Smart workflows

Case studies

What about distributed computing?

Infrastructure explained

Focus on computing techniques

sara.vallero@to.infn.it

Smart workflows

(On how to develop your application)

Reproducibility

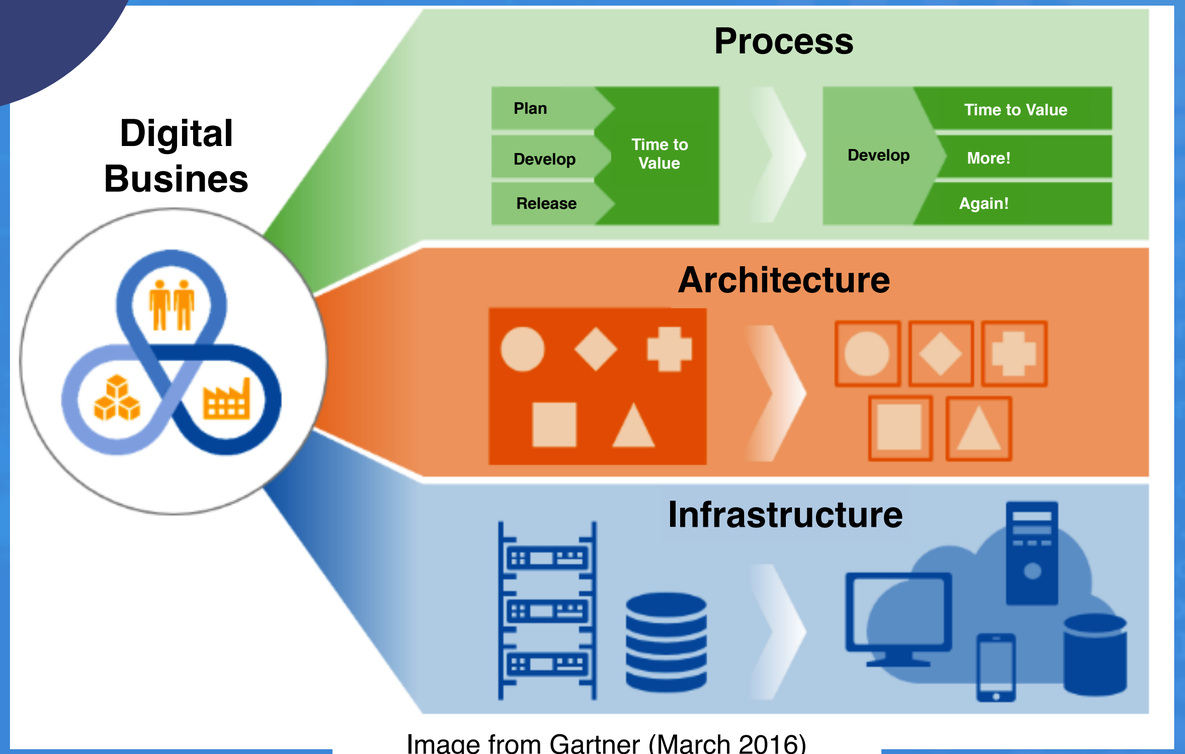


Image from Gartner (March 2016)

DevOps and Continuous Integration (CI)

Microservices (Wikipedia): service-oriented architecture (SOA) that structures an application as a collection of loosely coupled services. It improves modularity and makes the application easier to understand, develop and test. It also parallelizes development.

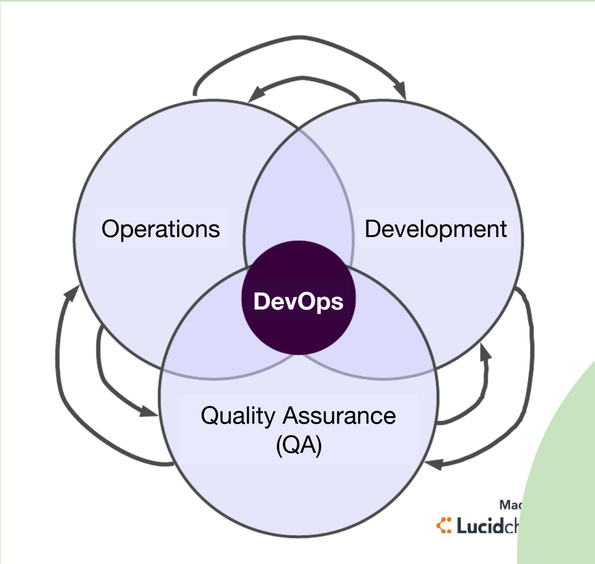
Internet of things and heterogeneous infrastructures

DevOps and Continuous Integration (CI)

DevOps

DevOps (Development and Operations) is a philosophy and practice focused on agility, collaboration, and automation within IT and development team processes.

