From: **Kasperovics, Romans** romans.kasperovics@sap.com
Subject: Position Statement for the W3C Workshop on Web Standardization for Graph Data
Date: 11 January 2019 at 12:10
To: group-data-ws-pc@w3.org

Dear workshop organizers,

Please consider the following text as my position statement. I am planning to expand it into a presentation with more details and examples for the workshop. I tried to follow the recommended format, however, please let me know if miss something.

Thank you and best regards,

Romans Kasperovics
Developer, Graph Engine, HANA Platform Core
SAP SE Dietmar-Hopp-Allee 16, 69190 Walldorf, Germany
T +49 6227 7-67066, M +4915117195538, E romans.kasperovics@sap.com

Having research background in database and information systems and professional experience in implementing business software, I am enjoying being a part of the SAP HANA development team. My current tasks include extending the SAP HANA database with graph data management features. I am interested in graph query and update languages and their efficient implementations, and I am also representing SAP in the ISO/IEC JTC 1/SC 32/WG 3 (INCITS DM32.2) standardization ad hoc group on the SQL extensions for property graphs.

In this workshop, I would like to talk about graph data models, graph query languages, with an emphasis on schema and closedness. While many natural phenomena can be modeled with schema-free graphs, many graph datasets used in business applications build-up on tabular data that come with schema information. Therefore, I would argue for the importance of schema support both in graph databases and graph query languages.

The relational algebra provides an elegant mathematical foundation for the SQL. One of its key features is the composability or closedness: any relational expression evaluates into a relation. While there were several attempts to introduce a graph algebra or a graph query language closed on graphs [1,2], I would argue for a formal foundation having graphs and relations as base types. Such a foundation would resolve some closedness issues of graph query languages and make the tabular data manipulation, which is already present in many real-world applications, a part of the graph data manipulation language. It will also allow to interpret an arbitrary relational schema as a graph [3], thus enabling a dual graph/relational view on the same data without requiring a major model transformation effort, as suggested in [4].

References:
[1] Renzo Angles et al.: G-CORE: A Core for Future Graph Query Languages. SIGMOD Conference 2018: 1421-1432
[2] Huahai He, Ambuj K. Singh: Graphs-at-a-time: query language and access methods for graph databases. SIGMOD Conference 2008: 405-418
[3] Benjamin A. Steer et al.: Cytosm: Declarative Property Graph Queries Without Data Migration. GRADES@SIGMOD/PODS 2017: 4:1-4:6
[4] Unpublished NII Shonan Meeting No. 138 vision paper