicates an absence of a systematic, integrated and automated compliance solution to various elements of privacy compliance from laws to policies and standards and risk assessment. In a situation where concepts in these elements are related, the integrated solution that has been offered in the proposed approach makes links between different laws and policies concepts and apply them to a modelled system, thus makes the task of compliance easier. The links are in a way that requirements of each compliance component are refined by further and detail requirements from others. These tasks all have been managed using a knowledge-based semantic approach in which a common ontological infrastructure is provided. In this sense we use a collection of ontologies to shape and categories the information regarding each component of our framework.

The reminder of this paper is structured as following: Section 2 introduces the overall and classical procedure of our framework to provide a conceptual model for the ontology-based framework. This is later followed by number of sub-sections in order to describe each Ontology model. Section 3 represents a discussion of ontological concepts and techniques which has been used in order to achieve the objectives of our framework and following our automated approach RUL-SoPD. Conclusion and recommendation for future works had been discussed in Section 4.

2. SEMANTIC RULE-BASED APPROACH for SOFTWARE PBD (RUL-SoPD)

Providing automated IT solutions for legal professionals has a long history in computer science. Among them, using semantic webs and developing ontology of legal concepts is a well-established approach in the field of artificial intelligence ([10]-[11]-). Semantic Web is the vision of web of the future with the structure of information that is understandable to computer, so the later can perform many tasks instead of humans"[12]-. The essence of semantic web is the resources of information represented in triples of knowledge. The representation is done by a formal language in order to encode an ontology. As the foundation and main structure of ontological systems, ontology languages allow construction of knowledge in a specific domains such as law. They normally include reasoning rules in order to define and impose more knowledge. Resource Description Framework (RDF) which describes the resources in form of triples is one of the basic ontological language. The triples of subject, relation and object include components of ontology as classes or concepts, objects or individuals and relations which are defined by properties.

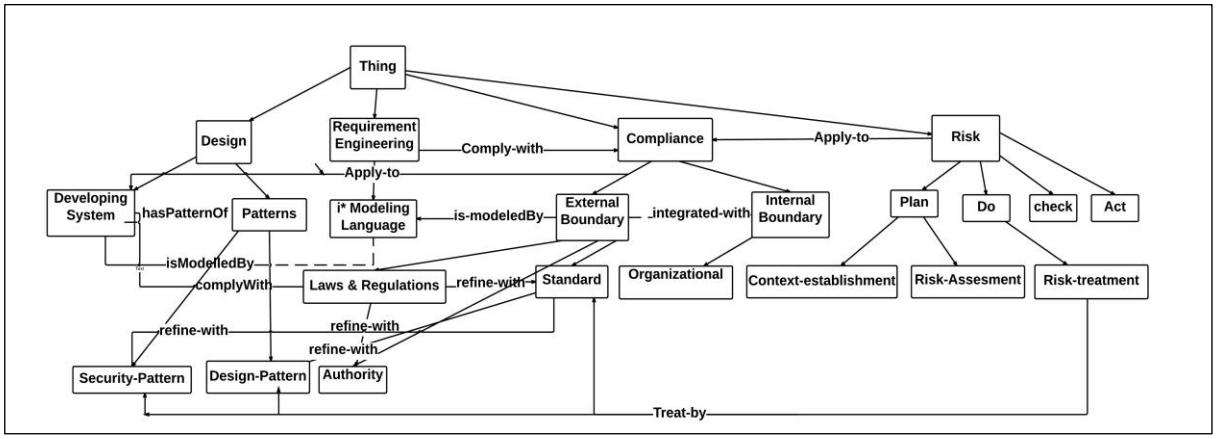

Fig. 1 RUL-SoPD Framework

The more advanced relationship between concepts is being constructed using Web Ontology Language (OWL). OWL extends the vocabulary of RDF by providing more meanings to the triples. OWL is based on description logic, thus its construction has well-defined meanings which are used to describe domain concepts and their relationships in an ontology. Description logic enables automated logical reasoning techniques. The reasoner allows logical conclusion and consistency checks on classes, individual instances and properties. However, OWL does not include a composition conductor in order to capture chain relationships. Semantic Web Rule Language (SWRL) extends OWL with Horn-like rules based on the rule mark-up language RuleML. It enables automatic deduction of new knowledge from existing facts. Thus, SWRL rules ultimately increase the expressivity of OWL-DL. Ontology can be constructed manually using dedicated software tools such as TERMINAE, PROTEGE, HOZO and others [13]-. Here we have used Protégé and FACT ++ as the reasoner in order to construct our ontologies and further our approach. Our semantic model which is called RUL-SoPD, consists of four main ontologies: *Requirement Engineering*, *Design, Compliance* and *risk* (Fig 1). Design ontology is where designing system and its relevance compliance and design resources are being discovered.

