Characterization of a Computational Grid as a Complex System

Lovro Ilijasic (lovro@di.unito.it)
Lorenza Saitta (saitta@mfn.unipmn.it)

Introduction

- Grid Observatory aims to develop a scientific view of the dynamics of grid behavior and usage.
    - Analysis
    - Models

- Models: to generate test data for simulating a grid in future research, for prediction of oncoming events in order to optimize the scheduling and workload distribution, as well as for detection of outliers, intrusion or other anomalous behaviors in the system.

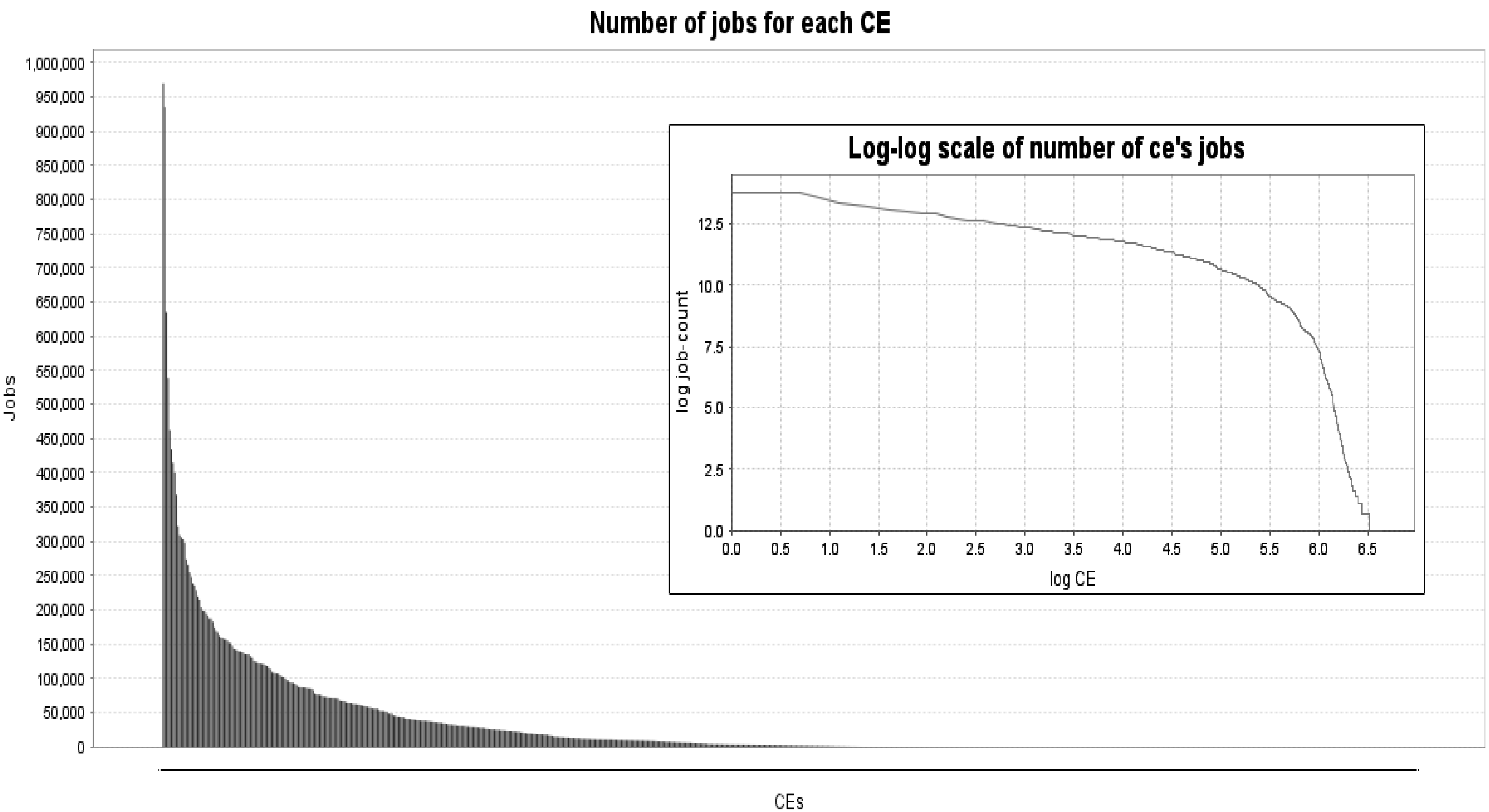
Context

- Global characterization of a grid system
- High level of abstraction - Computational grid is a system of computing elements created and used by humans
- Main objects: Users and Computing Elements (CEs)
- …and jobs that can be seen both as a link between Users and CEs, and as separate objects having their own attributes

