

## Summer Student Project

# An alternative approach to configure permanent tasks in LHCb Online farm nodes

### Krzysztof Wilczyński

Supervisors: Markus Frank, Beat Jost

August 28, 2018



CERN, Meyrin



## About me

#### Introduction

#### About me

#### The system

Online Farm Farm controller Boot script

#### The motivation

- Upgrades
- Development

#### The results

Results Tools



### I am... an engineer!

- Master student of Automatic Control and Robotics (spec. Robotics)
- Faculty of Power and Aeronautical Engineering, Warsaw University of Technology

### I used to work at...

- Bosch Rexroth
- Airbus Military Defence and Space
- Student Association for Vehicle Aerodynamics
- Polish National Opera





## Online farm computing nodes

#### Introduction

About me

#### The system

#### Online Farm

Farm controller Boot script

#### The motivation

Upgrades

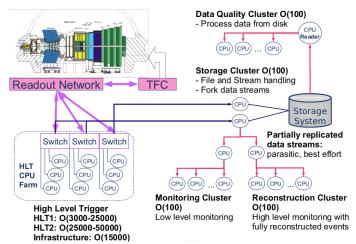
Development

#### The results

Results Tools



### Data Processing Apps > 80.000 Instances





#### Introduction

About me

#### The system

Online Farm

#### Farm controller

Boot script

#### The motivation

Upgrades API Development

#### The results Besults

Tools

The handling of the permanent processes on the data processing nodes is based on sending commands to pcSrv process running on each of the corresponding "Controls PCs".

LHCb Online Farm Process Controler on Researchgate



Krzysztof Wilczyński

Summer Student Project



## Farm boot script

#### Introduction

About me

### The system

Farm controller

Boot script

#### The motivation Upgrades API Development

The results Results In the current solution, all processes (scripts) started on the farm nodes are grouped in a single, huge python script that prints out ready to execute pcAdd commands for a given node name.



### A command used to start a task on node(s):

pcAdd(regex, start parameters, script, script parameters)



Krzysztof Wilczyński

Summer Student Project



## Farm boot script

#### Introduction

About me

#### The system

Online Farm

Farm controller

Boot script

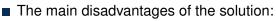
#### The motivation

Upgrades API

Development

#### The results

Results Tools



- Modifications of task parameters are difficult
- It is easy to make a modification that harms dependencies in the task sets (no error prevention mechanism)
- Only a specialist who knows the boot script structure can use it (no high-level interface)
- There is no easy way of knowing which tasks are running on given node (one has to analyze the boot script line by line)

## The boot script has been created as a "quick hack" about 10 years ago. The time has come to upgrade it!





## My project

#### Introduction

About me

#### The system

Online Farm Farm controller Boot script

#### The motivation

Upgrades

Development

#### The results Results

Tools



The solution: create a system for the process controller infrastructure utilizing a database driven approach. The main goals were to:

- simplify the modifications of hierarchical structure of tasks running on the nodes
- prevent human errors breaking the system integrity
- create one source of information regarding processes running on a given node
- create a reliable and future-proof API for future developments



## What do I mean by Main API?

#### Introduction

About me

#### The system

Online Farm

Farm controller

#### The motivation

Upgrades

#### API

Development

#### The results

Results Tools



### **API: Application Programming Interface**



The created API is a Python class containing methods (add, delete, modify, get, assign, inSet) that allow safe access to the underlying database. It is a high-level connector providing an easy integration of different client applications.



## Stages of development

Introduction

About me

#### The system

- Online Farm
- Boot script

#### The motivation

- Upgrades
- Development

#### The results Results Tools

### Back-end

- Database schema architecture
- Main database API
- New boot script
- Unit testing script (internal error prevention)
- Frontend connectors: JSONRPC, (REST, XMLRPC)

### Front-end

- Command line user interface
- Graphical user interface (web application)





