

Acceleratori e rivelatori di particelle

Nicolò Biesuz, Ph.D.



INTERNATIONAL

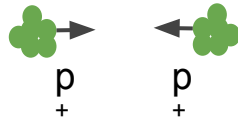


hands on particle physics



Cosa facciamo ad LHC?

```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAd
#include "xAODRootAd
#include "xAODRootAd
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
include "FourMomUti
include "xAODBTaggi
include "xAODBTaggi
include "MuonSelect
include "PathResolv
include "TEfficienc
include "TH1.h"
include "TH1F.h"
include "TH1D.h"
include "TTree.h"
include "TFile.h"
include "TStopwatch
include <iostream>
include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
/// This is a pair
const xAOD::EventI
/// This is a pair
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```



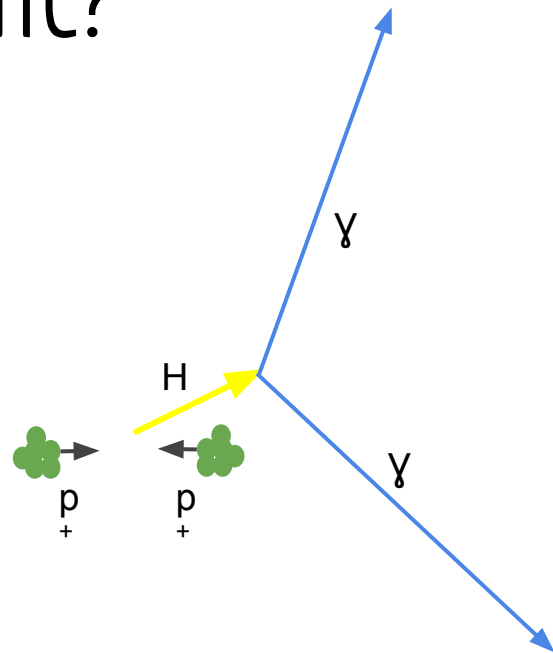
Cosa facciamo ad LHC?

```
<EventLoop/
<EventLoop/
<EventLoop/
<EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTagg1
#include "xAODBTagg1
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnalys

// put your config
// that way they c
protected:
// float cutValue;
/// This is a pair
const xAOD::EventI
/// This is a pair
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP

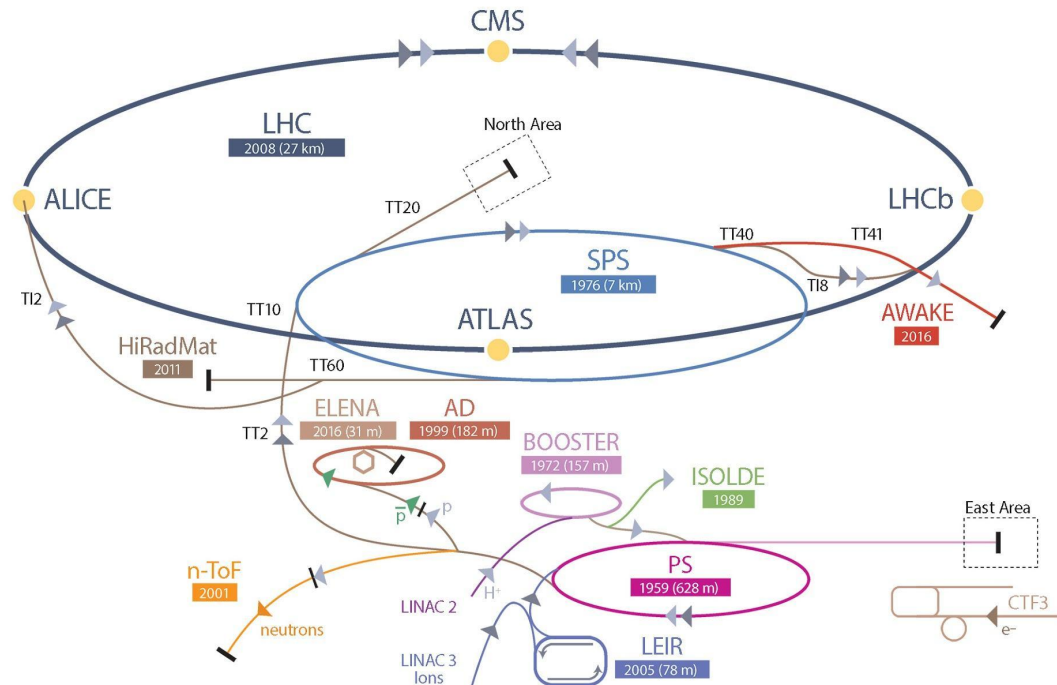
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev

