

# Increase ILC Software performance using Cloud Computing

Tomasz Wojtoń

Institute of Nuclear Physics PAN, Kraków

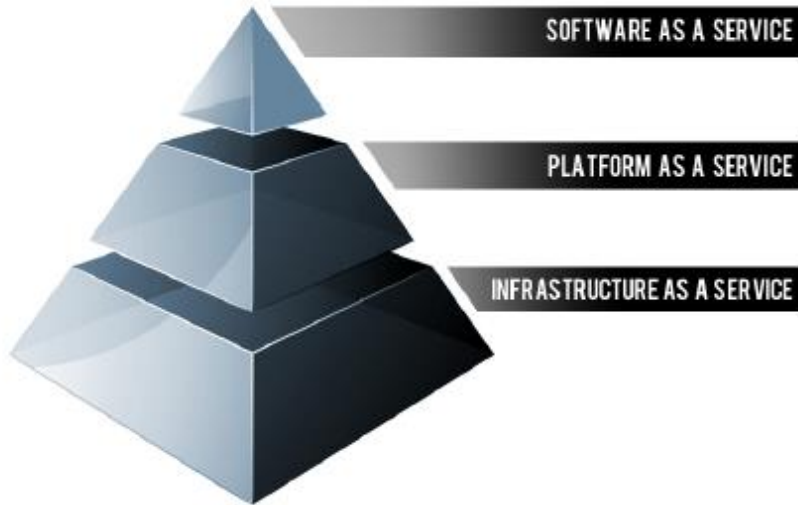


# Introduction

- ILC Simulations
  - Frequently changing input parameteres
  - High statistics
- Takes a lot of time
- Fast simulation and reconstruction data to analysis
- Solution – **Cloud Computing**

# Cloud Computing

- Provide computer infrastructure to end-users in friendly way **on demand**
- Elastic resource allocation
- Efficient usage of computing resources in institution
- Cloud Computing models:



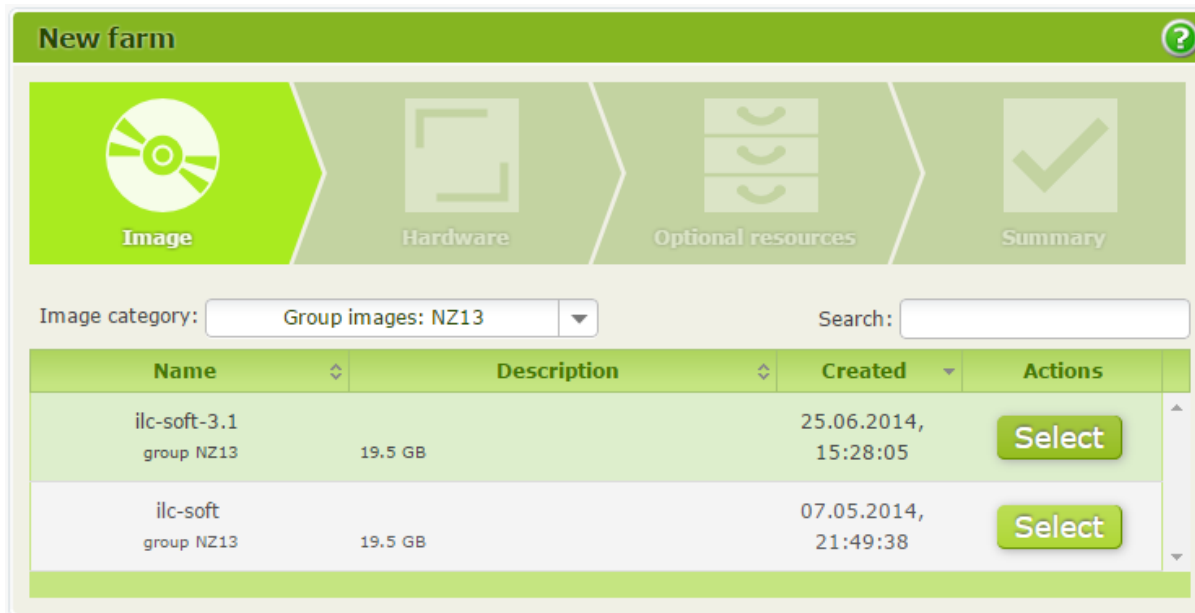
- SaaS – applications designed for end-users, delivered over the web
- PaaS – set of tools and services designed to make coding and deploying those applications quick and efficient
- **IaaS** – is the hardware and software that powers it all – servers, storage, networks, operating systems

