

Summer of Code 2013

Apache Giraph Google Summer of Code 2013 Ideas Page

Please see the directories below for this years proposals.

Each project proposal should be annotated as fully and in as relevant a manner as possible, clearly defining the project scope, aim and objectives.

- [Apache Giraph Google Summer of Code 2013 Ideas Page](#)
 - [1.Project: Giraph implementation of Nutch LinkRank Algorithm](#)
 - [Model](#)
 - [Project Aim](#)
 - [Project Objectives](#)
 - [Project Scope](#)
 - [References](#)
 - [Proposals](#)
 - [2.Project: Giraph integration with Tinkerpop](#)
 - [Project Aim](#)
 - [Project Objectives](#)
 - [Project Scope](#)
 - [References](#)
 - [Proposals](#)
 - [3.Project: Remove maven-plugin from Giraph](#)
 - [Project Aim](#)
 - [Project Objectives](#)
 - [Project Scope](#)
 - [References](#)
 - [Proposals](#)
 - [How to apply](#)

[Apache Giraph Google Summer of Code 2013 Ideas Page](#) | [1.Project: Giraph implementation of Nutch LinkRank Algorithm](#) | [Model](#) | [Project Aim](#) | [Project Objectives](#) | [Project Scope](#) | [References](#) | [Proposals](#) | [2.Project: Giraph integration with Tinkerpop](#) | [Project Aim](#) | [Project Objectives](#) | [Project Scope](#) | [References](#) | [Proposals](#) | [3.Project: Remove maven-plugin from Giraph](#) | [Project Aim](#) | [Project Objectives](#) | [Project Scope](#) | [References](#) | [Proposals](#) | [How to apply](#)

1.Project: Giraph implementation of Nutch LinkRank Algorithm

Model

Project Aim

Project Objectives

Project Scope

References

Proposals

<https://cwiki.apache.org/confluence/display/GIRAPH/GSoC+2013++Giraph+Implementation+of+Nutch+LinkRank+Algorithm>

[Ahmet Emre Alada](#)

2.Project: Giraph integration with Tinkerpop

This project relates to the respective JIRA: <https://issues.apache.org/jira/browse/GIRAPH-549>

Project Aim

Graph databases are dbms that are very efficient to run graph traversals to answer queries. Typical examples are Neo4j, OrientDB, Dex, etc. Graph databases and Giraph solve two different problems in the graph processing world. The first ones solve needs for queries that touch a small portion of the graph, and that need to be answered with low latency (milliseconds). The second one solves the need for large computations that touch all the graph, possibly many times, hence potentially lasting for hours. Both tools are not good in solving the problem the other one is good at.

However they can work together. By injecting the graph from a graph database into Giraph, it is possible to run analytics that cannot be run on the graph database itself, as much as MapReduce can be used to run OLAP queries on data stored in a standard RDBMS.

Project Objectives

The aim of the project is to integrate graph databases as inputs for Giraph, along with the current ones (e.g. HBase, Accumulo etc.). In particular, use Tinkerpop with Giraph as a way to inject data into Giraph for graph analytics. Precisely, to leverage Blueprints and Rexter, that build a vendor-agnostic API over the vendor-specific ones, and export it via a REST API.

Project Scope

The project scope is divided in three main milestones:

- Integrate Rexter into Giraph inputs, e.g. leveraging existing classes in Faunus for MapReduce
- Define a mapping between the flexible Property Graph data model to Giraph's data model
- Allow to specify the input graph from the graph database to Giraph via a query (not only support injecting the whole graph)

For very successful students, there are possibilities to extend the work with more deliverables:

- Support Titan as an input
- Support GraphSON as an InputFormat

References

Tinkerpop: <http://www.tinkerpop.com/>

Blueprints: <https://github.com/tinkerpop/blueprints/wiki>

Rexter: <https://github.com/tinkerpop/rexster/wiki>

Faunus: <http://thinkarelius.github.io/faunus/>

Titan: <http://thinkarelius.github.io/titan/>

GraphSON: <https://github.com/thinkarelius/faunus/wiki/GraphSON-Format>

Neo4j: <http://www.neo4j.org/>

OrientDB: <http://www.orientdb.org/>

Proposals

3.Project: Remove maven-plugin from Giraph

This project relates to the respective JIRA: <https://issues.apache.org/jira/browse/GIRAPH-101>

Project Aim

Munge is a hacky way of support multiple versions of Hadoop. The shim layers in Pig and Hive could be a cleaner way to do this.

Project Objectives

Project Scope

References

Maven-munge: <http://sonatype.github.io/munge-maven-plugin/>

Proposals

How to apply

Please read GSoC guide for student to apply. It is highly recommend to discuss your interest before you apply. The best way to discuss is to comment on individual Jira or send mail to dev list.