The goal is to **bridge the gap between IT operations and development** to improve communication and collaboration, create more seamless processes, and align strategy and objectives for faster and more efficient delivery.



DevOps philosophy principles:

- Automation
- Iteration
- Self-service
- Continuous improvement
- Collaboration
- Continuous testing

Continuous Integration (CI)

CI is a software development practice in which developers regularly merge their code changes into a **shared repository** where those updates are automatically tested.
CI ensures that the **most up-to-date and validated code** is always readily available to developers.

Travis CI

Jenkins

GitHub

Continuous Delivery

Code changes are automatically built, tested, and **packaged** for release into production.

Continuous Monitoring and Feedback

Continuous Deployment

Every validated change is automatically released to users.

docker

(Wait for Lesson 3)

kubernetes

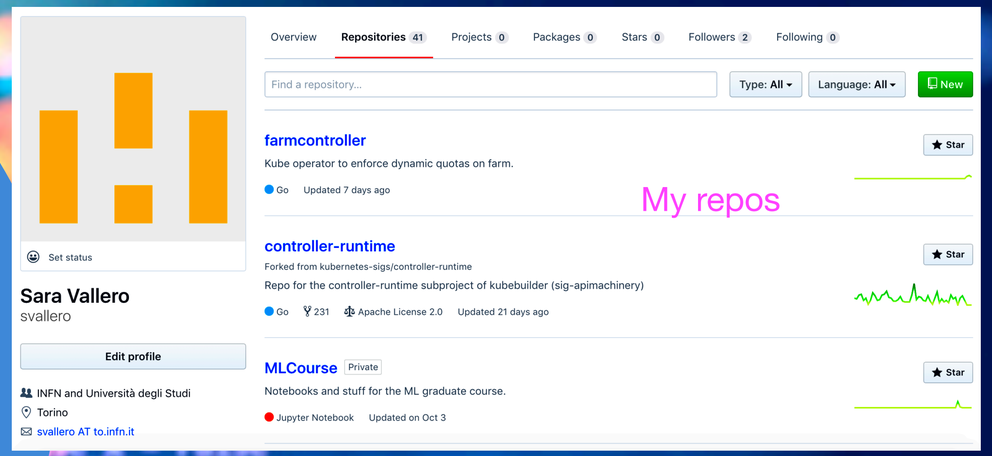
(Wait for Lesson 3)

GitHub

The hub

A **Web Platform** to:

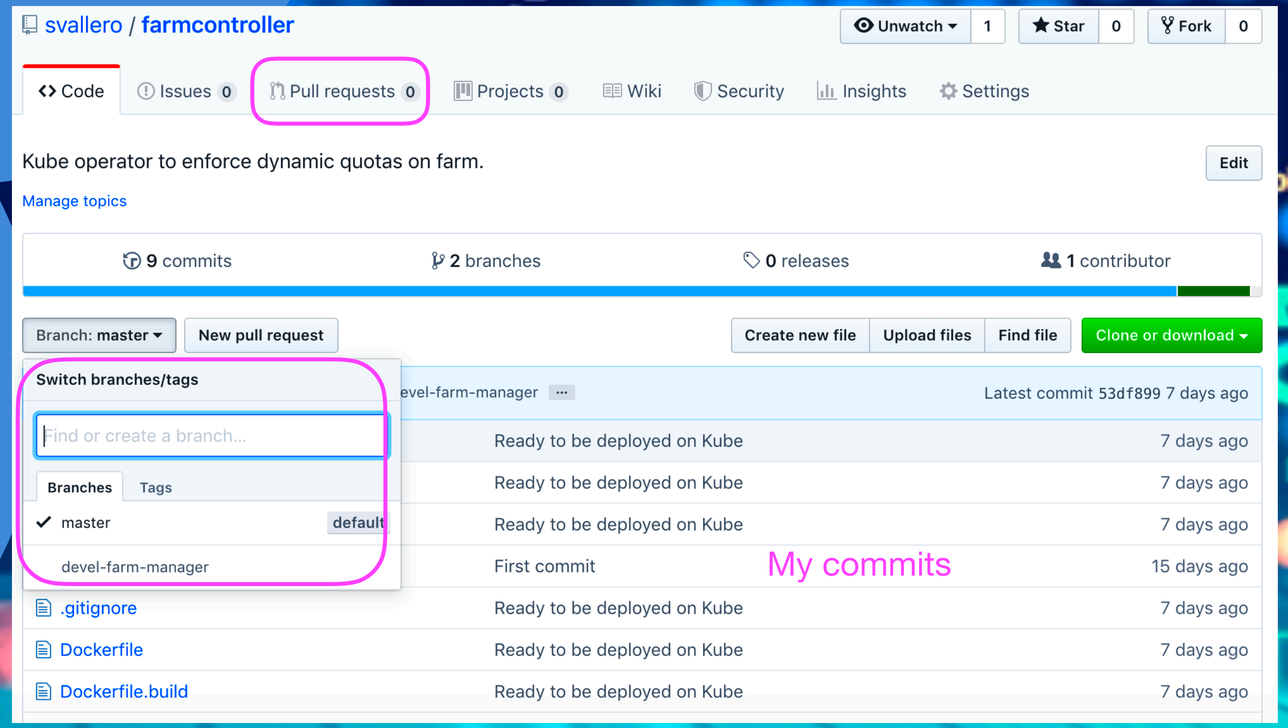
- Store your projects
- Collaborate
- Host your project's documentation and more
- Implement Continuous Development
  - Automatic builds
  - WebHooks