# CC1 – Private Cloud Solution

- Complete solution for Cloud Computing developed and deployed in IFJ PAN
- Use by many scientists in several experiments
- Convenient and flexible access to computing power
- Web access interface
- Dedicated feature for research institutes and universities – **Farms (Virtual Clusters)**

# CC1 Virtual Clusters – in few easy steps

- 1. Select Virtual Machine image

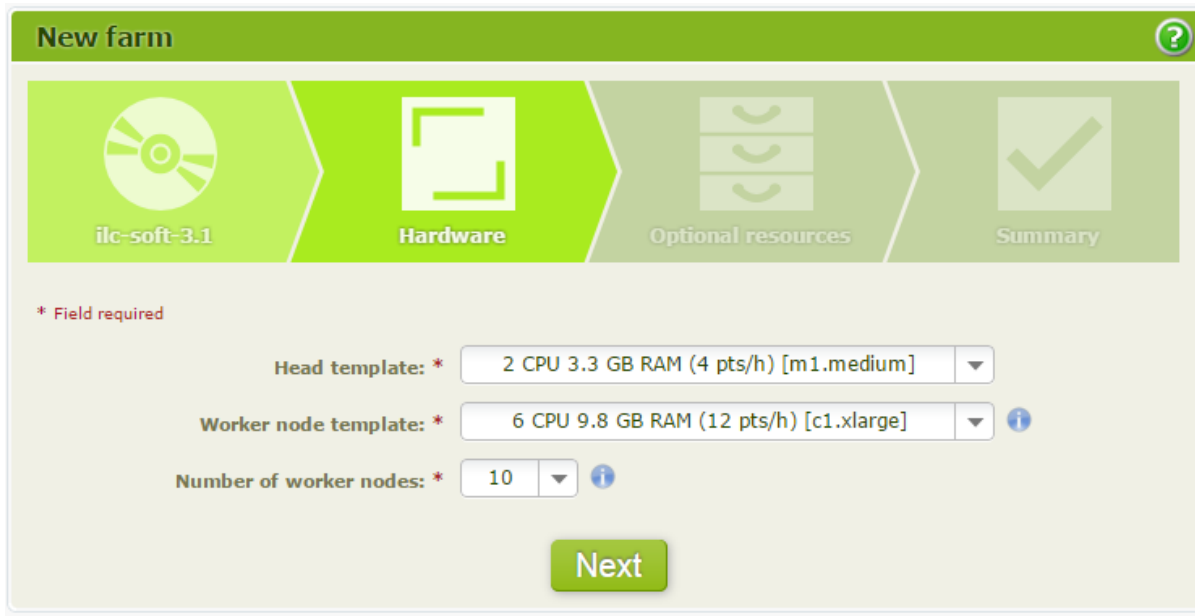


The screenshot shows a 'New farm' wizard interface with a green header and a progress bar. The progress bar has four steps: 'Image' (highlighted in green), 'Hardware', 'Optional resources', and 'Summary'. Below the progress bar, there is a search area with 'Image category:' and a dropdown menu showing 'Group images: NZ13'. To the right is a 'Search:' input field. Below this is a table with columns: 'Name', 'Description', 'Created', and 'Actions'. The table contains two rows of image data.

