# Progress report

## Neronet

*Toolbox for managing the training neural networks*

CSE-C2610
Software Project

Aalto University

December 4, 2015

Aalto University
School of Science
and Technology

# Outline

# Introduction

## Goals

Our goal is to develop a tool for computational researchers to enable easy

- specification and management of experiment queues

# Introduction

## Goals

Our goal is to develop a tool for computational researchers to enable easy

- specification and management of experiment queues
- batch submission of experiment jobs to computing clusters

# Introduction

## Goals

Our goal is to develop a tool for computational researchers to enable easy

- specification and management of experiment queues
- batch submission of experiment jobs to computing clusters
- monitoring of ongoing experiments' logs and parameter values

# Introduction

## Goals

Our goal is to develop a tool for computational researchers to enable easy

- specification and management of experiment queues
- batch submission of experiment jobs to computing clusters
- monitoring of ongoing experiments' logs and parameter values
- access to experiment information during and after the run

# Introduction

## Goals

Our goal is to develop a tool for computational researchers to enable easy

- specification and management of experiment queues
- batch submission of experiment jobs to computing clusters
- monitoring of ongoing experiments' logs and parameter values
- access to experiment information during and after the run
- configurable notifications on experiment state and progress

# Introduction

## Goals

Our goal is to develop a tool for computational researchers to enable easy

- specification and management of experiment queues
- batch submission of experiment jobs to computing clusters
- monitoring of ongoing experiments' logs and parameter values
- access to experiment information during and after the run
- configurable notifications on experiment state and progress
- configurable criteria for experiment autotermination

# Introduction

## Goals

Our goal is to develop a tool for computational researchers to enable easy

- specification and management of experiment queues
- batch submission of experiment jobs to computing clusters
- monitoring of ongoing experiments' logs and parameter values
- access to experiment information during and after the run
- configurable notifications on experiment state and progress
- configurable criteria for experiment autotermination
- logging of experiment history

# Introduction

## What

In essence the product is a Python-based tool that enables computational researchers to conduct their research more effectively.

# Introduction

## What

In essence the product is a Python-based tool that enables computational researchers to conduct their research more effectively.

- It utilizes SSH and TCP sockets to distribute the computational workload into computer clusters. It supports the Slurm job and resource manager but can function without it as well.

# Introduction

## What

In essence the product is a Python-based tool that enables computational researchers to conduct their research more effectively.

- It utilizes SSH and TCP sockets to distribute the computational workload into computer clusters. It supports the Slurm job and resource manager but can function without it as well.
- It is framework agnostic in that it permits the use of a very wide variety of tools to actually conduct the computing.

# Results

## Sprint 0

Goal: Team building and preparing for sprint 1

# Results

## Sprint 0

Goal: Team building and preparing for sprint 1 Done

# Results

## Sprint 0

Goal: Team building and preparing for sprint 1 Done
Product Backlog Items: *None*

# Results

## Sprint 0

Goal: Team building and preparing for sprint 1 Done
Product Backlog Items: *None*
Sprint 0 took a lot of effort from us since the project topic was very challenging to dive into. Also none of us had done this course before. Interviews with Jelena & Simo helped us to understand the project.

# Results

## Sprint 0

Goal: Team building and preparing for sprint 1 Done
Product Backlog Items: *None*
Sprint 0 took a lot of effort from us since the project topic was very challenging to dive into. Also none of us had done this course before. Interviews with Jelena & Simo helped us to understand the project.
We were proud of our efforts in the sprint.

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI.

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI.
  Done

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources.

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources. Done

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources. Done
- US3: As a user, I want to specify experiments by name, files and parameters.

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources. Done
- US3: As a user, I want to specify experiments by name, files and parameters. Done

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources. Done
- US3: As a user, I want to specify experiments by name, files and parameters. Done
- US4: As a user, I want to submit experiments to unmanaged nodes.

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources. Done
- US3: As a user, I want to specify experiments by name, files and parameters. Done
- US4: As a user, I want to submit experiments to unmanaged nodes. Done

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources. Done
- US3: As a user, I want to specify experiments by name, files and parameters. Done
- US4: As a user, I want to submit experiments to unmanaged nodes. Done
- US5: As a user, I want an experiment status report so that I can review experiment status details.