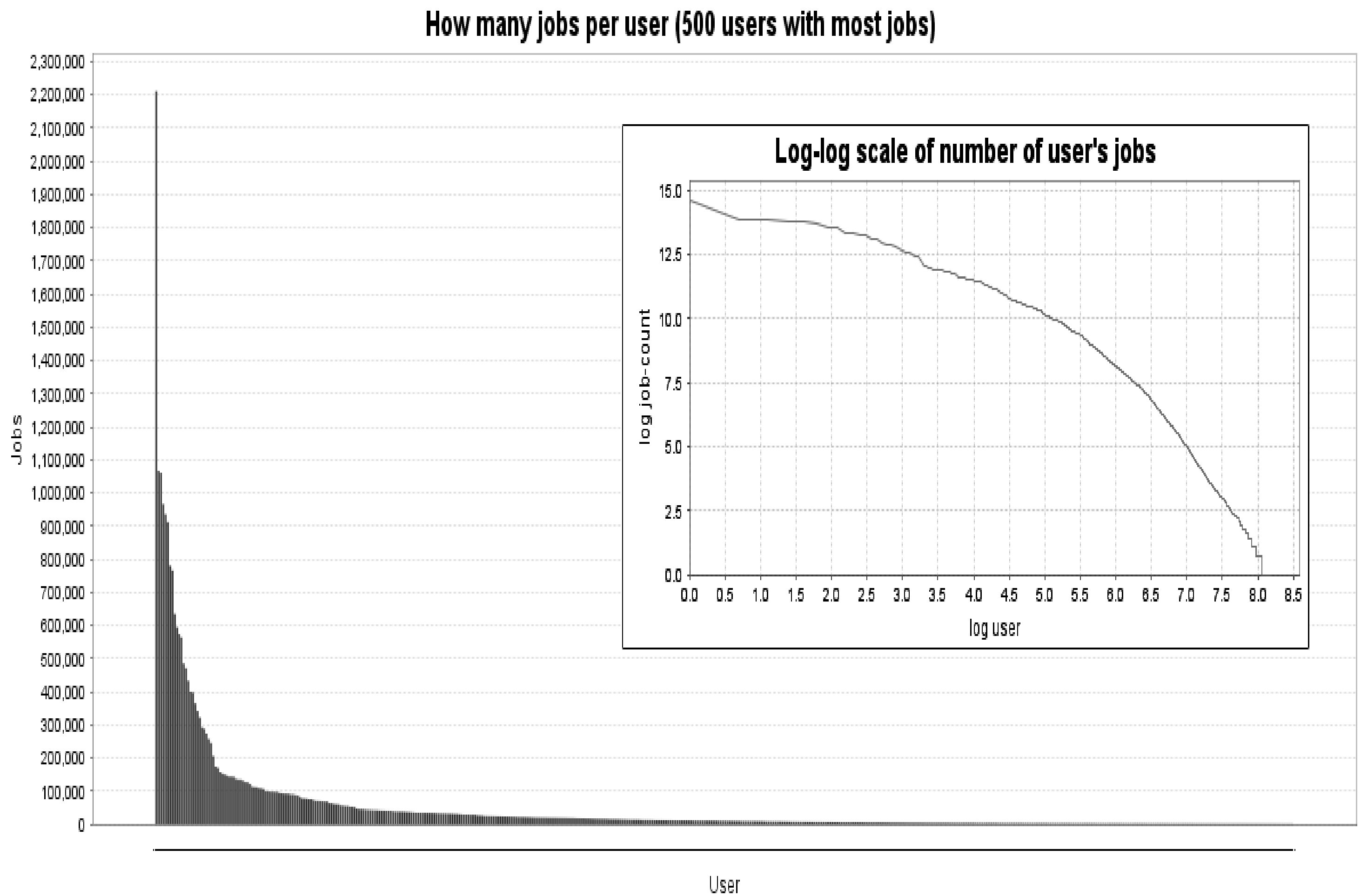
Data

- Dataset is collected from all major Resource Brokers (RBs) by The Real Time Monitor, developed by Imperial College, London.
- Log data was gathered during 20 months period (September 1st, 2005 – April 30th, 2007).
- Each row in the dataset contains the summary of a single job.
- 28,384,971 rows (i.e., jobs)
- 3,529 users
- 760 Computing Elements
- 607 days