Name	Description	Created	Actions
ilc-soft-3.1 group NZ13	19.5 GB	25.06.2014, 15:28:05	Select
ilc-soft group NZ13	19.5 GB	07.05.2014, 21:49:38	Select

# CC1 Virtual Clusters – in few easy steps

- 2. Choose hardware (CPU, memory amount) and number of nodes in cluster



The screenshot shows a 'New farm' configuration interface with a progress bar at the top. The progress bar has four steps: 'ilc-soft-3.1', 'Hardware', 'Optional resources', and 'Summary'. The 'Hardware' step is currently active and highlighted in green. Below the progress bar, there is a red asterisk and the text '\* Field required'. The configuration fields are:

- Head template:** \* 2 CPU 3.3 GB RAM (4 pts/h) [m1.medium] (dropdown menu)
- Worker node template:** \* 6 CPU 9.8 GB RAM (12 pts/h) [c1.xlarge] (dropdown menu with an information icon)
- Number of worker nodes:** \* 10 (dropdown menu with an information icon)

A green 'Next' button is located at the bottom center of the form.

# CC1 Virtual Clusters – „Farms”

- 3. Assign IP or attach external storage

### New farm

\* Field required

Assign IP address:  ⓘ

Attach disk volume:  ilc-data (195.3 GB) ⓘ

Attach ISO image:  ▼

VNC:  ⓘ

# CC1 Virtual Clusters – „Farms”

- 4. Create and run

The screenshot shows a management interface for a virtual cluster named "ilc". At the top, there is a green header with the cluster name "ilc". Below the header, there is a row of action buttons: "Destroy" (with a red X), "Save and shutdown" (with a floppy disk icon), "Assign disk" (with a green checkmark), "Revoke disk" (with a red X), and "Assign external IP" (with a blue globe icon). Below these buttons are three more options: "Graphical console (VNC)" (with a monitor icon), "Reset password" (with a key icon), and "Set SSH key" (with a key icon).

The main content area displays the following details:

<b>Name:</b> ilc	<b>Image:</b> ilc-soft-3.1
<b>Created:</b> 01.01.2015, 18:28:13	<b>Uptime:</b> 1 day, 5 h, 53 min, 9 s
<b>Head template:</b> 2 CPU 3.3 GB RAM (4 pts/h)	<b>Worker node template:</b> 4 CPU 6.6 GB RAM (8 pts/h)
<b>Head private IP:</b> 10.16.16.118	<b>Head public IP:</b> empty
<b>VNC:</b> 192.245.169.1:5949	<b>VNC password:</b> <a href="#">Show</a>
<b>Disks:</b> ilc-data,	<b>ISO images:</b> none

Below the details, there is a green bar indicating the cluster state: "State: Running" with an information icon.

The bottom section shows five worker nodes, each in a green box with a red progress indicator and a small bar chart:

- ilc-head: 3% running ctx, IP: 10.16.16.118
- ilc-wn1: 0% running ctx, IP: 10.16.16.86
- ilc-wn2: 0% running ctx, IP: 10.16.16.122
- ilc-wn3: 1% running ctx, IP: 10.16.16.78
- ilc-wn4: 0% running ctx, IP: 10.16.16.110



# ILC Software & Virtual Clusters

- **Example** – Generate 1000 events (500 GeV,  $e^+e^- \rightarrow e^+e^-X$ ), simulate and reconstruct data – more than **10 hours**
- The same case using farm of nodes with total 25 CPU cores takes **~1 hour**

