Maestro

A language for job scheduling
Team Members

Vaggelis Atlidakis
Georgios Koloventzos
Mathias Lecuyer
Arun Swaminathan
Yiren Lu

TA: Junde Huang
Agenda

❖ What is Maestro?
❖ Why Maestro?
❖ Who uses Maestro?
❖ Example # 1
❖ Syntax Explanation
❖ Example # 2
❖ Syntax Explanation
❖ System Architecture
❖ Testing
❖ Demo
❖ Lessons
❖ Thank you!!
What is Maestro?

Declarative, Interpreted Scripting Language for Job Scheduling

Dynamically Typed

Powerful Semantics for Job Distribution
Why Maestro?

Consider an experiment that is divided into 3 consecutive steps. Maestro can help:

- Express each step with a script and define a Maestro Job.
- Express the correlation of steps using Maestro Job Dependencies.
- Execute each step only after its dependencies are Resolved.
Who Uses Maestro?

❖ Research Labs like CERN that run large-scale distributed jobs
❖ Academics running thousands of experiments on a strict timetable
❖ Anyone who is tired of hand-holding a script through a conditional pipeline
master("systems-yellow.cs.columbia.edu:6379");

a = Job("print.rb", "Hello");
b = Job("print.rb", "World");
c = Job("print.rb", "!");
run(a -> b -> Wait(10) -> c);
Example # 1 (Hello World)

master("systems-yellow.cs.columbia.edu:6379");
a = Job("print.rb", "Hello");
b = Job("print.rb", "World");
c = Job("print.rb", "!");
run(a -> b -> Wait(10) -> c);
Example # 1 (Hello World)

master("systems-yellow.cs.columbia.edu:6379");
a = Job("print.rb", "Hello");
b = Job("print.rb", "World");
c = Job("print.rb", "!");
run(a -> b -> Wait(10) -> c);
Example # 2 (MapReduce)

master("systems-yellow.cs.columbia.edu:6379");
a = Job("split.rb", "/tmp/big_file_name.data");
maps = map(a, "map.rb");
red = reduce(maps, "reduce.rb");
run(red);
Example # 2 (MapReduce)

master("systems-yellow.cs.columbia.edu:6379");
a = Job("split.rb", "/tmp/big_file_name.data");
maps = map(a, "map.rb");
red = reduce(maps, "reduce.rb");
run(red);
Example # 2 (MapReduce)

master("systems-yellow.cs.columbia.edu:6379");
a = Job("split.rb", "/tmp/big_file_name.data");
maps = map(a, "map.rb");  
red = reduce(maps, "reduce.rb");
run(red);
Example # 2 (MapReduce)

master("systems-yellow.cs.columbia.edu:6379");
a = Job("split.rb", "/tmp/big_file_name.data");
maps = map(a, "map.rb");
red = reduce(maps, "reduce.rb"); Reduce
run(red);
Example # 2 (MapReduce)

master("systems-yellow.cs.columbia.edu:6379");
a = Job("split.rb", "/tmp/b...
System Architecture

- **Symbol Table** (Python Dictionary)
  - Input: Program from File, Line from REPL
  - Outputs: Tokens, AST

- **Lexical Analysis** (Python Lex)
  - Inputs: Program from File, Line from REPL
  - Outputs: Tokens

- **Syntax Analysis** (Python Yacc)
  - Inputs: Tokens, AST
  - Outputs: AST

- **Semantic Analysis**
  - Inputs: AST
  - Outputs: AST

- **Translator**
  - Inputs: AST

**Master Runtime** (python + maestro specific code)
- API
- Global Priority Queue of waiting Jobs

**Worker Runtime** (python + maestro specific code)
- Redis Pub/Sub
Testing

- Custom test engine -no library or module used
- One framework for testing all parts of the program
- Tests not only the execution of Maestro, but also the execution of the job sent to Maestro
- Supports individual or batch testing
- Logs test results for deeper analysis and to help locate/fix errors quickly
Testing

Batch tests

Test log

---

Tests Passed: 15 / 15
Check log for test details

dyn-200-2-211-141:src rameshswaminathans
Demo

MY PROGRAM WORKS PERFECT

EXCEPT WHEN I DO A DEMO
Lessons Learned

❖ Start early
❖ Pick scope of project wisely
❖ Constantly reprioritize
❖ Integrate continuously and often
❖ Modularize intelligently
Thank you!!