About Computing Elements

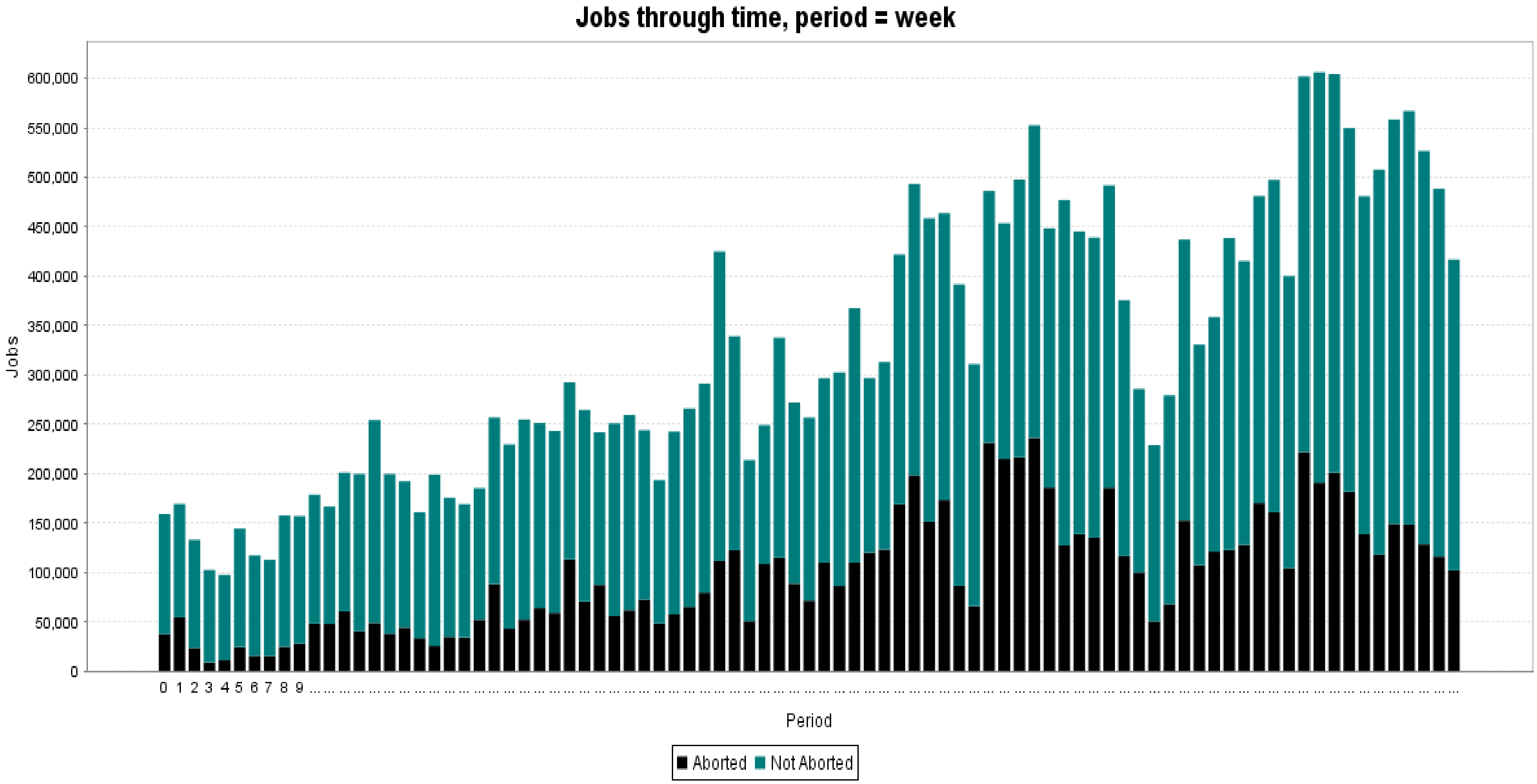


About Users

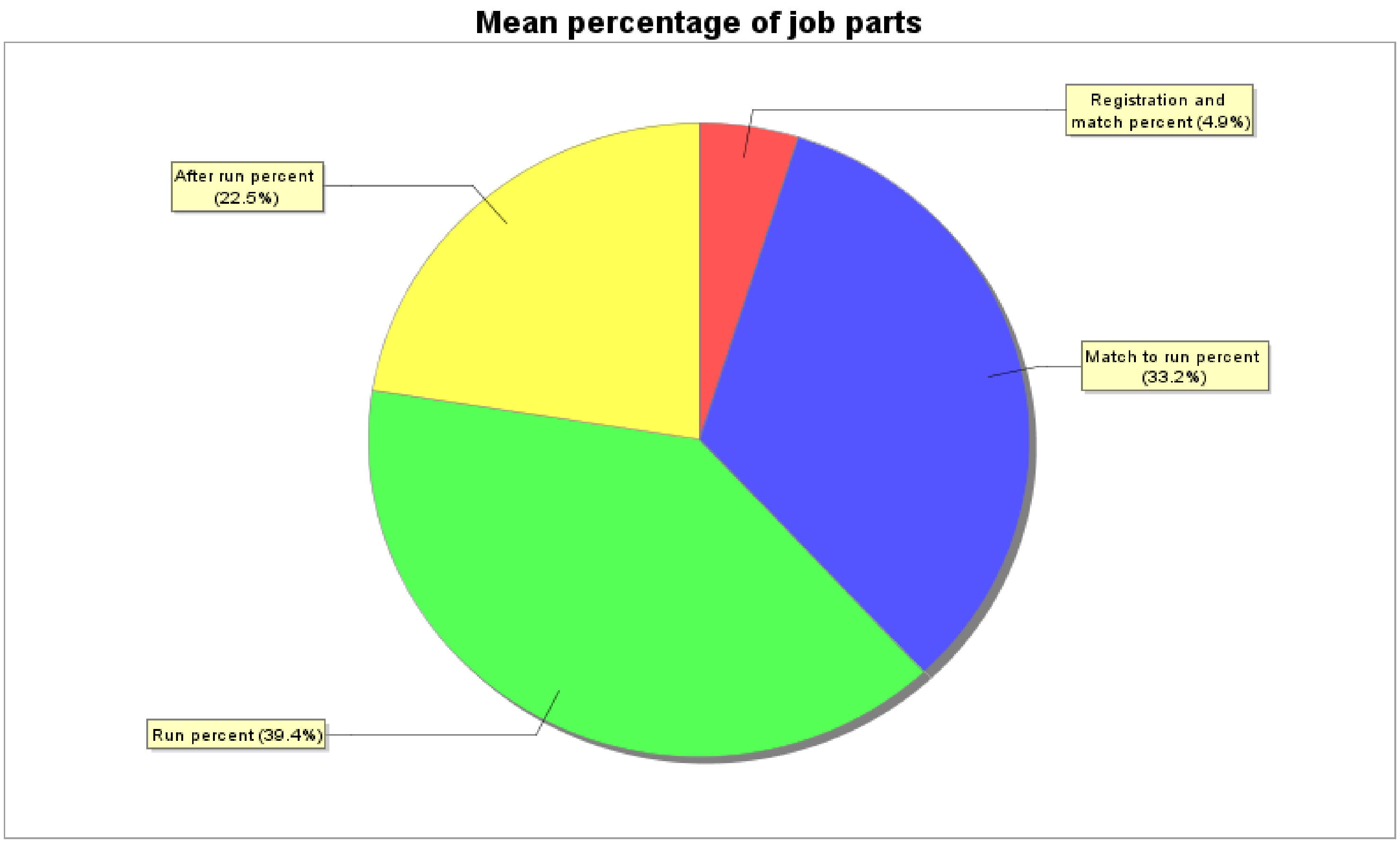

How many jobs per user (500 users with most jobs)
Jobs
2,300,000
2,200,000
2,100,000
2,000,000
1,900,000
1,800,000
1,700,000
1,600,000
1,500,000
1,400,000
1,300,000
1,200,000
1,100,000
1,000,000
900,000
800,000
700,000
600,000
500,000
400,000
300,000
200,000
100,000
0
User
Log-log scale of number of user's jobs
log job-count
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0 6.5 7.0 7.5 8.0 8.5
log user

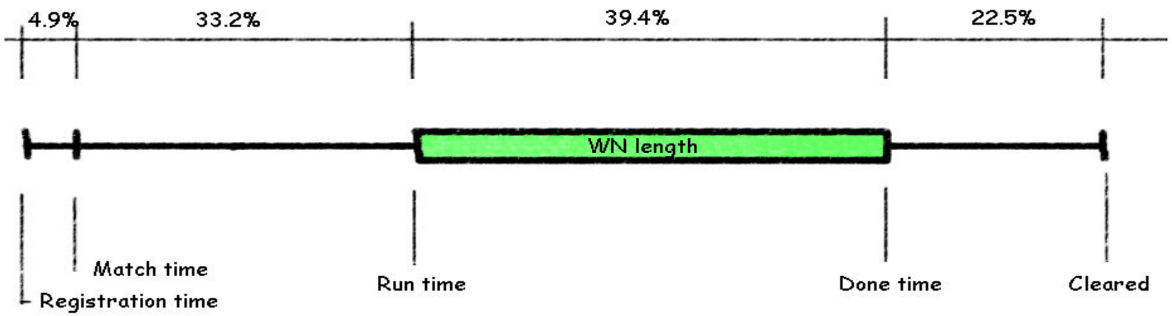
About Jobs

- Distributions, correlations, dynamics… (number of jobs, of job lengths, Abort rate)



