

# Answering Controlled Natural Language Questions over RDF Clinical Data

Naouel Karam<sup>1</sup>, Olga Streibel<sup>2</sup>, Aray Karjauv<sup>1</sup>, Goekhan Coskun<sup>2</sup>, Adrian Paschke<sup>1</sup>

<sup>1</sup>Fraunhofer FOKUS, Berlin, Germany

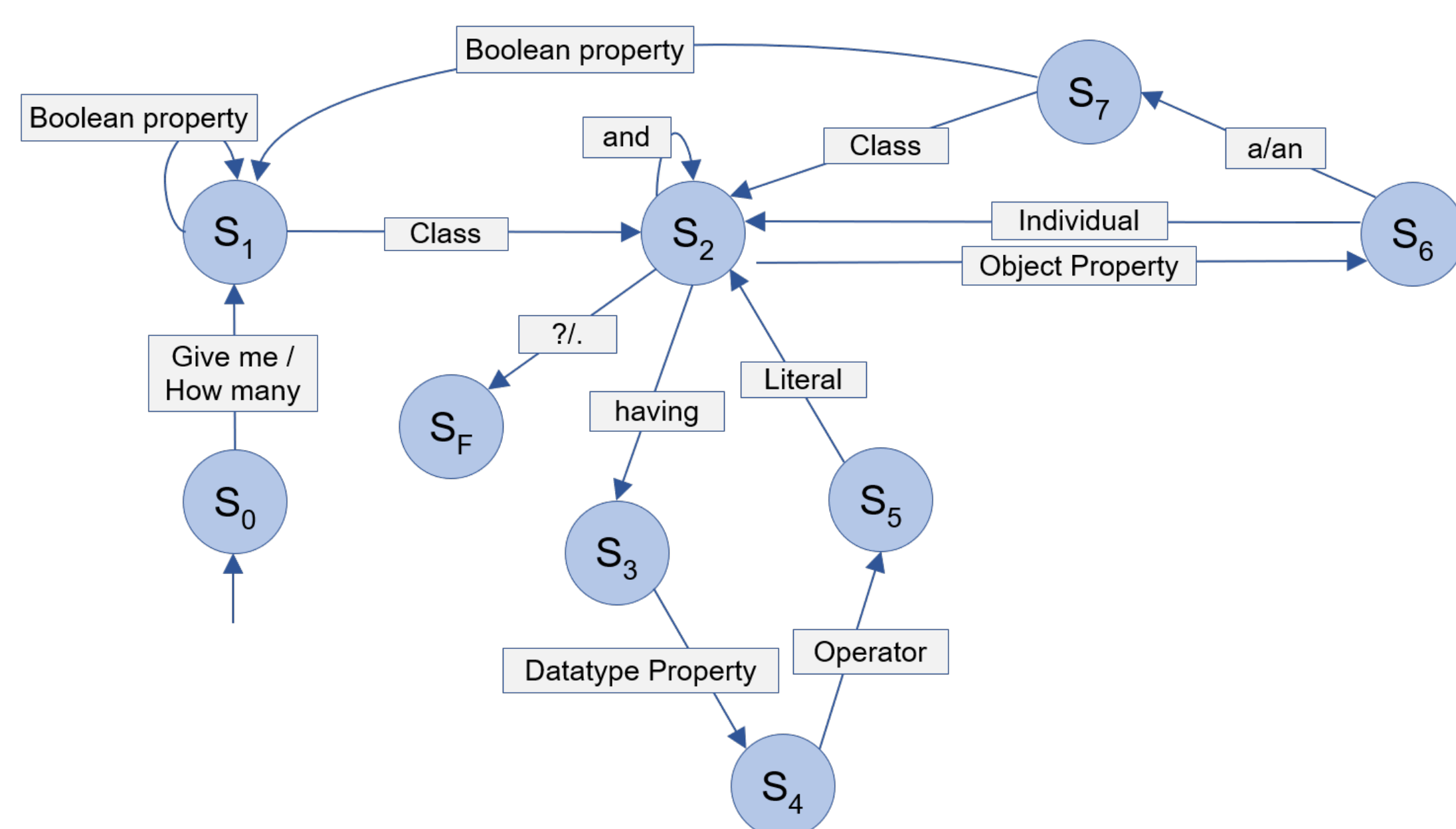
<sup>2</sup>Bayer AG, Berlin, Germany

## Objective

In this demo, we present our system for answering controlled Natural Language questions over RDF clinical data. Questions entered by a user through the proposed interface are annotated on the fly and suggestions are displayed based on an ontology driven auto-completion system. This approach assures a high level of usability and readability while preserving semantic correctness and accuracy of entered questions.

