Workflow Management for User Analyses in Particle Physics

Robert Fischer
Ralf Florian von Cube, Martin Erdmann, Benjamin Fischer, Marcel Rieger

ACAT 2017, Seattle
Overview

- Motivation for workflow management solutions
- Existing workflows in HEP
- Two Approaches
- Experience
- Consequences
Landscape of HEP Analyses

- Increasing scale and complexity
- Undocumented dependencies between workloads, only exist in the physicist’s head
- Bookkeeping of data, revisions, …
- Manual execution / steering of jobs
- Error-prone & time-consuming

→ Analysis workflow management essential for future measurements
Wishlist

- Reproducible intermediate and final results
- Adaptable, e.g. during review process, new recipes
- Collaborative development and processing
- Arbitrary programming language

- `make`-like distributed execution
- Opportunistic: run and storage locations
- Automatisation: bookkeeping, data retrieval, dependency resolution, etc.

- Steering in Python
Abstraction:
Particle Physics Analysis

- Workflow comprises smaller workloads
- Workloads related to each other by common interface
- Computing resources
  - Run location (CPU, GPU, grid, …)
  - Storage location (local, dCache, eos, …)
- Software environment

→ Large overlap with Workflow Management Solutions
Existing Workflows: MC Production

- Workflows are static, one-dimensional, recurring
- Homogeneous software requirements
- Special infrastructures: Databases, storage, workload management system

→ Static workflows not flexible enough for user analyses
Two Approaches

Report Based
• Tailored for HEP from scratch
• Store report file for each execution, evaluate reports of predecessors
• Should perform well if storage is slow

Target Based
• Which community tools can be adopted?
• Check for output target of predecessors before execution
• Should reduce complexity if storage is fast
Report Based Approach

Scheduler

Python analysis config

- step 1
- step ...
- step n

one step

command line interface

tasks

Metadata Storage Location

- task reports

Run Location

- task reports

- pre-compiled software packages

- on-demand data retrieval

Binary Storage Location

Data Storage Location

user

8
Real Workflow

ttbb cross section measurement
Real Workflow

ttbb cross section measurement
Target Based Approach

- Luigi: Package for building complex pipelines
- Initially developed at Spotify, now open-source

!!! Diagram

Python analysis config

Task 1
Task ...
Task n

one Task

tasks

command line interface

Run Location

Binary Storage Location

pre-compiled software packages

Data Storage Location

on-demand data retrieval

file target exists?

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source

Target Based Approach

• Luigi: Package for building complex pipelines
• Initially developed at Spotify, now open-source
Luigi Execution Model

- Execution is **make**-like

- Trigger one task
  1. Create dependency tree
  2. Walk down the tree
  3. Run incomplete tasks in *n* workers
Software: Environments and Sandboxes

• Opportunistic
  Define environments using existing software on run location
  (dynamic .bashrc)

• Pro-active
  Build and install user software (locally or on run location)

• Sandboxing
  Coherent isolated environment

• Future
  • Docker / containers
  • Lightweight virtual machines
Achievements

• Both approaches proofed with real data analyses
  ttbb and ttH cross section measurements

• Toolbox providing building blocks for analyses, not a software framework
  → Design pattern

• All inputs and parameters transparently encoded
  → Reproducibility

• make-like execution across distributed resources demonstrated
  → Reduces overhead
  → Focus on physics

Changed paradigm from Executing to Defining Analysis
Collaboration

→ Actually work on same analysis across groups
Analysis Preservation

- DPHEP: *Tools and best practices for "adding value" to data*
  
  ![DPHEP Logo]
  

- HEPData: *open-access repository for scattering data from experimental particle physics*
  
  ![HEPData Logo]
  
  [https://hepdata.net](https://hepdata.net)
Take Away

• Trend to larger, more complex analyses continues

• Workflow systems help to manage analyses

• Additional benefits
  • Foster collaboration among analysts
  • Gateway to analysis preservation
Example Application: ttH Analysis

- Large-scale:
  - ~50k files, ~50 TB of storage, ~1k unique tasks
- Complex:
  - ~40 systematic variations, DNNs/BDTs/MEM, multiple categorization schemes
- Run locations:
  - 7 CEs, local machines, GPU machines
- Storage locations:
  - 2 SEs (dCache), local disk, Dropbox, CERNBox
- Aware of entire workflow at all times, fast dev.
- Clear allocation of duties in group
- Entire analysis operable by everyone at all times

→ Successful proof of usability & suitability
# reco.py

```python
import luigi

from analyses.ttH.tasks import Selection

class Reconstruction(luigi.Task):
    dataset = luigi.Parameter(default="ttH125")

    def requires(self):
        return Selection(dataset=self.dataset)

    def output(self):
        return luigi.LocalTarget("reco_%s.root" % self.dataset)

    def run(self):
        # do whatever a reconstruction does
        ...
```

> python reco.py Reconstruction --dataset ttJets