// this is a stand
PisaxAODAnalysis (
```



Cosa impareremo oggi?

CERN's Accelerator Complex



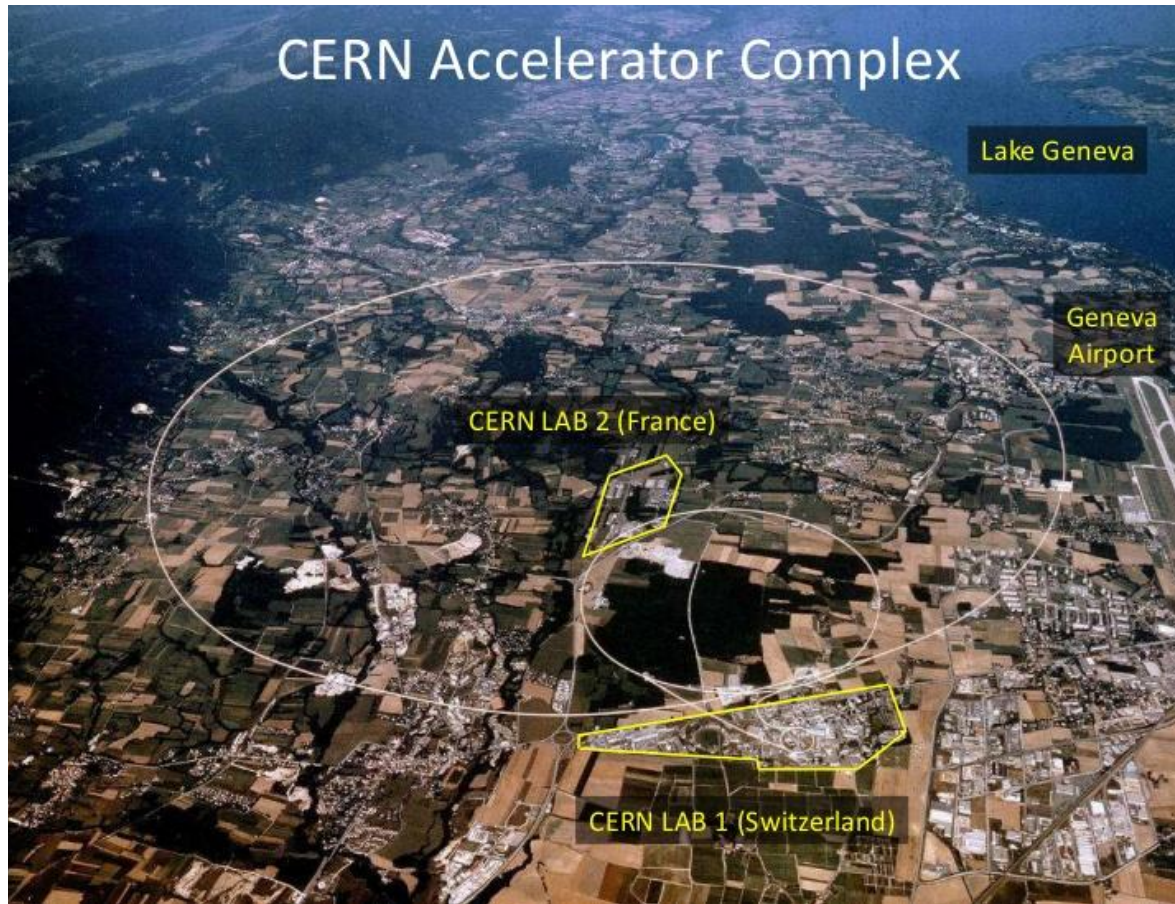
Cosa rappresentano tutte quelle linee?

Cosa sono i 'puntini gialli'?

- ▶ p (proton)
- ▶ ion
- ▶ neutrons
- ▶ \bar{p} (antiproton)
- ▶ electron
- ▶▶▶ proton/antiproton conversion

Cosa Impareremo oggi?

CERN Accelerator Complex



Cosa rappresentano tutte quelle linee?

- Sì, sono acceleratori di particelle

Quanto è vecchia quella foto?

- Eeeeeehh!