## Overview of askTONI

The process of question construction is based on the following finite state machine. askTONI makes the distinction between datatype and object properties, individuals, classes and literals.



## SPARQL query generation

The SPARQL query is generated on the fly. Each transition is associated with a query pattern that is added to the where clause of the query:

$$(S_1)(S_2) \text{ [class-variable] rdf:type } \langle \text{[class12]} \rangle$$

$$(S_6)(S_2) \text{ [class-variable] } \langle \text{[property26]} \rangle \langle \text{[individual62]} \rangle$$

where **class12** is the URI of the class entered by the user between the states  $S_1$  and  $S_2$ , **property26** the URI of the property entered between the states  $S_2$  and  $S_6$  and **individual62** the URI of the individual given by the user between  $S_6$  and  $S_2$ .

## Acknowledgements



This work was partially funded by the German Federal Ministry of Education and Research (BMBF) through the project QURATOR (grant no. 03WKDA1A).

## Contact Information

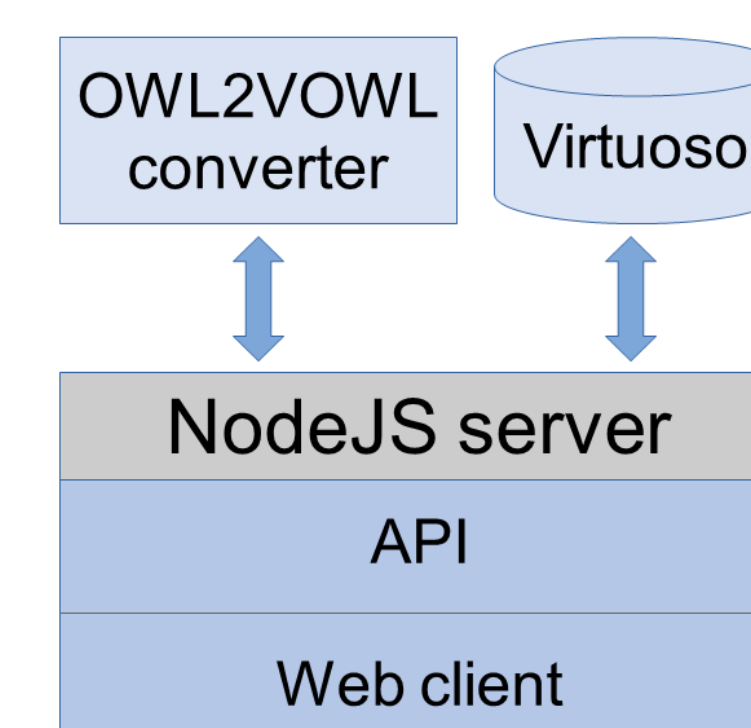