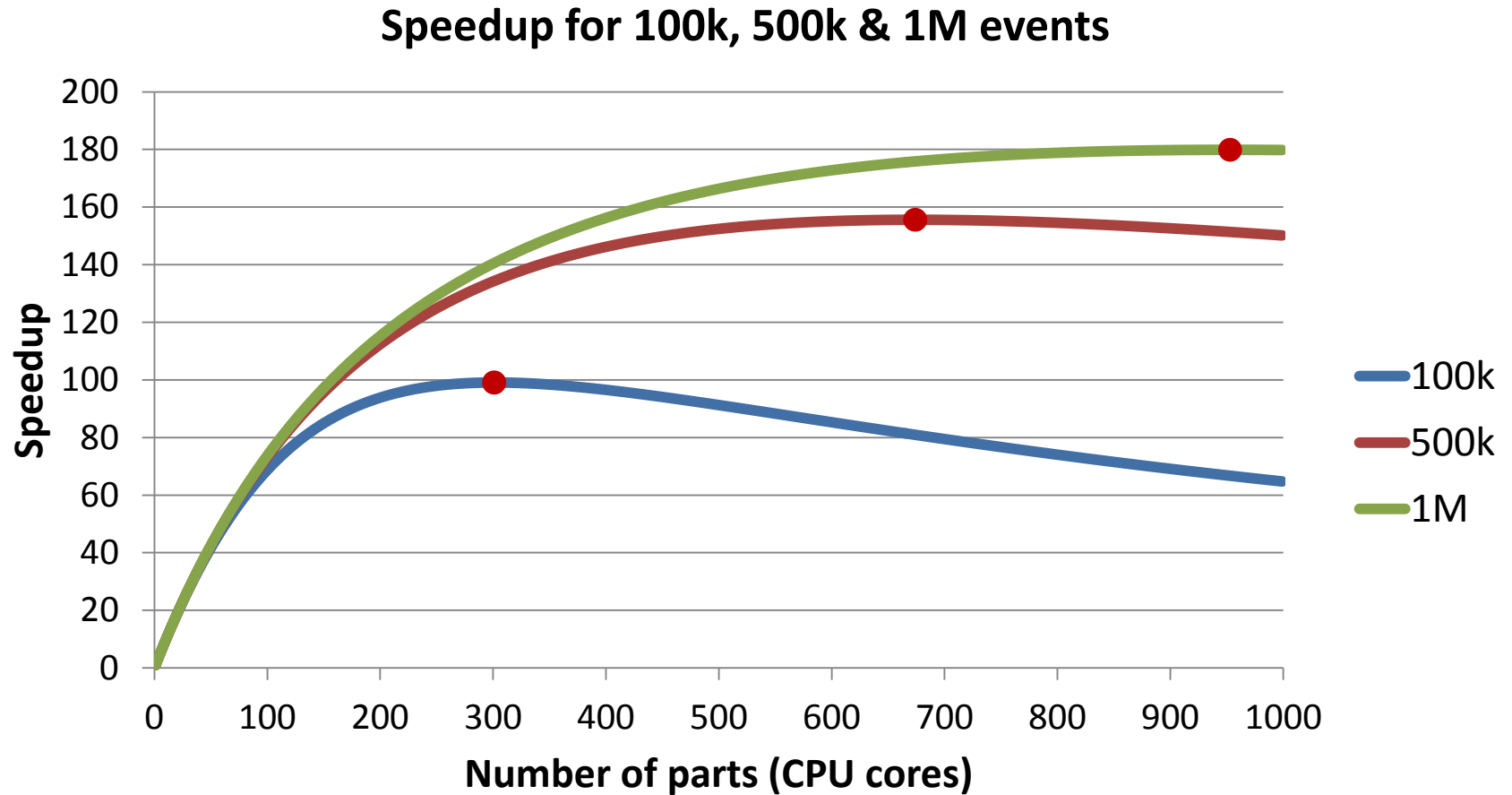


# ILC Software & Virtual Clusters

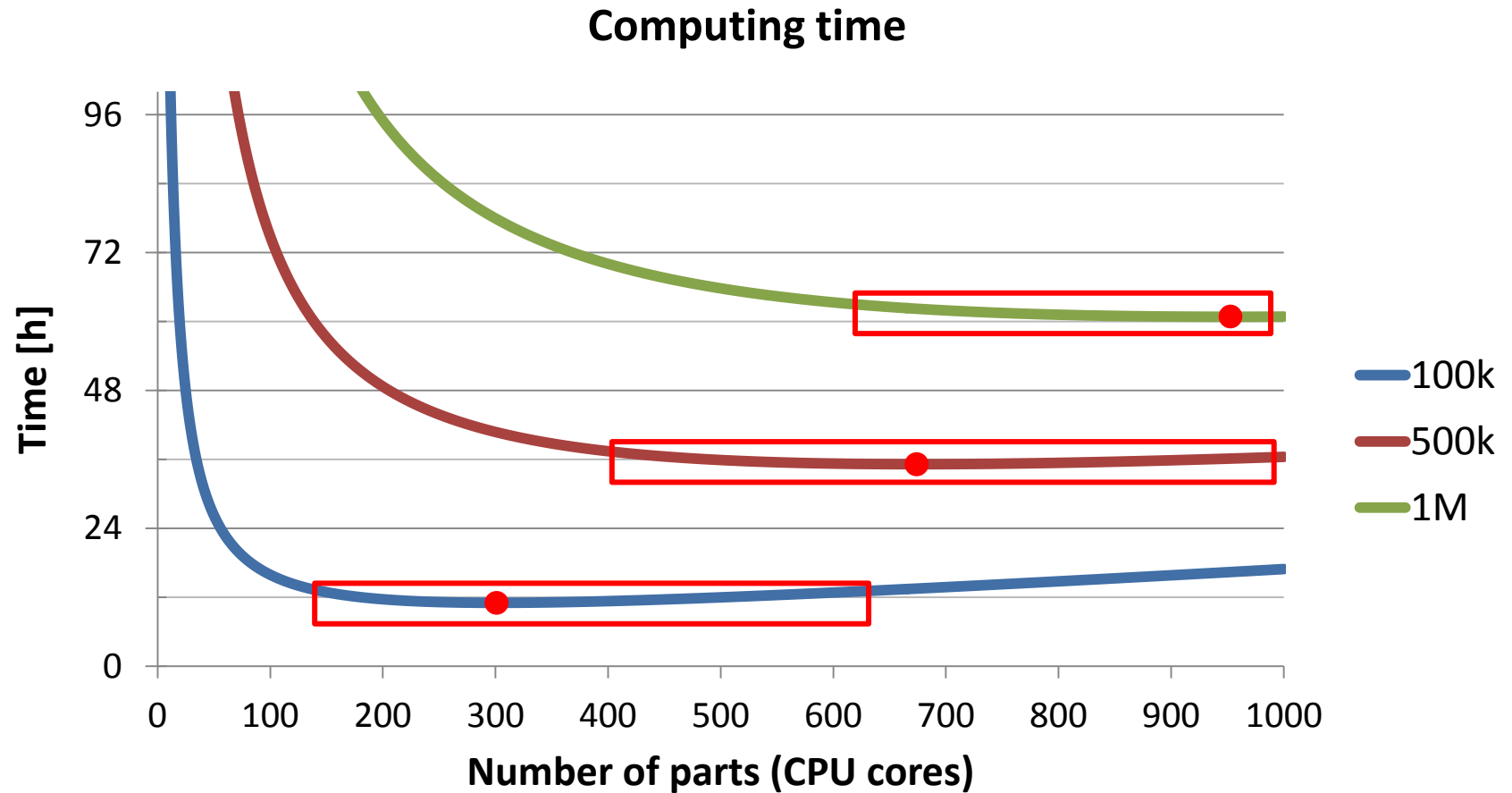
- ILC Software automation
  - Splitting job to smaller parts
  - Automatic tasks processing
  - Merge parts
- Preparing:
  - Farm with attached storage for data
  - Events generator (pythia)
  - Mokka and Marlin steering files
  - Fill up the starting script
- Just `./start`

```
EVENTS=100000  
PARTS=240  
GAMMA_FILE="/data/res.HEPEvt"  
GAMMA_DIR_PARTS="/data/events-parts"  
STORAGE_DEVICE="/dev/vdb1"  
STORAGE_PATH="/data"
```

# Performance analysis – speedup



# Performance analysis – computing times



# Summary

- More CPU cores is **not** always better
- Performance measurements are **important**
- It could be used as a benchmark to select how many events could be processed at specific time with defined number of CPU cores.
- Cloud computing is **good candidate** for ILC Computing backend