Job Lifecycle

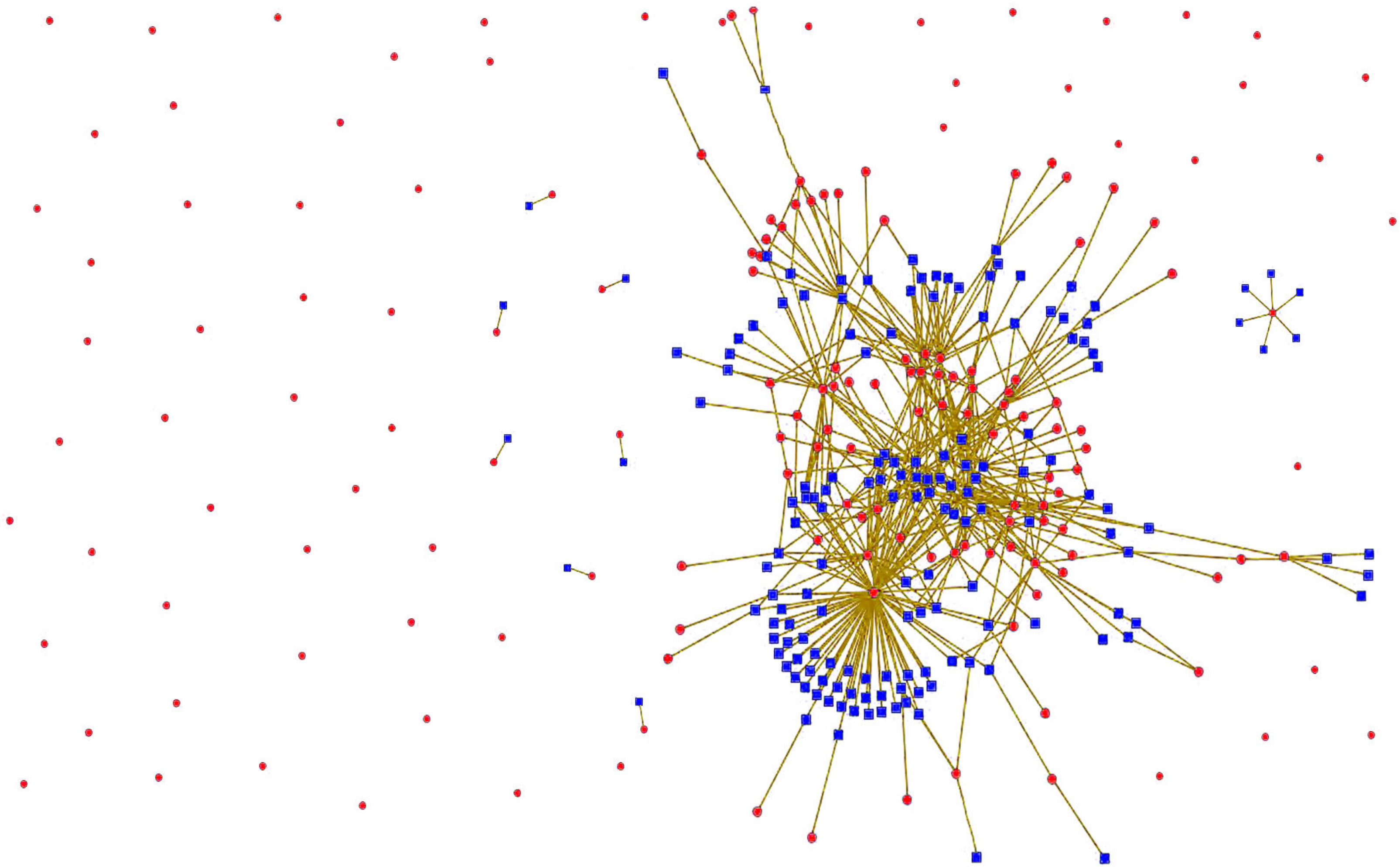




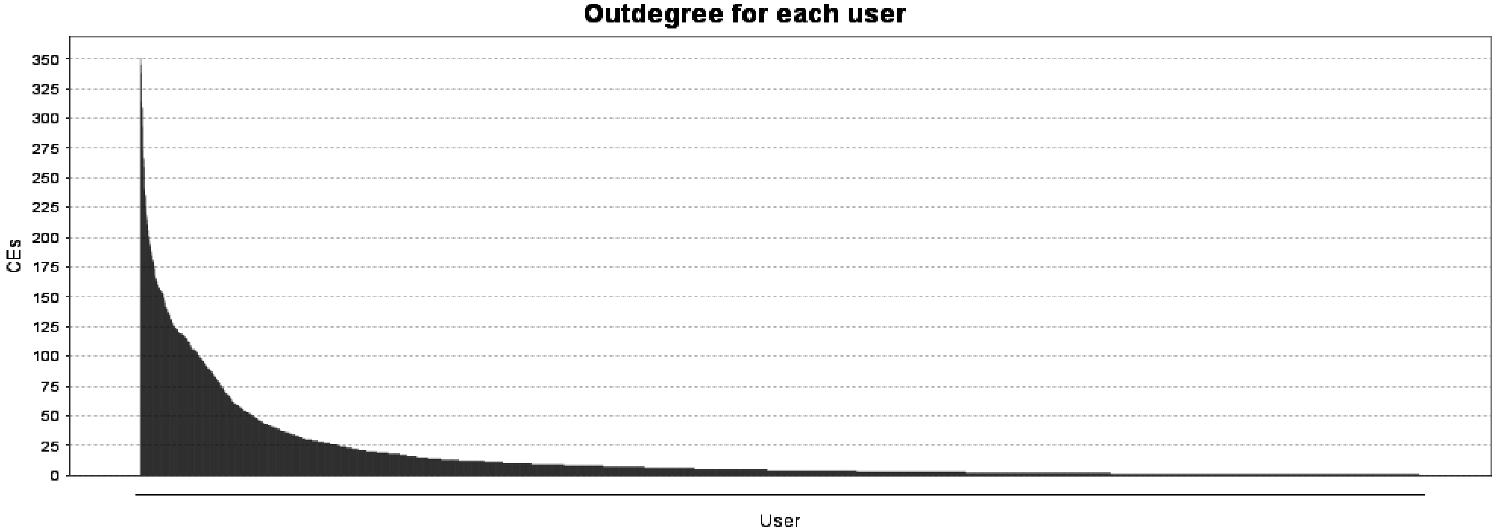
Overall Distribution of Job WN Length

Grid as a Complex Network

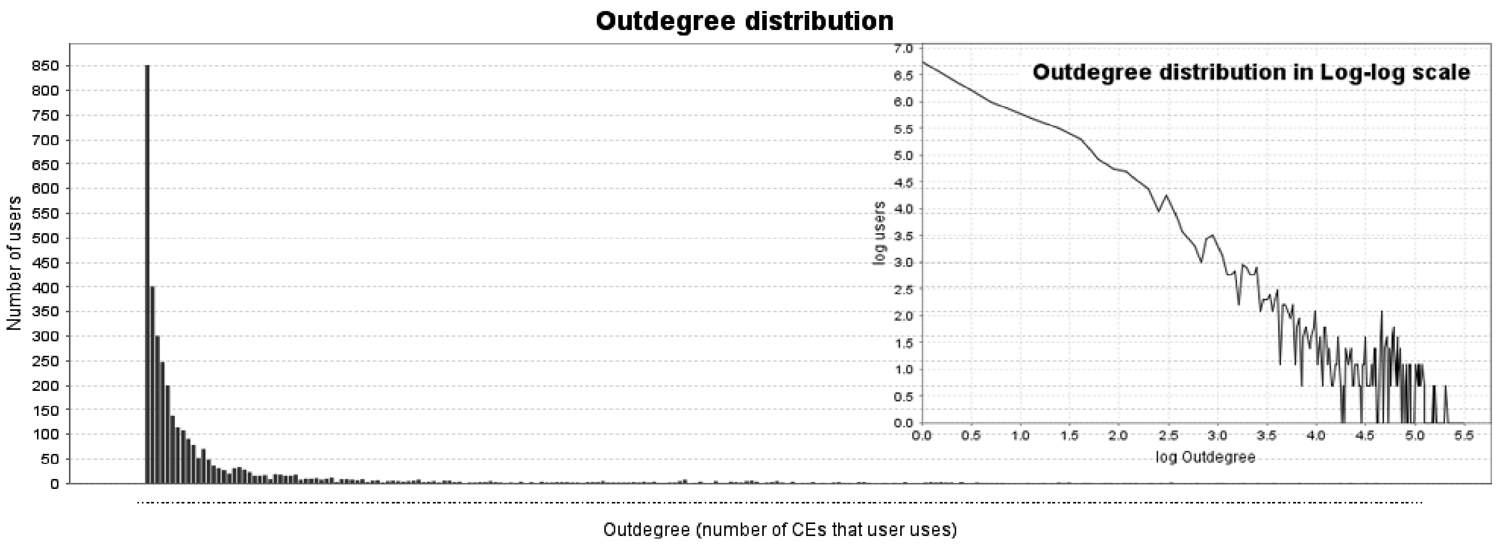
- Grid = Bipartite, directed, weighted graph



Outdegree



Outdegree for each user
CEs
350
325
300
275
250
225
200
175
150
125
100
75
50
25
0
User



Outdegree distribution
Number of users
850
800
750
700
650
600
550
500
450
400
350
300
250
200
150
100
50
0
Outdegree (number of CEs that user uses)
Outdegree distribution in Log-log scale
log users
log Outdegree
7.0
6.5
6.0
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5

