

# A workflow visualisation model for Pegasus WMS

Michał Matusiak, Mateusz Woś, Włodzimierz Funika

AGH, Faculty of Computer Science, Electronics and  
Telecommunication,

Department. of Computer Science

al. Mickiewicza 30, 30-059, Kraków, Poland,

emails: {mmatusia,wos}@student.agh.edu.pl, funika@agh.edu.pl



# Motivation

- **Pegasus WMS: a framework for executing scientific workflows**
- **Lack of built-in visualisation facility in Pegasus WMS.**
- **Monitoring what is going on with workflow execution**

# Objectives

- **Clear, automated workflow visualisation.**
- **Simple settings modification.**
- **Insight into resources usage while workflow execution.**

# Poster No. 7

We invite you to attend our poster for more details on the research and proposed solution.

### A workflow visualisation model for Pegasus WMS

Michał Matusiak, Mateusz Woś, Włodzimierz Funika  
AGH, Faculty of Computer Science, Electronics and Telecommunication, Dept. of Computer Science,  
al. Mickiewicza 30, 30-059, Kraków, Poland,  
emails: mmatusia@student.agh.edu.pl, wos@student.agh.edu.pl, funika@agh.edu.pl

#### Introduction

Pegasus WMS is system designed to manage scientific workflows. Unfortunately it comes without a good graphical visualisation solution. In our work we propose a prototype of a possible solution. We have decided to use directed acyclic graphs to represent workflows as that is how they are represented in Pegasus WMS.

#### Main goals

- clear workflow visualisation
- simple settings modification
- insight into resources usage

#### Visualisation

- Jobs are represented by graph nodes
- Dependencies between jobs are represented by graph edges
- Data is displayed inside nodes

#### Application flow

With usage of Python, Numpy library and regular expressions we extract data and create necessary structures. Pandas logs are used to connect information about jobs and used resources by the job name. Job names are assigned by Pegasus WMS.

Having all crucial data we generate directed acyclic graph structure using graph-tool library which also allow us to generate image of defined graph.

#### Encountered Problems

During the implementation we have encountered major problem. Our goal was to automatize visualisation, but many graph libraries offered visualisation without automated nodes arrangement. Main objective was that the displayed graph should be clearly visible without any need of manual user manipulation therefore nodes should be of appropriate size and properly placed that as little as possible edges would cross each other. Fortunately we found this library called Graph-tool that uses sophisticated algorithms to solve this issue.

#### Summary

The outcome of the research is a fully functional prototype. Graphs generated by application gives an accurate reflection of a workflow, connections between jobs and consumed resources, but it surely requires further work. Text formatting needs improvement, as current solution results in unexpected increase in nodes size, making visualisation a lot less clear. Visualisation of bigger workflows should be extended. It could be realised by defining central node and a limiting graph radius. Moreover, future work aims to develop "online" version of visualization. Online mode will allow the user to observe application data flow in real time.

#### References

Pegasus website: <http://pegasus.isi.edu>  
Graph-tool website: <https://graph-tool.skewed.de>  
[https://github.com/mwoos/PegasusWMS\\_workflow](https://github.com/mwoos/PegasusWMS_workflow)

#### Acknowledgments

This research received financial support from AGH University of Science and Technology Statutory Project. This research was supported in part by PLGrid Infrastructure.