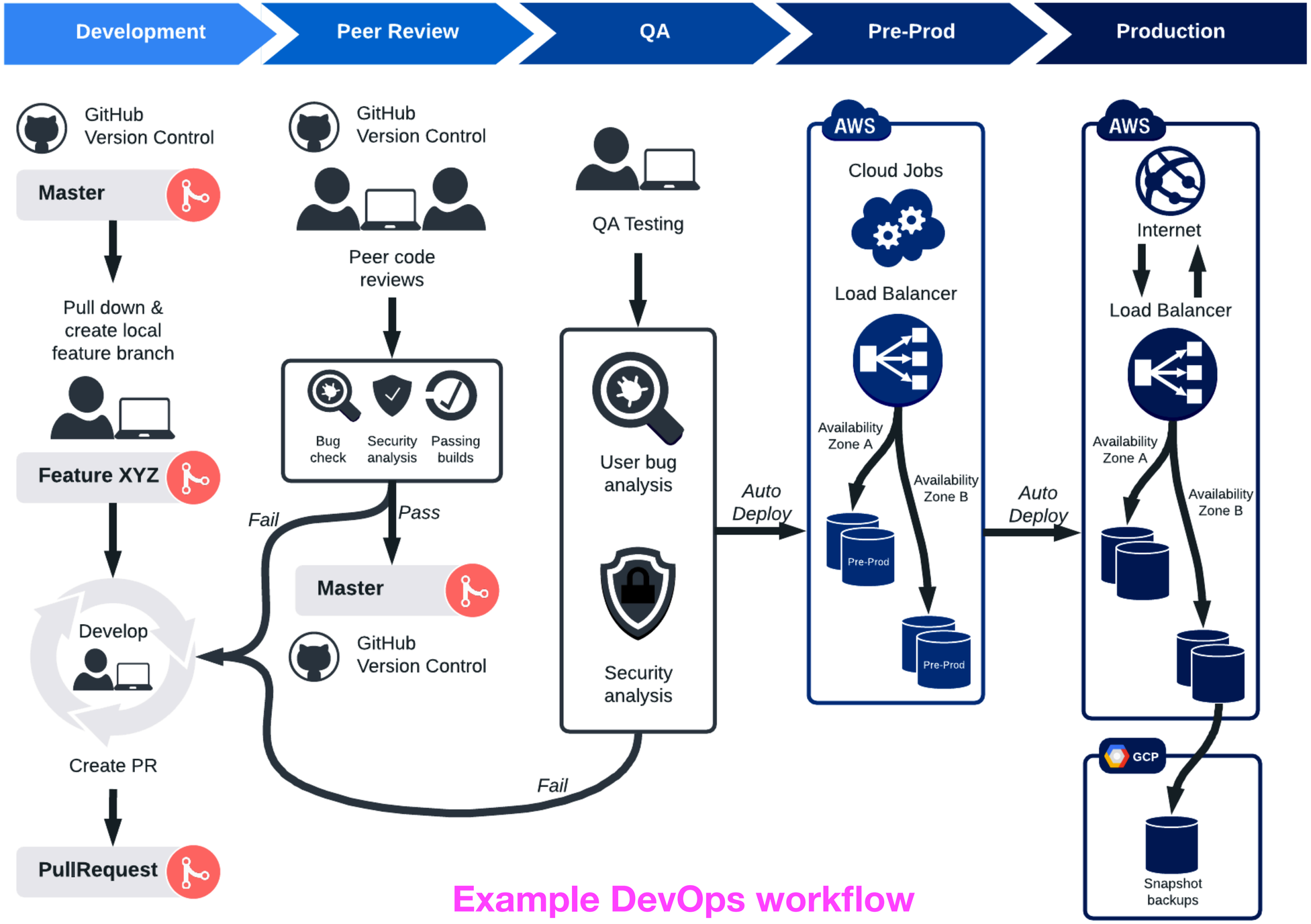


Git

A **Version Control System**:

- You make constant changes to the code, **releasing new versions**
- Keep the revisions straight, storing modifications in a **central repository**
- Make it easy to **collaborate**: download/upload new revisions
- Efficient storage of file changes
- File integrity checks





Example DevOps workflow