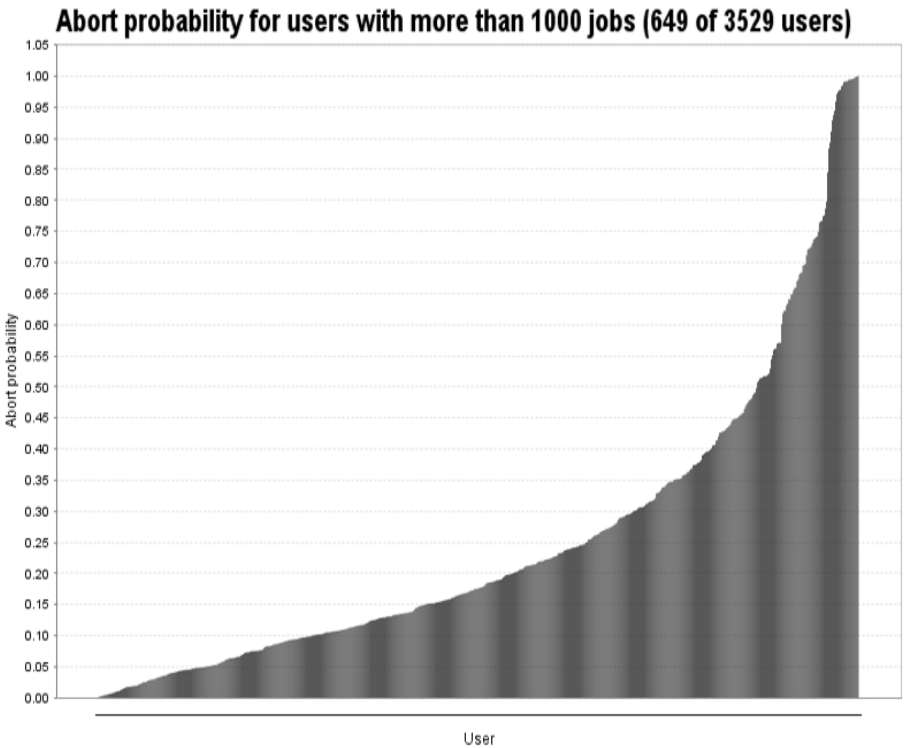
Building Models

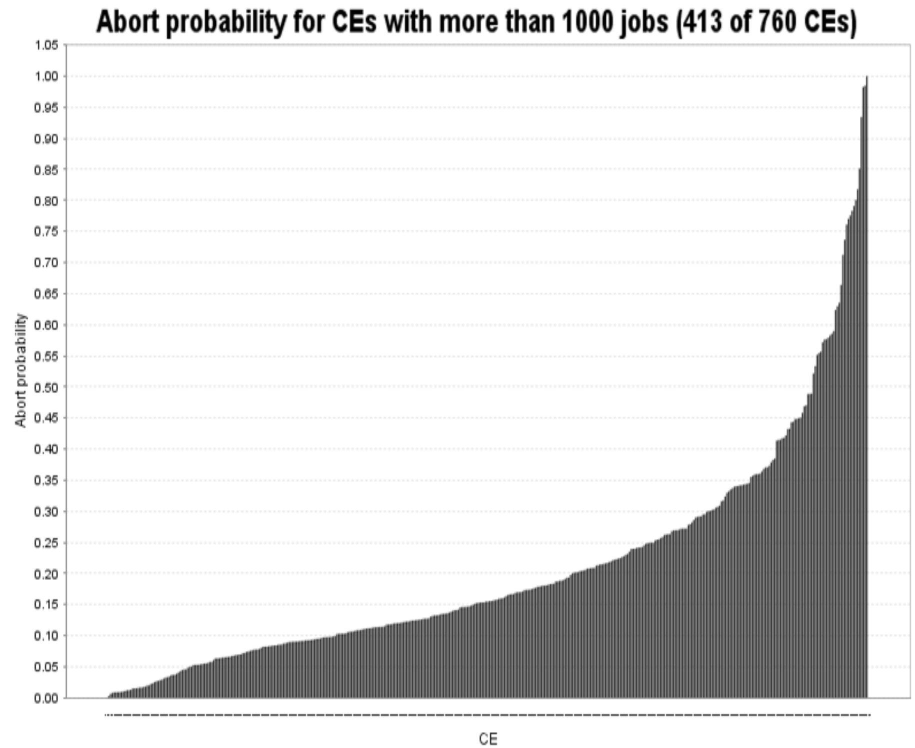
Predicting Job Abortion

- 30.3% of all the jobs in our dataset are aborted (8.6 million jobs)
- Of all the aborted jobs, 38.4% are aborted on Resource Broker, mostly for the "No compatible resources" reason.
- Reasons:
  - 45.7% Job RetryCount hit
  - 31.6% Cannot plan: BrokerHelper: no compatible resources
  - 7.6% Job proxy is expired.
- We are more interested in predicting job abortions on a Computing Element, as this result could be applicable to optimizing schedulers and grid performance.

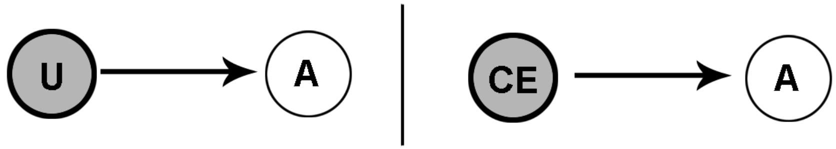
What to Use for Prediction

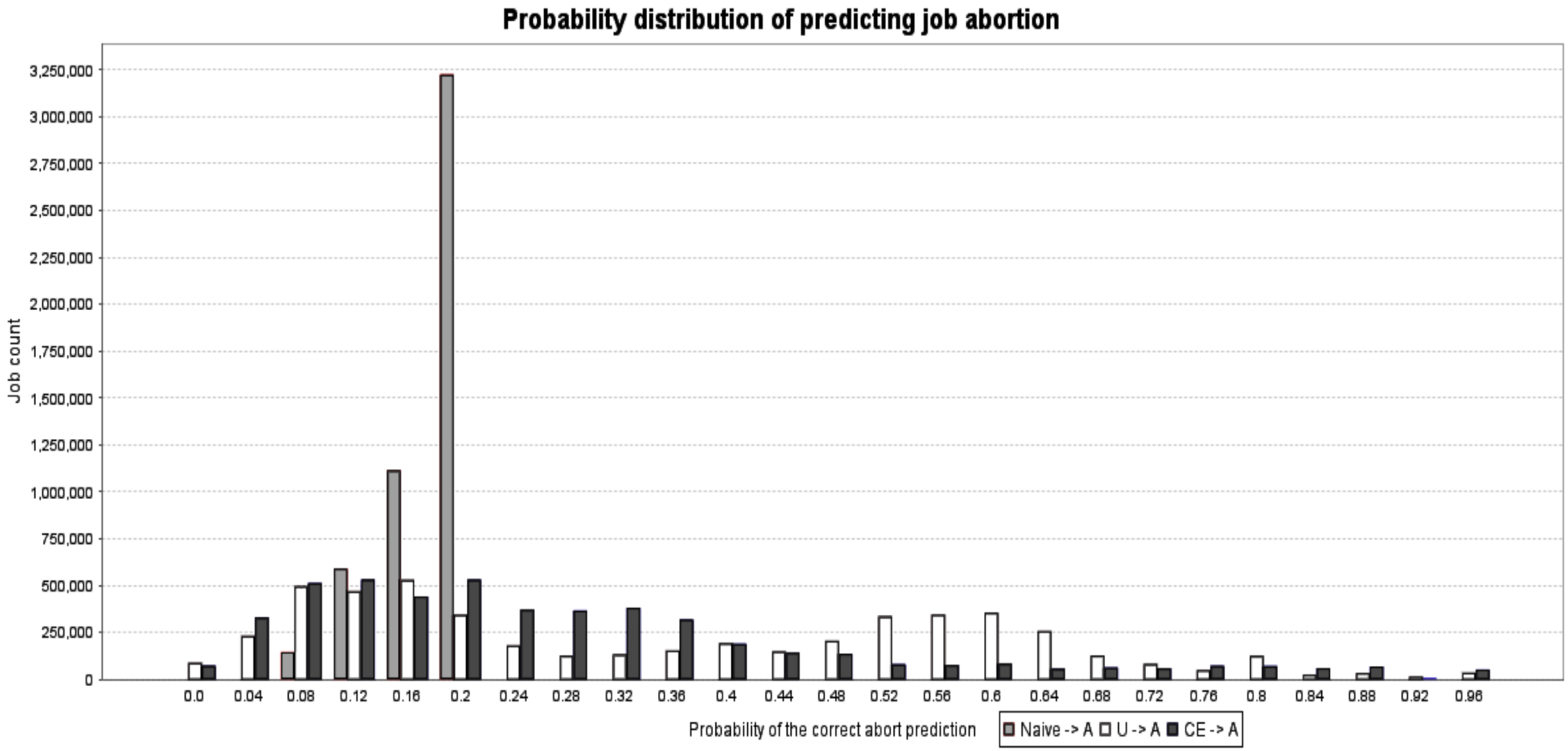
- Some users (CEs) are more prone to have their jobs aborted than the others