### Current results

#### Introduction

About me

#### The system

Online Farm

Farm controller

Boot script

#### The motivation

Upgrades

API

Development

#### The results

Results

Tools

									LH
		Operation							
Tasks Task Sets Node C	lasses Nodes			Unique name	Script name	Script parameters	Process identifier	٣	PoAdd parame
LogSumDefSiv	*			LogDebultSrv	AugServer.sh	-no-authentication -K1	LogDefaultSrv	11	Sort Ascending
LogDef				LogSaudSrv	AugServer.sh	- no-authentication - K 1	LogGaudSrv	12	Sort Descending
RDpublish		8	3	LogLHEb1Sev	Augierverah	-no-authentication -K1	LogLHOb1Sev	α	Columns
TaskSupenisor			-4	Logi.Htb2Srv	AugServer.sh	-no-authentication -K1	LogiH0s25rv		-5 100 -11 -s Pic
webDID			5	LogLHCbAGrv	AugServer.sh	-no-authentication -K1	LogLHOsASrv		-5 100 -11 -5 Pc
PropertyServer		8	6	TANServ	/TanSencsh	- no-authentication -K1	TANServ		
LogGaudGumSrv		8		pinglings	ringirvah	-no-authentication -K1	pinglin_u		
LogGaudi		B	4	RDcollect	/ROMONCollect.sh	-no-authentication -K1	ROcollect		
LogLHCb15umSrv		8	- 9	SSSRecover	/home/beat/ssscheck.sh	-no-authentication -K1	SSSRecover		
LogLHCb1			20	TorrentLoader	/BitTorrentLoader.sh	- no-authentication -K 1_	TorrentLoader		
LogLHCb2SumSrv				LogSumDelSrv	AugServer.sh	-no-authentication -K1	LogiumDefirv		-13-5100-s fm
LogLHCb2			12	LogDef	AugCollectah	-no-authentication -K1	LogDef		43 s fmc -o /m
LogLHCbASumSrv		8	13	R0publish	/MONPublishish	-no-authentication -K1	R0publish		-statDelay-800
LogLHCbA			- 14	TaskSupervisor	/TaskSupervisor.sh	- no-authentication -K 1_	TaskSupervisor		
LogPV555rv_HLT			15	wibDID	/webDID.sh	-no-authentication -K1	webDD		
LogPVSS_HLT			15	PropertyServer	/PropertyServer.sh	-no-authentication -K1	PropertyServer		
HLTEQBridge				LogGaudSumSrv	AagServer.sh	-no-authentication -K1	LogGaud/SumGrv		41-5100-5ga
	v		18	LogSaudi	/LogCollect.sh	- no authentication - K 1	LogGaudi		-11-s gaudi-o /
	ription 🔺		73	LogLHCb1SumSrv	AugServer.sh	-no-authentication -K1	LogLHOs1Sum5rv		-11-5100-5 Pc
ROpublish: This is an incredibly important taxid it is also			20	LogLHEb1	/LogCollect.sh	-no-authentication -K1	LogLHCb1		-11-s hcb1-o /
important to keep this important tar	k as important as it is.	8	21	LogLHDb2Sum5rv	AugServer.sh	-no-authentication -K1	Logi.HOs25um5rv		41-5100-shc

### Open-source code repositories:



## Bitbucket Repository (K. Wilczynski)

Will be moved to LHCb Gitlab Repository (M. Frank)

Krzysztof Wilczyński

Summer Student Project

August 28, 2018

10/13



#### Introduction

About me

#### The system

- Online Farm
- Root script

#### The motivation

- Upgrades API Development
- The results
- Tools

- Back-end programming language: Python 2.7.15rc1
- Front-end programming language: JavaScript
- Database engine: SQLite (+ python sqlite3, sqlalchemy)
- Front-end connector protocol: JSONRPC
- Front-end framework: Sencha Ext JS ver. 6.2.0
- Git version control: Bitbucket + GitKraken

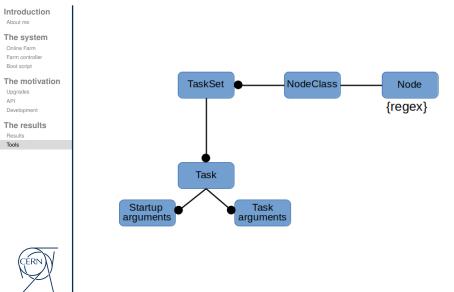


# Thank You,





## Task grouping hierarchy



Krzysztof Wilczyński

Summer Student Project