Reproducibility

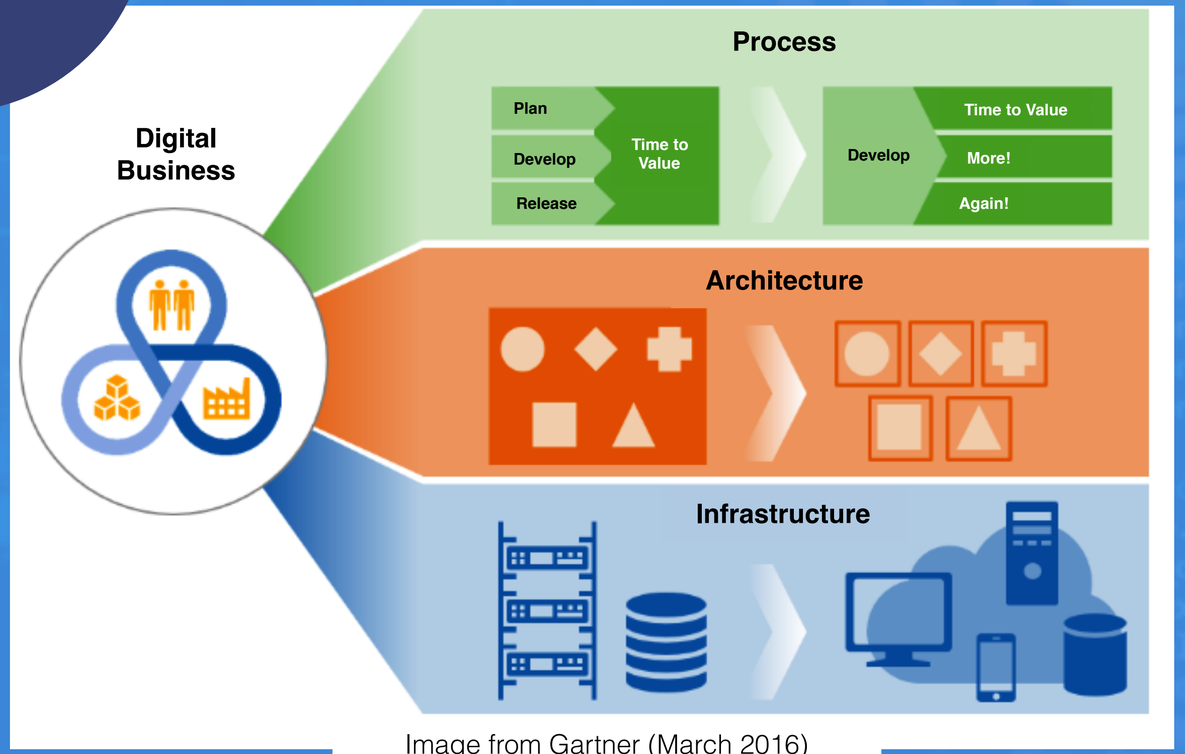


Image from Gartner (March 2016)

DevOps and Continuous Integration (CI)

Microservices (Wikipedia): service-oriented architecture (SOA) that structures an application as a collection of loosely coupled services. It improves modularity and makes the application easier to understand, develop and test. It also parallelizes development.