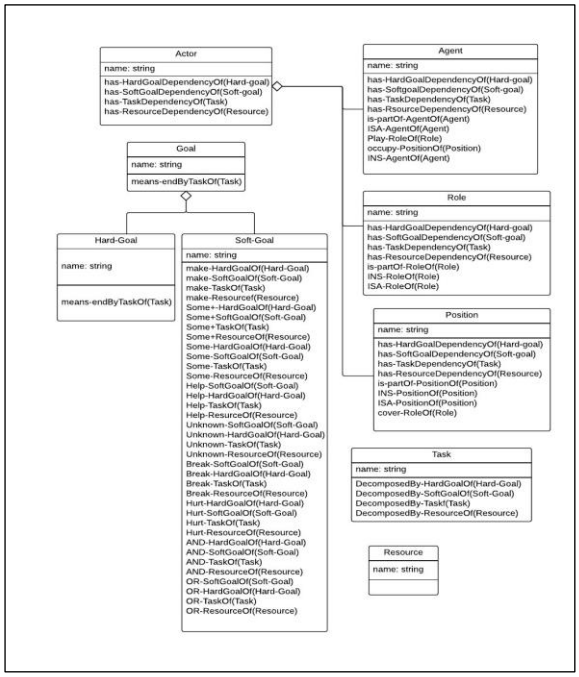

Fig. 2 Requirement Engineering (i*) Ontology

In Requirement Engineering ontology, system is being modelled and its requirements are elicited by a requirement engineering methodologies known as i*. Compliance ontology has concepts and definitions from laws and regulations and finally Risk ontology performs risk analysis.

A. System Design

In order to have a design element for privacy by design in our approach and to map laws to system context, we use a requirement engineering (system modelling) component in our framework and also a design-pattern-based component. System modelling component is known as i* which models the context of system in the format of its stakeholders (Actor) dependencies to other agents in order to achieve their Goals, perform their Task and access their Resources [14]-. i* modelling language is an agent-oriented and goal-modelling approach to the early stages of requirement engineering. A goal dependency is the highest level of an agent desire. A goal may be soft or hard, depending on whether it indicates a functional or non-functional requirement of the agent. At the refinement stage, an agent may adopt task dependency or resource dependency in

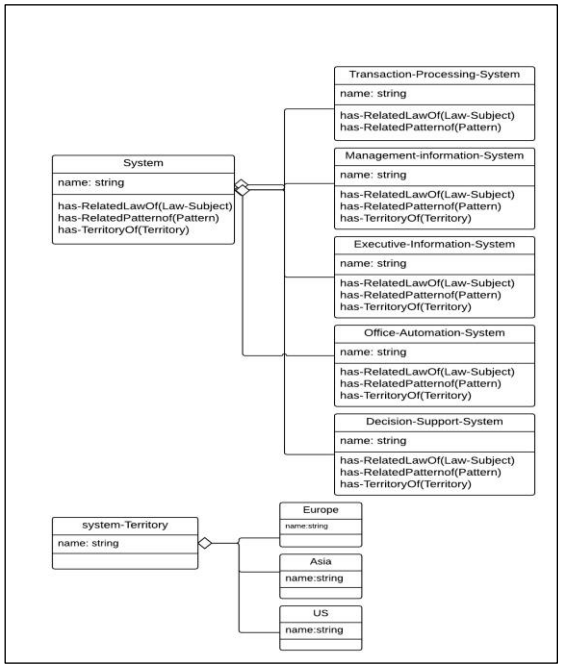

Fig. 3 Design Ontology

order to satisfy its goal or task. i* takes advantage from a number of other links between its concept, such as Means-end, Contribution and Association. They are almost used for the purpose of refinement of i* concepts or to initiate relationships between actors in each system. In such a systematic approach that utilizes concepts of Actors, Goals, Tasks and Resources, the requirement engineer is able to progress through an incremental process of extracting system requirements. A unique ontology is considered in our model in order to support the i* system modelling component of our framework. Fig 3 represents the taxonomy of i* as it is developed as a component of our compliance framework in the platform of i* ontology.