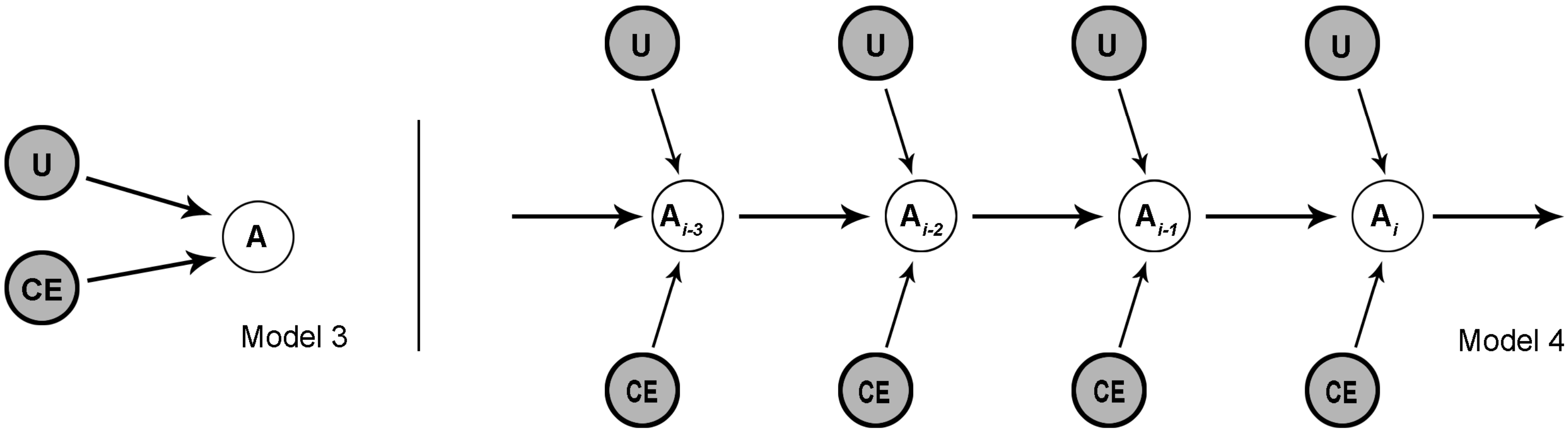
Simple Models



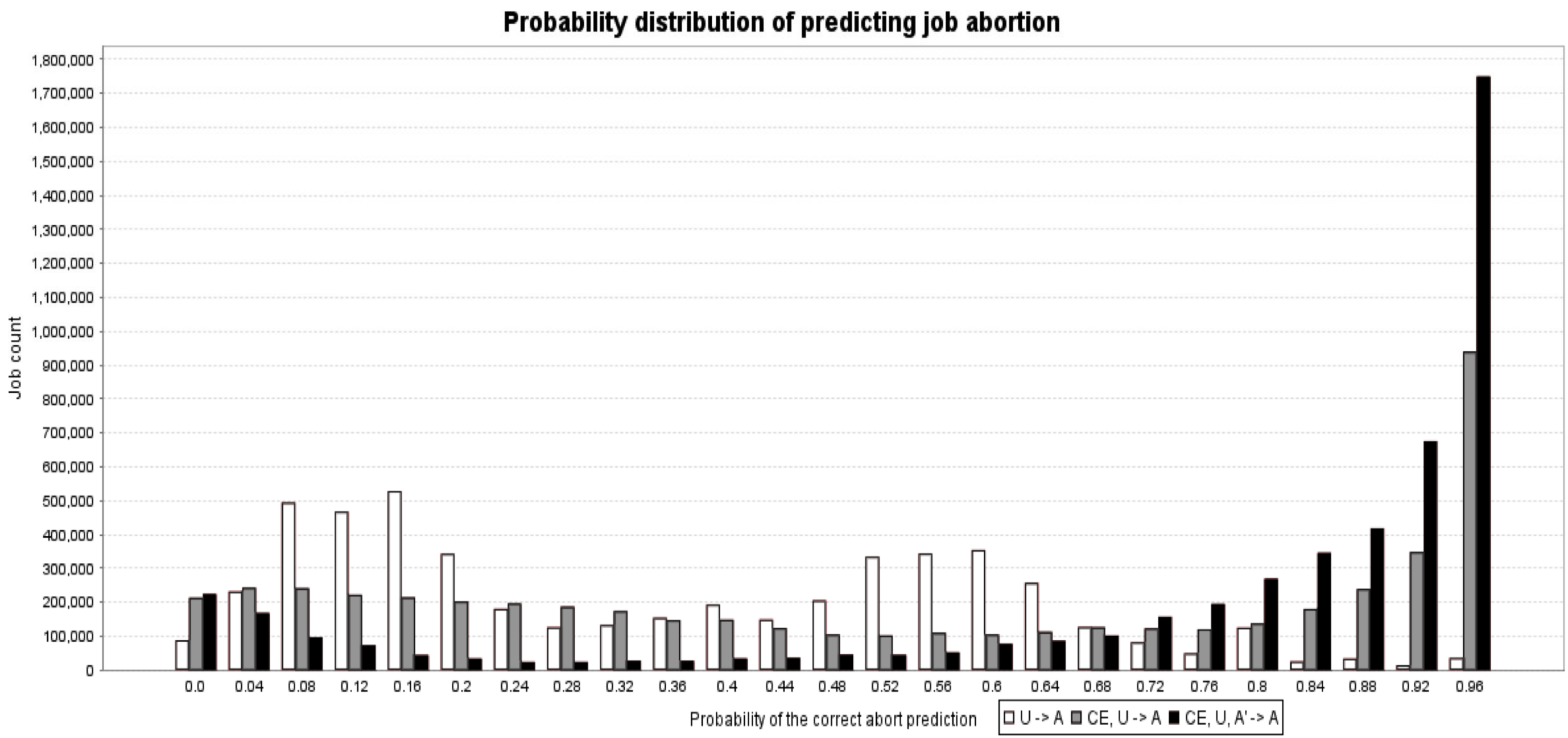


More Elaborate Models

- Users behave differently on different CEs
- User/CE pair
- Dynamic Bayesian Network (Markov property)



The Result: 82% Job Abortions Predicted Successfully



Confusion Matrix

- Confusion matrix for offline prediction of job abortion

	Real aborted (5,078,639 jobs)	Real not aborted (18,998,586 jobs)
Predicted aborted	4,195,674 82.61% of real aborts	557,432 2.93% of real not aborts
Predicted not aborted	861,365 16.97% of real aborts	18,360,638 96.64% of real not aborts

But ...

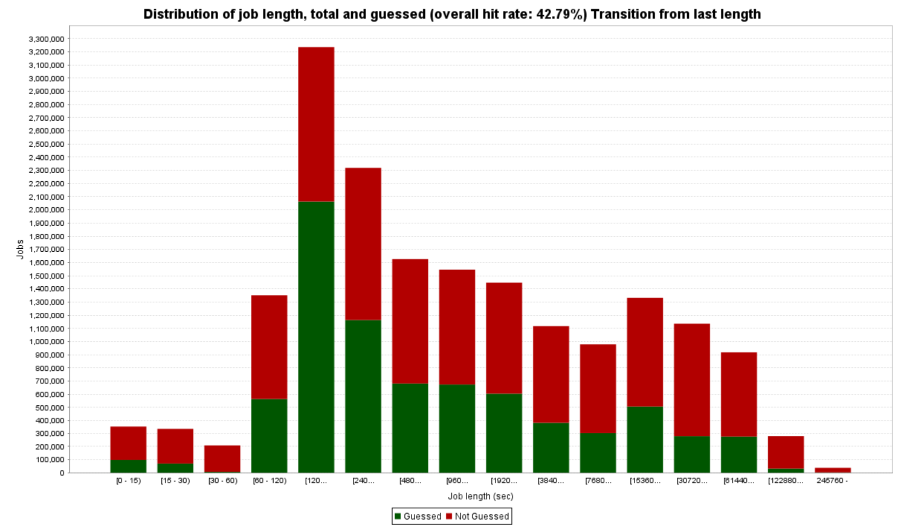
- These are the results of the offline analysis, where we have all the data
- …even the information on the previous job, even if it wasn’t finished
- Useful for investigating the dependencies and for some offline application, but not for scheduling
- We need to build the model online