Internet of things and heterogeneous infrastructures

Reproducibility

The more sophisticated science becomes, the harder it is to **communicate results**. Papers today are longer than ever and full of jargon and symbols. They depend on chains of computer programs that generate data, and clean up data, and plot data, and run statistical models on data. These programs tend to be both so sloppily written and so central to the results that it's contributed to a **replication crisis**, or put another way, a failure of the paper to perform its most basic task: to **report what you've actually discovered, clearly enough that someone else can discover it for themselves**.

- James Somers

The more sophisticated science becomes, the harder it is to **communicate results**. Papers today are longer than ever and full of jargon and symbols. They depend on chains of computer programs that generate data, and clean up data, and plot data, and run statistical models on data. These programs tend to be both so sloppily written and so central to the results that it's contributed to a **replication crisis**, or put another way, a failure of the paper to perform its most basic task: to **report what you've actually discovered, clearly enough that someone else can discover it for themselves**.

- James Somers

The Jupyter Notebook

The Jupyter Notebook is an open-source **web application** that allows you to **interactively** create and share documents that contain:

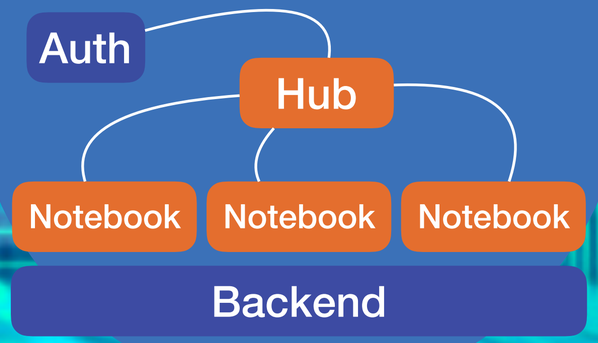
- live code
- equations
- visualizations
- narrative text

Uses include:

- data cleaning and transformation
- numerical simulation
- statistical modeling
- data visualization
- machine learning

The Hub

A multi-user version of the notebook.



Jupiter Notebooks

Multiple language support
(kernels)

jupyter HandsOnSession Last Checkpoint: 07/10/2019 (autosaved)

Logout | Control Panel

File Edit View Insert Cell Kernel Widgets Help | Trusted | Python 3

Run | Code

Hands on session: Analysis of Higgs data ¶

What you will learn

ML methods

- Gradient Boosting Trees with *pyspark.ml*
- MultiLayer Perceptron with *pyspark.ml*
- Deep NN with *Keras*

ML techniques

- Correlation matrix
- Hyperparameter optimisation with *spark_s*
- ROC curves

Dataset description

The dataset used in this example is described
signal process, the production of new theoretic

Each row of this dataset contains 28 features p

- 21 low-level features which represent the l
  - Momentum of the observed paricles
  - Missing transverse momentum
  - Jets and b-tagging information
- 7 high-level features computed from the lo
  (reconstructed invariant masses)

Prepare the execution envir

Your code will run on a single dedicated server
deployed as Kubernetes applications on this s

- JupytherHub
- Jupyter single-user servers
- the HDFS file-system

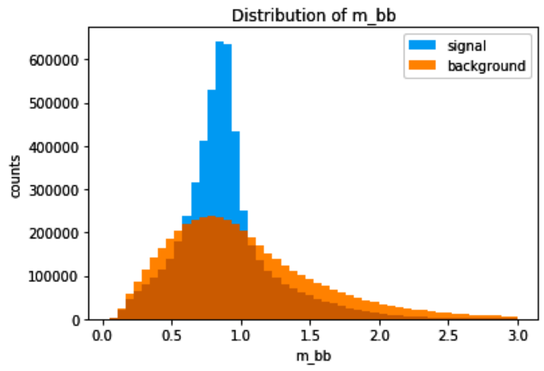
jupyter HandsOnSession Last Checkpoint: 07/10/2019 (autosaved)

File Edit View Insert Cell Kernel Widgets Help

Run | Code

Invariant mass distribution of b-quark pairs

In [12]:
```
plotSignalvsBg(df, 'm_bb')
```



QUESTION 1: is the dataset unbalanced? Do we need undersampling?

In [13]:
```
# motivate your answer here
df.groupBy('label').count().show()
```

```
+-----+-------+
|label|  count|
+-----+-------+
|  0.0|4700495|
|  1.0|5299505|
+-----+-------+
```

QUESTION 2: split the dataset for training and test

In [14]:
```
# your answer goes here
```

Lessons to take-home

- DevOps approach for fast time to value
- Version Control System (Git) to bookkeep code changes and ease collaboration
- Make your workflow understandable and reproducible (Notebooks)