naouel.karam@fokus.fraunhofer.de

www.data-analytics-center.org

## System architecture

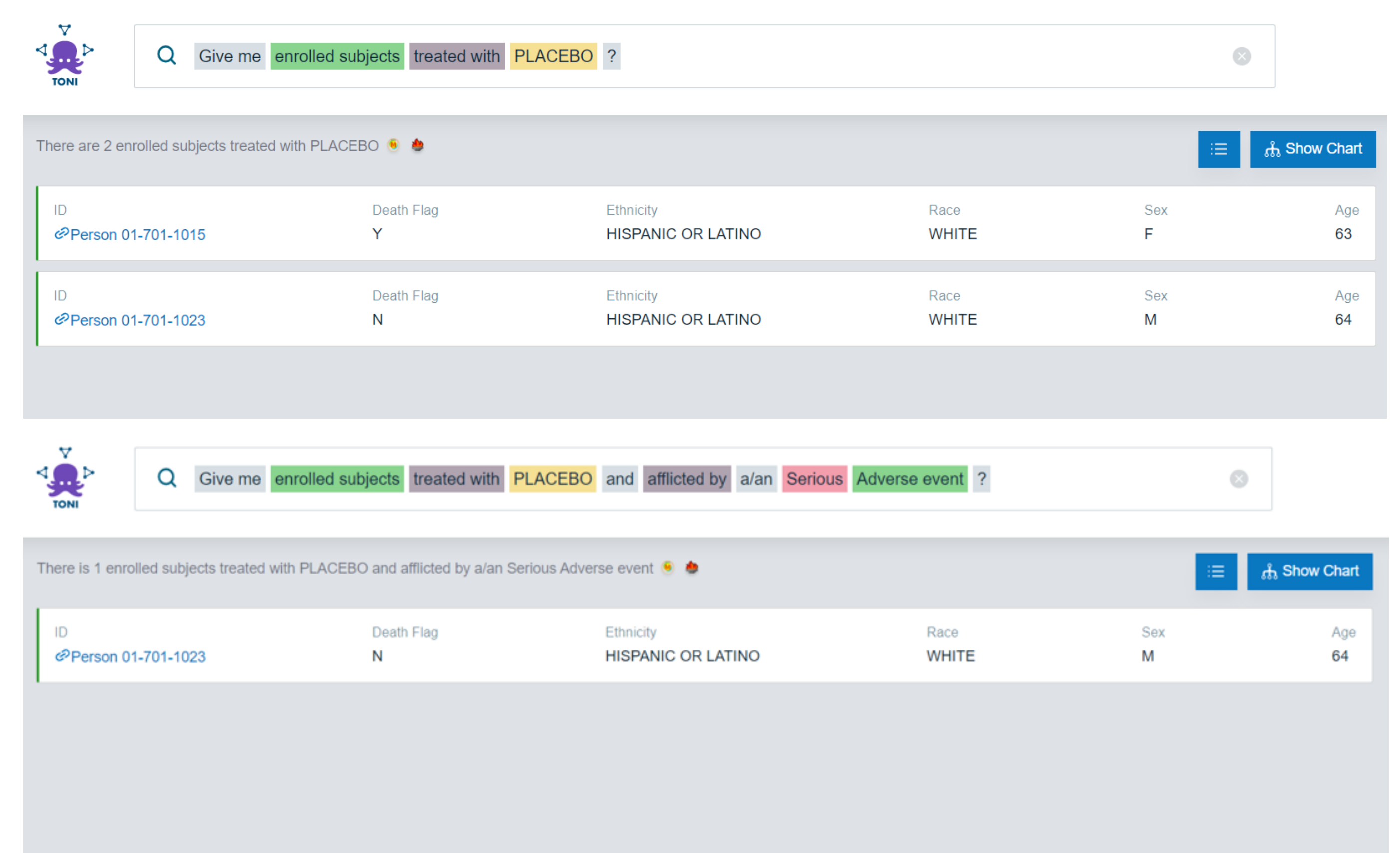
The main components of our system architecture are the following:

- **NodeJS server** intermediate layer between the user interface and the triple store. Communicates with other components over HTTP
- **Virtuoso server** triple store storing all RDF data
- **OWL2VOWL converter** service for converting RDF into JSON which is used to display the ontology graph



## Demonstration

We will demonstrate how the interface can be used to pose questions, guided by the system through suggestions. We will provide guidance and example questions. For visitors more interested in the models and RDF data behind the demo, we can also provide access to the PhUSE ontologies visualisation and the SPARQL endpoint.



ID	Death Flag	Ethnicity	Race	Sex	Age
Person 01-701-1015	Y	HISPANIC OR LATINO	WHITE	F	63
Person 01-701-1023	N	HISPANIC OR LATINO	WHITE	M	64

## Demo video



<https://owncloud.fokus.fraunhofer.de/index.php/s/5GWk6sG50UggI9o>