

# Assessing and Evaluating Documentation in Libre Software Projects

Gregorio Robles, Jesús González  
Barahona, Juan Luis Prieto  
[grex@gsyc.escet.urjc.es](mailto:grex@gsyc.escet.urjc.es)

<http://libresoft.urjc.es>

Universidad Rey Juan Carlos

Como (Italy), June 10<sup>th</sup> 2006

# Introduction

- Assessing documentation in “traditional” software development:
  - Specific resources, specific tasks
  - Evaluation of quality and design, of the formats used
- This is not the case in libre software projects:
  - We need to identify if there is documentation at all!
- Identifying doc is very time-consuming as a manual task
  - Automate documentation identifying

# Requirements

- Given a tarball, it has to identify the (developer and end-user) documentation it contains
  - Common documentation files, locations and types
    - README, ChangeLog, TODO, CREDITS, INSTALL, AUTHORS...
    - doc/ subdirectory...
    - PDFs, HTMLs, ...
  - In-line documentation (javadoc and the like)
  - Comments embedded in source code
- It has to provide the results in a report (with 3 layers)

# Three-layer report

## **Report for Project X**

**Management Summary**

**Summarized Report Indicators**

**Detailed Report**

# Case studies

- Selected random sample of 8 (medium-sized) projects
  - Manual inspection: approx. two and a half hours
- With the help of the tool, we have similar (in fact, richer) results in 20 minutes
  - Reduction of 80% of the time spent
  - Results are more accurate and comparable
- In addition, we can study the documentation over time

# Summary & further work

- We propose a methodology (and a tool that implements it) to quantitatively identify documentation in software projects
- Our approach lowers the time spent in assessing if a libre software project is documented
- Three-layer reports make it possible to satisfy different type of users
- Quality metrics should be included (probably, as we can find them in “traditional” software development)