Entering the Digital Customer Onboarding Era: How the Semantic Web Can Help

Fabien Chevalier[1] and Sébastien Ferré[2]

[1] AriadNEXT, ZAC des Champs Blancs, 1219 Avenue des Champs Blancs,
35510 Cesson-Sévigné, France
Email: `fabien.chevalier@ariadnext.com`
[2] IRISA/Université de Rennes 1, Campus de Beaulieu, 35042 Rennes, France
Email: `ferre@irisa.fr`

Abstract. In this presentation we showcase the use of Semantic Web in AriadNEXT's IDCHECK.IO document verification service. This service has been introduced for a number of years now. It has recently seen a speed up in its market adoption due in large amount to the introduction of a new semantic web data model workflow. We will start by introducing the research project behind this technology upgrade, then explain our approach, focusing on what problems the use of semantic web solves, and finally give some highlights of the perceived business benefits.

1 Introduction

The IDFRAud [3] project is an industrial research project led by French ID document verification leader company AriadNEXT, that provided the technology behind its document verification service IDCHECK.IO [2]. One of the objectives of IDFRAud is to propose an automatic solution for ID verification that can handle documents issued from a large set of countries. The solution should be able to execute specific controls according to the ID model (type, country, generation, etc) thanks to a knowledge base. The core idea of IDFRAud project is to provide an automatic verification system for identity documents in order to replace existing manual verification processes. The different components of ID analysis and verification in IDFRAud are driven by a set of control rules. In order to guarantee an interpretable and adaptive behavior at each ID analysis step, the identity document descriptions are organized by a knowledge management module.

2 Technical approach

One of the requirements of our knowledge management module is its interoperability and portability. It is preferred to store the data in a standard way in order to be able to use other tools such as Sewelis [4] to navigate our data. We also have another strong requirement: we must be able to easily extract a subset of the knowledge base to run on mobile platforms, where C/C++ language

rules. After a thorough analysis of existing technologies, we decided to use RDF for its versatility/flexibility of knowledge modelling using graphs, coupled with its capability to bring formal structure to the knowledge using Web Ontology Language (OWL). RDF serialization formats are also strongly standardized, as a difference to many database systems. This standardization encouraged engineers to write RDF processing tools in many differents langages, including those available on mobile platforms.

Providing an efficient, consistent, and descriptive enough model for ID documents proves to be a very challenging task. The biggest issue encountered is the very high diversity of how ID documents look like, which makes it really difficult to design a data model that fits all cases. As such it is anticipated that the model will see serious evolutions with the number of supported documents.

We decided to build two in-house frameworks to fulfill those requirements:

1. *ModelEditor* is a UI that provides guided data editing. It enables a non-expert user to fill in data that is consistent with the underlying OWL ontology. It can be made fully generic, as the UI is built dynamically for the OWL ontology.
2. *AutoRDF [1]* is a code generation utility which makes code base maintenance easy by being able to follow a constantly evolving ontology. AutoRDF can be of some use to any kind of project that needs to manipulate RDF data that conforms to an existing ontology. The use of modern C++ makes it portable to a wide variety of platforms, including all mobile phone platforms, as well as most of the embedded world systems.

The steps used in our document definition process are now the use of *Protégé* open-source ontology editor for formal concepts definition, the use of *ModelEditor* to input new ID model data, and lastly *AutoRDF [1]* to code processing logic using the newly defined concepts. We have experimented that those three steps make for an efficient document description environment.

3 Business benefits

We will give a few direct an indirect metrics of the performance gains of this approach. We will give some key figures:

- Decrease in time needed to specify and support a new document model, from 6+ month to only a few days.
- Increase in business turnaround, including an astonishing 800% year to year peak growth rate.

References

1. AutoRDF - A framework for C++ proxy class generation from Web Ontology Language. `https://github.com/ariadnext/AutoRDF`
2. IDCHECK.IO - Fast and reliable ID document checking. `https://idcheck.io`

3. IDFRAud: An Operational Automatic Framework for Identity Document Fraud Detection and Profiling - Joint research project with AriadNEXT, IRISA, ENSP and IRCGN funded by ANR grant ANR-14-CE28-0012. `http://idfraud.fr/`
4. Ferré, S., Hermann, A.: Reconciling faceted search and query languages for the Semantic Web. Int. J. Metadata, Semantics and Ontologies 7(1), 37–54 (2012)