We have totally 4 classes, 5 sub-classes and 70 object-properties in i* ontology (matrix of links & classes). The primitives in the category hieratically of classes include actor, goal, task and resource concepts. The child categories of goal entity as soft-goal and hard-goal share common characteristics but are otherwise heterogeneous. Same is true regarding sub-classes of actor as agent, role and position. Different types of dependencies between i* concepts are drawn as object properties which relates types of classes. Refinement levels of goals and tasks (means-end, decompose, contribution) are also available as properties. Associate links between actors are also considered as object-properties.

we also use *Design Ontology,* in order to design the system using design and security patter. The type of design patterns and the stage of process when it can be used, is depended on the abstract level of the requirement and is flexible. Design patterns which record the design experiences of expert programmers are being reused as references for those with fewer experiences. It is mostly used after system analysis when requirements are depicted. However, It also has been proved that design patterns have modified the traditional approach to system modelling [15]-. Therefore, they can be used before modelling in order to save time and effort. We also use security patterns in order to refine legal and standard and also risk depicted requirements with a solution from patterns. Design Ontology also helps developers in order to use other experiences to find relevant laws and regulations to the type of system. Fig 3 represents the taxonomy of Design Ontology.

B. Compliance

In legal and judgment system a rule (constitutional provision or a statute in law which as an enforcement statement establishes a standard of conduct) acts as a formula to make a decision in a case of judgement (a civil or criminal processing, action, suit or controversy at law or equity) [16]-. In such situation lawyers and judges argue and try to find and match

