

[GRADES2014](#)**Graph Data Management Experiences and Systems 2014****Reviews For Paper****Paper ID** 18**Title** Towards a Query-by-Example System for Knowledge Graphs**Masked Reviewer ID:** Assigned\_Reviewer\_3**Review:**

Question	
Overall Rating	Accept
Detailed Comments	<p>The article describes the system GQBE, a framework to query knowledge graphs by example tuples of entities (instead of typical complex structured queries). The authors present the architecture of the system by describing its main components, but putting a major interest on the techniques for query graph discovery and query processing. Their current research is also presented, setting new approaches for query suggestion, distributed query processing and the use of user feedback to improve the precision of the answers. Additionally, experiments for measuring accuracy and efficiency are presented.</p> <p>The paper is well-written and discuss several interesting ideas and problems. However, I feel that the content of the paper is unbalanced in the sense that some sections contain simple descriptions easy to read and other section are formal descriptions more difficult to understand (e.g., sections 3.2 and 4.1). Assuming that the authors decided to include more detail in some section due to its importance, I recommend the addition of examples to support their comprehension.</p> <p>In any case, I think the paper fits well with the focus of GRADES and it will be of interest for the audience.</p>

**Masked Reviewer ID:** Assigned\_Reviewer\_4**Review:**

Question	
Overall Rating	Accept
Detailed Comments	<p>This is a very useful paper highlighting potential of query by example on large knowledge graphs. The draft suffers from a number of editorial errors, that while they do not distract from the overall value of the paper, that should be corrected. Some suggested edits:</p> <p>* Simple examples of end-user query (pseudo query examples) and resulting answer tuples would be desirable in the introductory section. Perhaps a small table, with each line illustrating the refinement process?</p> <p>* Some examples of iterative query suggestion would be helpful, building on the Silicon Valley tech company knowledge graph.</p>

**Masked Reviewer ID:** Assigned\_Reviewer\_5**Review:**

Question	
Overall Rating	Neutral
	The paper discusses an approach towards improving the usability of

Detailed  
Comments

knowledge graphs and is based on the use of entity tuples in order to formulate in a semi-automatic way queries that reflect the user's intent. The GQBE (Graph Query By Example) system automatically derives a query graph based on input query tuples. It also finds approximate matching answer graphs and performs a top-k ranking of the answer tuples. An interesting aspect of the system is that it also gives the possibility to the user to refine the answers and the system takes as input this information in order to reformulate the query graph.

The paper is well written and provides an approach complementary to the ones that have been examined recently in the context of RDF data management where the users formulate a text query describing their intent, and the system tries to formulate a SPARQL query from the input text, these approaches are based on natural language and text mining techniques.

Some minor comments: 1) Referring to the unimportant edges example you point to the edges e2 and e3 in Figure 1. There is a problem with the naming of the edges here, please correct. 2) please explain the heuristic that you use for computing the maximal query graph (Section 3.1). 3) in Section 4 you say that "The MQG obtained there tries to identify edges that are important based on the data graph statistics and not on user-based statistics of the edges". Could you please explain what "user-based statistics of the edges" mean? 4) in Section 4.1 when you discuss the workload that you could use to test your system: it might be a good idea to consider any IR benchmark like TREC or INEX which is closer to your use case scenario. 5) regarding Section 4.2: I am not sure as to what is the challenge in this context since there is a lot of work that has been done in distributed query processing and I am not sure what additional challenges the problem and the solution that you propose introduce here.