A SOLID Architecture to Weather the Storm of Real-Time Linked Data

by Miguel A. Martínez-Prieto, Carlos E. Cuesta, Javier D. Fernández and Mario Arias

Linked Open Data has increased the availability of semantic data, including huge flows of real-time information from many sources. Processing systems must be able to cope with such incoming data, while simultaneously providing efficient access to a live data store including both this growing information and pre-existing data. The SOLID architecture has been designed to handle such workflows, managing big semantic data in real-time.

The successful Linked Data initiative has driven the publication of big, heterogeneous semantic datasets. Data from the governments, social networks and relating to bioinformatics, are publicly exposed and interlinked within the Web of Data. This can be seen as part of the more general emerging trend of Big Data, where the actual value of our data depends on the knowledge which can be inferred from them, in order to support the decision making process. The term "Big Semantic Data" refers to those RDF datasets for which *volume*, *velocity*, and *variety* demand more computation resources than are provided by traditional management systems. Whilst this can usually mean terabytes or petabytes, a few gigabytes may be enough to collapse an application running on limited devices. Both high-performance and small devices must be considered at the confluence of the Web of Data and the Internet of Things (IoT).

The IoT hosts diverse potential real-time sources of RDF, such as RFID labels, Web processes, smartphones and sensors. All of these can be exploited as a whole in the emergent "smart-cities". With the cohabitation of diverse devices, this scenario results in a *variety* of data flows; a smart-city comprises sources about the daily life of the city (weather sensors, distributions of people at points of interest, traffic congestion, public transport), but it generally requires a less-dynamic background knowledge, such as road maps, infrastructure and organization relationships. The most interesting applications are those integrating different sources to provide services, such as predictions of traffic patterns depending on the weather or certain specific days, or other decision support for city management.

Traditionally, platforms managing Big Semantic Data and those dealing with real-time information follow completely different design principles. Big Semantic Data management involves, in general, heavyweight batch processes focused on generating suitable indexes to enable efficient SPARQL resolution. Real-time management, on the contrary, focuses on continuous queries executed directly against the incoming stream. In practice, some data need to be discarded, or in the best case, are stored in non-optimized data structures. We argue that combining the two philosophies provides the benefits of both worlds, but how to do it correctly is still an open challenge.

To this end, we propose Service-OnLine-Index-Data (SOLID) [1] as a scalable architecture that addresses separately the complexities of Big Data and real-time data management. SOLID proposes a three-tiered architecture (see Figure 1). The *Content tier* contains the repository of information. It comprises three layers optimized to provide efficient data retrieval depending on the stage of the information (historic versus live data). The *Data layer* stores raw RDF in compressed but accessible binary form, leveraging compactness and guaranteeing data immutability. The *Index layer* deploys a lightweight configuration of data structures on top of the *Data layer* to allow efficient querying capabilities with a minimal extra memory footprint. The *Online layer* is designed to store incoming RDF triples at a very high throughput. Since its performance degrades progressively, it must be dumped to the historic store eventually. The *Merge tier* implements this responsibility. It runs a batch process which integrates data from the *Online layer* into the *Data Layer*, updating the *Index layer* when the process finishes. Note that this integration process does not stop the overall operation. The *Service Tier* provides a SPARQL-based API to user applications. Its query processor

instantiates a dynamic pipeline which delegates the retrieval to the *Online* and the *Index* layers, and binds their answers to obtain the final result.

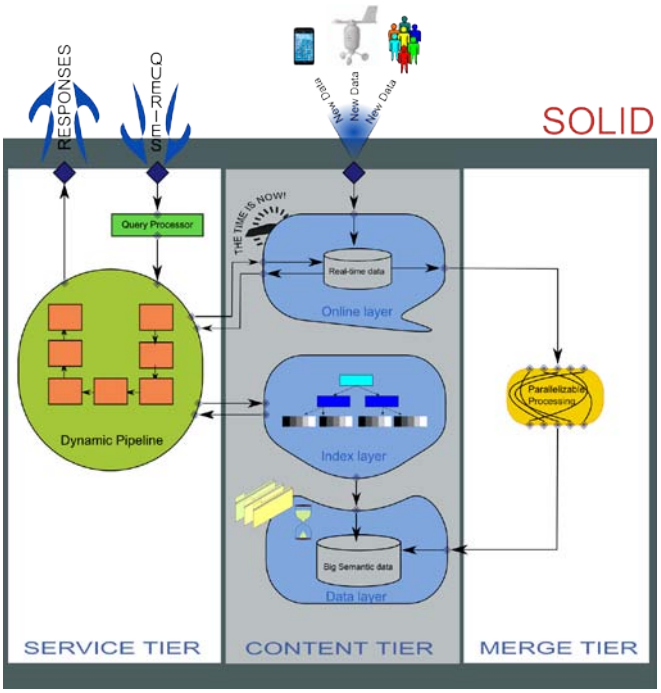

Figure 1: The SOLID architecture.

We have implemented the SOLID components using state-of-the-art technology. On the one hand, the *Data* and *Index* layers are deployed using the open RDF/HDT format [2]. It serializes the Big Semantic Data in highly compressed space, while providing SPARQL resolution. The *Online layer* uses a traditional RDF Store. Here, we make sure a competitive insertion throughput by keeping small to medium size collections; when the size is too big, data are sent to the *Data Layer* to be stored as historical information. Finally, the *Merge* and the *Service* tiers are implemented natively to ensure their efficiency. Our preliminary results show that the different layers excel at their tasks, and given that the *Online layer* contains few data, the integration is very lightweight, leading to competitive query performance in the latest SPARQL benchmarks, comparable to state-of-the-art RDF Stores but with a better write throughput.

Our future work includes tuning these layers and improving the algorithms that communicate them, providing SOLID as a general architecture, able to adapt to special needs of potential applications.

Links:
DataWeb Research Group: http://dataweb.infor.uva.es
RDF/HDT Project: http://www.rdfhdt.org
HDT W3C Member Submission: http://www.w3.org/Submission/2011/03/

References:
[1] C.E. Cuesta, M.A. Martínez-Prieto, J.D. Fernández: "Towards an Architecture for Managing Big Semantic Data in Real-Time", in proc. of *ECSA*, 2013, dx.doi.org/10.1007/978-3-642-39031-9_5
[2] J.D. Fernández, M.A. Martínez-Prieto, C. Gutiérrez et al.: "Binary RDF representation for publication and exchange", *JWS* vol. 19, 2013, dx.doi.org/10.1016/j.websem.2013.01.002

Please contact:
Miguel A. Martínez-Prieto
University of Valladolid, Spain
E-mail: migumar2@infor.uva.es

Carlos E. Cuesta
Rey Juan Carlos University, Spain
E-mail: carlos.cuesta@urjc.es

Javier D. Fernández
University of Valladolid, Spain
E-mail: jfergar@infor.uva.es

Mario Arias
INSIGHT @ National University of Ireland Galway
E-mail: mario.arias@deri.org