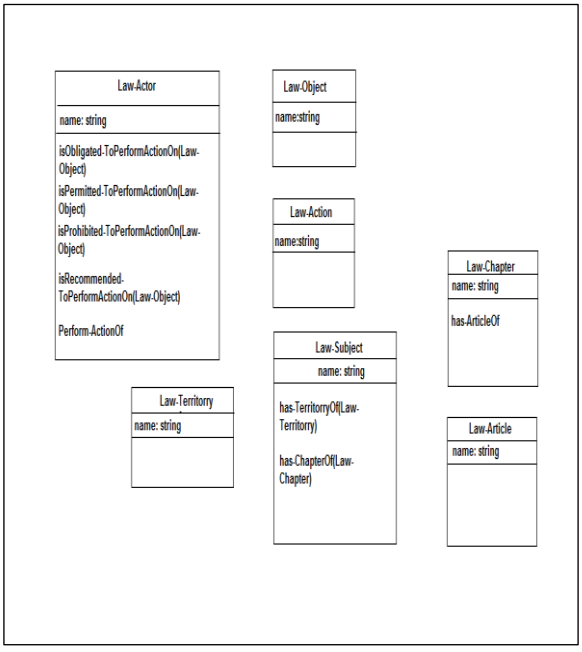

Fig. 4 Law Ontology

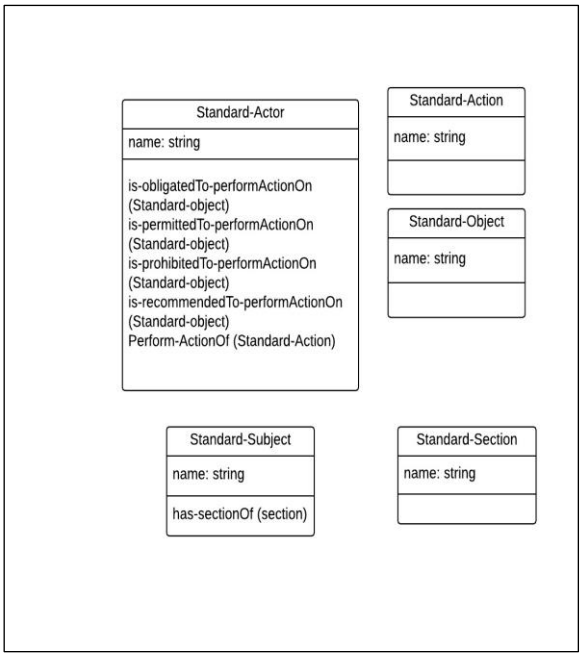

Fig. 5 Standard Ontology

the rules of law which applies to a given set of facts of the occurred case in a logical process of syllogism. This process is called Legal Reasoning or Legal Analysis. To do so, rules are broken down into three separate components: Test (Fact: condition and circumstances of law), Result (happens when tests are available) and Mandatory Terms (determines if the result is mandatory/obligation (shall), prohibitory/prohibition (shall not), discretionary/permission (May) or a recommendation (should)). Same analysis process with some differences is being followed in our approach in order to analyse a rule of law and apply it to a case (developing system context here), thus compliance will be achieved. Extracted facts and results from law text, and the cellular analysis of them build the compliance Ontology. Each extracted cell and elements from facts and results (nouns, verbs) provide a thesaurus of legal concepts that are categorised in this ontology into number of obligation, permission, prohibition or recommendation of a rule. We have categorised the extracted arguments from the rules of Data Protection Regulation into number of classes depending on their meaning and types (Fig 4). Law's stakeholders such as controller, data subject, processor, data representative and others are subclasses of the class *Legal-Actor.* Resources such as personal data, information, consent, contact details, identity and others are under a general category of Object*,* but still are categorised to sub-classes of object based on their type. *O*bligations, permissions, recommendations, prohibitions and results as the connector between two classes to build relationship between them, are represented in our ontology as object-properties*.* For refinement purposes, we also have ontologies made of standard (ISO 27000) and authority guidelines (ICO) taxonomies here (Fig 5). Since the standards and guidelines also impose recommendation (rights) to stakeholders to perform or not to perform an action, the categorization almost follow the same order in Law Ontology consisting of top classes of *ISO-Action*, *ISO-Actor* and *ISO-Object* (same is applicable to Authority (ICO) Ontology).

C. Risk Assessment

Based on the defied procedure of privacy by design, also GDPR, PbD shall always be supported by a well establishment of privacy impact assessment. Therefore, we opted an element of risk assessment in our framework. Our approach to risk assessment is based on ISO 27005 [17]- and all concepts and definition are taken from this standard. ISO 27005 introduces four stages of plan, do, act and check. Here we only address the first two stages and their sub activities of context establishment, Risk assessment and risk treatment. Information system risk assessment is a continues process in which the context of system is established and risks and threats are assessed using a risk assessment plan. Context establishment includes activities of determining scope and boundary of system (identifying system assets), determining risk evaluation criteria, risk acceptance criteria and Impact criteria. In Risk assessment, values of assets and risks are evaluated. Asset valuation is measured based on business criteria of the system, assets and the organization defined in context establishment and measured between 0-4 quantitative values. Risk valuation on assets is measured based on asset values, threat and vulnerabilities, likelihood of threat happening and chance of system exploitation. Here we are using Matrix with Predefined Values method from ISO . ;27500Table5 for risk evaluation. This Metrix is based on qualitative values of low, medium and high for threat likelihood and chance of exploitation and quantitative values from 0-9 for risks. Finally, in treatment, controls from standards and further security patterns are taken to control the high-valued risks, otherwise risk is ignored, trained or accepted. The ontology supporting the risk assessment component of this framework, includes relevant concepts to risk such as asset, threat, vulnerability, risk-actor, value and others (Fig 6). we also have sub-classes for assets and threat categorization based on Annex B, C and D of ISO27005.

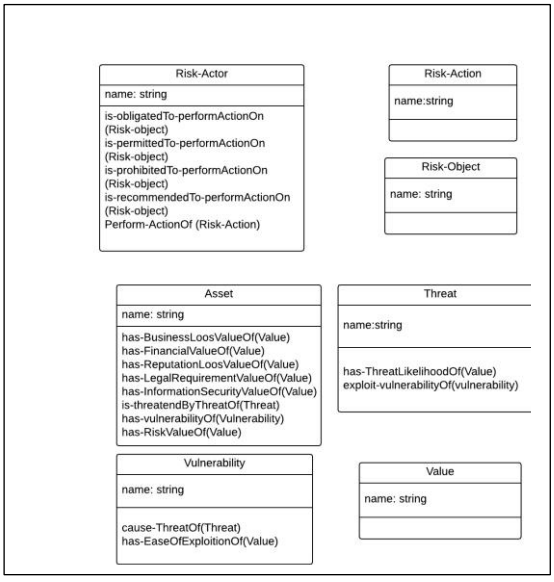

Fig. 6 Risk Ontology

The general orders of risk assessment address by ISO 27005, as explained before, are provided in Risk Ontology by object-properties and further ontological rules on them. The types of these properties are same to types of rights (obligation, permission, prohibition and recommendation) as explained in compliance ontology, plus to object-properties specified for Context Establishment tasks, Risk and asset Evaluation Criteria and Risk Acceptance Criteria. The considerable here is that we have omitted to have concepts from Scope & Boundary in Risk Ontology due to the fact that this task is already performed in our framework with component of i* modelling. In fact, the context of system is modelled there and scope and boundaries are specified before.