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources. Done
- US3: As a user, I want to specify experiments by name, files and parameters. Done
- US4: As a user, I want to submit experiments to unmanaged nodes. Done
- US5: As a user, I want an experiment status report so that I can review experiment status details. Done

# Results

## Sprint 1

Goal: *Develop a prototype that offers the most basic functionality via a CLI*

Product Backlog Items:

- US1: As a user, I want a basic user guide that would cover the installation of Neronet and its use via CLI. Done
- US2: As a user, I want to specify clusters by address and type to specify my computing resources. Done
- US3: As a user, I want to specify experiments by name, files and parameters. Done
- US4: As a user, I want to submit experiments to unmanaged nodes. Done
- US5: As a user, I want an experiment status report so that I can review experiment status details. Done

Just a prototype, a lot of work to do before user testing.

# Demo

Demo script:

1. Neronet Installation, preferences and initial setup of clusters
2. Specification of clusters via CLI
3. Specification of an experiment
4. Submission of the specified experiment to an unmanaged node
5. Retrieval of experiment status report

# Quality

Definition of done:

- We defined Done in three levels: BI, sprint and project
- BI level: unit tests done where applicable, functional test coverage 80%, conformity (PEP-8), commented, documented, peer reviewed
- Sprint level: BI:s are Done, increment is tested and reviewed, sprint goal is achieved
- Updates to DoD:
  - We replaced *unit test coverage 90%* with *unit tests are written where applicable* – the old metric was not useful for all BIs
- Otherwise, we have followed our DoD as planned.

# Quality

| US | UTC | FTC | Com | Doc | Rev |
|---|---|---|---|---|---|
| 1 | 3 | 3 | 3 | 3 | 3 |
| 2 | 2 | 2 | 3 | 2 | 3 |
| 3 | 3 | 3 | 3 | 3 | 3 |
| 4 | 3 | 3 | 3 | 3 | 1 |
| 5 | 1 | 3 | 2 | 1 | 1 |
| | 2 | 3 | 3 | 2 | 2 |

Qualitatively we achieved our standards only partially:

- Unit and functional test coverage – good
- Quality of comments and documentation – good
- Peer review – ok (done rather quickly)

# Quality

Used QA practices and tools:

- Commenting & documentation – forces to rethink from another perspective, facilitates peer review
- Python standard unittest framework – test automation
- Peer review – quality assurance

# Quality

Relevant quality attributes:

- Usability – We developed a basic user guide in the first sprint which will help even newbies understand our software → The usability of our software should be good
- Reliability – Unfortunately, we didn't have as much time to test our software in the first sprint as we'd hoped. We will make up for this by using more of our second sprint for testing and less for making new features
- Extendability – At the moment our software's extendability is ok, difficult to say anything about the final product
- Performance – At the moment, our software's performance is ok

# Effort

| S | Sa | Te | Tu | Jo | Ii | Ma |
|---|---|---|---|---|---|---|
| 0 | 140/50 | 36/35 | 45/35 | 40/35 | 36/35 | 43/35 |
| 1 | 46/30 | 28/33 | 33/33 | 38/33 | 25/33 | 33/33 |
| 2 | 0/30 | 0/33 | 0/33 | 0/33 | 0/33 | 0/33 |
| 3 | 0/15 | 0/33 | 0/33 | 0/33 | 0/33 | 0/33 |
| 4 | 0/15 | 0/33 | 0/33 | 0/33 | 0/33 | 0/33 |
| 5 | 0/15 | 0/33 | 0/33 | 0/33 | 0/33 | 0/33 |
| 6 | 0/20 | 0/25 | 0/25 | 0/25 | 0/25 | 0/25 |
| | 186/175 | 64/225 | 78/225 | 78/225 | 61/225 | 76/225 |

# Retros: Sprint 0

Sprint planning:

- BI clarity and simplicity (user guide helps)
- It might have been better if the PO had created the stories from scratch -Matias, Tuomo
- We should actively seek more input from PO when developing the user guide
- we should make sure we reserve enough time for the actual story selection on Monday -Matias

# Retros: Sprint 0

Daily scrums:

- We have mostly been doing teamwork, so there has been little new info in the scrums -Matias -Joona -Teemu
- They have been overly long and they have extended due to inexperience.
- People are late.

# Retros: Sprint 0

Teamwork sessions:

- Sessions are too long and sometimes people get hungry
- Generally someone has to leave early or comes late

# Retros: Sprint 0

Tools:

- Flowdock is good x6
- Agilefant has a steep learning curve. -Iiro
- People tend to forget to log their time at agilefant
- Github hasn't been used much. Hope to use it more during future sprints
- Floobits is very buggy.
- Top 3 tools: 1) GitHub 2) Flowdock 3) Agilefant
- Worst 3 tools: 1) Floobits 2) Six tactics 3) Agilefant

# Retros: Sprint 0

How teamwork could be improved:

- People should be more on time
- Scrum Master shouldn't have to work as a team leader too.
- Hard to think of improvements since we haven't really started coding yet