- …but we don’t have the information on the most recent jobs, which are the most similar ones

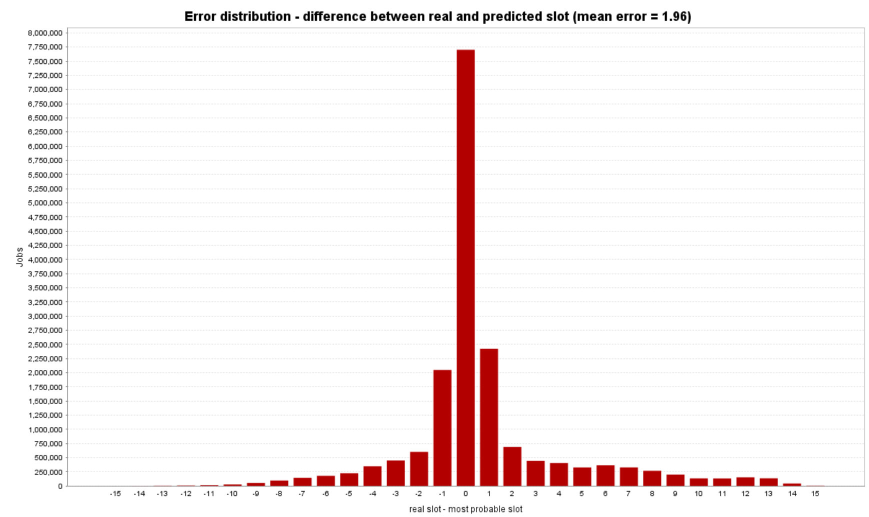
Predicting Job Length

- Using analogous concept – information on user, CE and previous job – to predict the correct time slot
- Time slots are in logarithmic scale. 16 slots, each is twice as wide as the previous one: [0, 15s), [15s, 30s), [30s, 60s), [60s, 120s),…
- Again, offline, we get excellent results using only user, CE and previous length. We predict the correct slot for total length for 70.4% of all jobs, and correct WN (run) length for 64.8%

Predicting Job Length Online

- At registration time
- Using the information on last job that has already ended
- Correct slot for 42.8% of jobs
- Mean error: 1.96 slots





Future Work

- Working on a meta learner which would use the job and model data to predict which model to use
- Building up a simulator for scheduling jobs using the information and model developed so far.