3. RUL-SoPD ONTOLOGY IMPLEMENTATION

In order to produce compliance effects, where laws and authority guidelines can be applied to system context and be mapped and refined to each other, our system needs an expressive rules language and a reasoning engine for to interpreting the rules. Indeed, we proposed a framework of integration of different components of privacy by design (RUL-SoPD). All these process as we called it legal reasoning in order to match facts of laws to system context, also to integrate our framework components, are supported in our approach with the aid of Logical Reasoning in Ontology and Semantic Web.

TABLE 1 RUL-SoPD ONTOLOGY CONCEPTS

Concept	Ontology
Fact	RO: <rdfs: Property rdf:id ="has-GoalDependencyOf" > <rdfs: domain rdfresource= "Actor"> <rdf:range rdfresource= "Goal" >
Rule	I. RO: <ruleml:imp <ruleml:_body> <swrlx:individualPropertyAtom swrlx:property=" process-PersonalDataOf "> <ruleml:var>x1</ruleml:var><ruleml:var>x2</ruleml:var></swrlx:individualPropertyAtom><swrlx:individualPropertyAtom swrlx:property="process-processOf"><ruleml:var>x2</ruleml:var> <ruleml:var>x3</ruleml:var></swrlx:individualPropertyAtom> <swrlx:individualPropertyAtom <owlx:Individual owlx:name="#Processor" /> </swrlx:individualPropertyAtom></ruleml:_body><ruleml:_head><swrlx:individualPropertyAtom swrlx:property="is-obligated-ByDPA-Art5(a)-To-processLawfully-PersonalDataOf"> <ruleml:var>x1</ruleml:var><ruleml:var>x3</ruleml:var> </swrlx:individualPropertyAtom> </ruleml:_head> </ruleml:imp II. RsO: <ruleml:imp <ruleml:_body> <swrlx:individualPropertyAtom swrlx:property="has-BusinessLoosValueOf"><ruleml:var>x1</ruleml:var> <ruleml:var>x2</ruleml:var> </swrlx:individualPropertyAtom><swrlx:individualPropertyAtom swrlx:property="hasFinancialValueOf"><ruleml:var>x2</ruleml:var><ruleml:var>x3</ruleml:var></swrlx:individualPropertyAtom> <swrlx:individualPropertyAtomswrlx:property="has-egalRequirementValueOf"><ruleml:var>x3</ruleml:var> <owlx:Individual owlx:name="#Asset" /> </swrlx:individualPropertyAtom></ruleml:_body><ruleml:_head> <swrlx:individualPropertyAtom swrlx:property="has-AssetValueOf "> <ruleml:var>x1</ruleml:var><ruleml:var>x3</ruleml:var> </swrlx:individualPropertyAtom> </ruleml:_head> </ruleml:imp
DL map, Integrate	I. LO: <swrlx:sameIndividualAtom> <ruleml:var>x1</ruleml:var> <ruleml:var>x2</ruleml:var><ruleml:var>x3</ruleml:var> <owlx:Individual owlx:name=" $Controller_{lo}$ " /> <owlx:Individual owlx:name="$Controller_{SO}$"/> <owlx:Individual owlx:name="$Controller_{ICO}$"/> </swrlx:sameIndividualAtom> II. SO: <ruleml:imp <ruleml:_body> <swrlx:individualPropertyAtom swrlx:property=" process-PersonalDataOf "> <ruleml:var>x1</ruleml:var><ruleml:var>x2</ruleml:var></swrlx:individualPropertyAtom><swrlx:individualPropertyAtom swrlx:property=" process-processOf "> <ruleml:var>x2</ruleml:var><ruleml:var>x3</ruleml:var></swrlx:individualPropertyAtom> <swrlx:individualPropertyAtom <owlx:Individual owlx:name="#Processor"/></swrlx:individualPropertyAtom> <swrlx:individualPropertyAtom swrlx:property="is-obligated-ByDPA-Art5(a)-To-processLawfully-PersonalDataOf"><ruleml:var>x1</ruleml:var> <ruleml:var>x3</ruleml:var></swrlx:individualPropertyAtom><ruleml:_head> <swrlx:individualPropertyAtom swrlx:property="is-obligated-ByISO29100-ToProvideNoticeOf " "><ruleml:var>x1</ruleml:var><ruleml:var>x3</ruleml:var> </swrlx:individualPropertyAtom> </ruleml:_head> </ruleml:imp

Here we are listing ontological concepts and techniques which are used in our approach in order to aim us achieving our compliance objectives; Ontology Facts as triple of knowledge; *subject-objectProperty/Predicate-Object (a statement)* to represent knowledge in ontology has been used in all ontologies of our approach; Rules which enable automatic deduction of new knowledge from existing facts are used in Compliance and Risk ontologies ; description logic operators including mapping (equalization), integration (combined Rules), inheriting (individualization) and refinement which makes connection between different ontologies and finally Logical Reasoner which works based on Description Logics and Rules in order to enable automated logical reasoning techniques, are also used in all ontologies based on their application. The reasoner allows logical conclusion and consistency checks on classes, individual instances and properties. Applying laws to modelled system is provided using some ontological processes such as individualization and reasoning. Individuals are instances of ontology classes which inherit all the attributes and properties from classes and are the last in their heretical and cannot be instanced anymore. This process is where a real-world scenario from a specific domain is constructed using the knowledge represented in our ontology (Requirement Engineering Ontology/System Context). Some examples of mentioned concepts are illustrated below in TABLE1 and in formats of RDF, OWL or SWRL languages. To make an abstract view, we have assigned Requirement Engineering Ontology in short as RO, Law Ontology as LO, Standard Ontology as SO and Risk Ontology as RSO.

4. CONCLUSION & FUTURE WORK

Here we represented RUL-SoPD as an integrated approach toward privacy by design which can be used in software development or business process compliance to GDPR. More works in future can be focused on other laws and standards or to the integration of GDPR with other laws such as E-Commerce Law. We also are focusing on developing and introducing Privacy Patterns in order to be added in our System Ontology as framework solution for privacy requirements.

RFERENCES

[1]- (2012), European Parliament, *REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on the protection of individualswith regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation).* [online]. Available: https://ec.europa.eu/info/law/law-topic/data-protection_en

[2]- G. Rochelle, *Microsoft plans App to aid companies with Financial Controls*. Contents Issue Magazine. ISSN 0893-8377, 2003.

[3]- International Business Machine Corporation, *IBM Lotus workplace for Business Controls & Reporting*. IBM Redbooks Publication. REDP-4021-2004.

[4]- S. Scholer and O. Zink, *SAP Governance, Risk and Compliance*. 2nd edition. SAP PRESS, 2008.

[5]- A. Cavoukian, *Privacy by Design The 7 Foundational Principles Implementation and Mapping of Fair Information Practices*, Information & Privacy Commissioner, Ontario, Canada, 2011.

[6]- H. Rossum, *Privacy-enhancing Technologies: the path to anonymity*. Den Haag: ISBN 90 346 32 02, 1995.

[7]- (2008), Information Commission Office, United Kingdom, *Privacy by Design*, [online], Available http://www.ico.gov.uk/upload/documents/pdb_report_html/privacy_by_design_report_v2.pdf.

[8]- (2013), International Organisation of Standardisation. *Information technology — Security techniques, Information security management systems — Overview and vocabulary. ISO/IEC 27000*. [online] Available: http://www.27000.org/.

[9]- (2012), International Organisation of Standardisation, *Privacy Framework. ISO/IEC 29100*. [online]Available: http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

[10]- J. Brekeur, R. Winkel, *Use and Reuse of Legal Ontologies in knowledge Engineering and Information Management*. ICAIL Workshop on Legal Ontologies & Web Based Legal Information Management, 2003.

[11]-R. Schmidt, CH. Bartch and R. Oberhauser. *Ontology Based Representation of Compliance Requirements for Service Processes*

[12]-G. Jakus, V. Milafinovic, S. Omerovic, and S.Tomazic. *Concepts, Ontologies and Knowledge Representation*. SpringerBriefs in Computer Science, 2013.

[13]-T.R. Gruber, *Toward Principle for the Design of Ontologies Used for Knowledge Study.* Computer. Stud. 43(5-6)

[14]- E.Yu, P. Giorgini, N. Maider and P. Mylopoulos, *Social Modeling for Requirement Engineering,* Cambridge, MA:MIT Press. 2011.

[15]- E. Gamma, R. Helm, R. Johnson, J. Vissides, *Design Patterns: Elements of Reusable Object-oriented Software*, 40th Edition. Addison Wesley, 2012

[16]- H. Zellig, *A Grammar of English on Mathematical Principles*, New York: John Wiley and Sons, ISBN 0-471-02958-0

[17]- (2013), International Organisation of Standardisation Information technology — *Security techniques, Information security risk management — Overview and vocabulary. ISO/IEC 27005*. [online]. Available: http://www.27000.org/