```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
/// This is a poin
const xAOD::EventI
/// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

Acceleratori di particelle

- Un acceleratore di particelle è una macchina il cui scopo è quello di produrre fasci di ioni o particelle subatomiche con "elevata" energia cinetica.
- Un tipico esempio: una vecchia televisione
- Vengono usate per scopi:
 - (60%) industriali: impiantazione ioni, sterilizzazione
 - (35%) medici: adroterapia, produzione isotopi
 - (5%) ricerca: materiali, particelle
- Per capire come funziona un acceleratore abbiamo bisogno di un'unica formula:

$$\mathbf{F} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B})$$

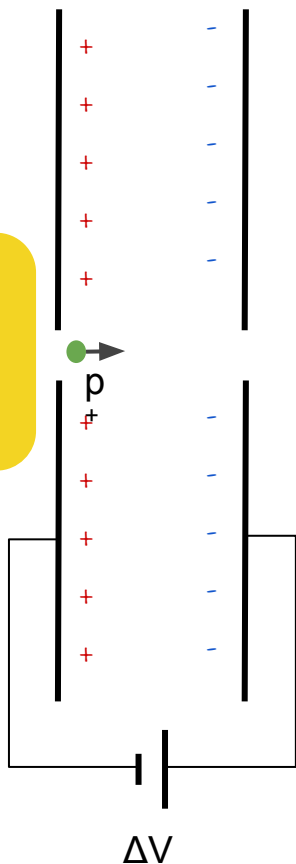


Agol - Briovenga (1964)

```
<EventLoop/
<EventLoop/
<EventLoop/
<EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
/// This is a poin
const xAOD::EventI
/// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::Event* m_ev
// this is a stand
PisaxAODAnalysis (
```

Acceleratori elettrostatici

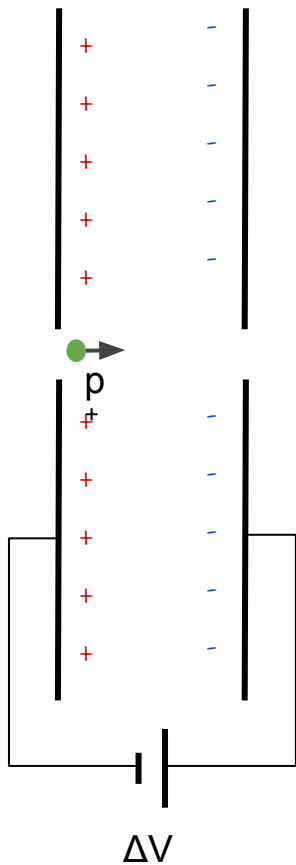
Sorgente di
particelle
(protoni)



- È il principio con cui funzionavano i televisori a tubo catodico;
 - Ho una sorgente di elettroni;
 - Gli elettroni vengono iniettati tra le armature di un condensatore
 - Il campo elettrostatico accelera gli elettroni
- La forza che agisce su un elettrone è:
 - $\mathbf{F} = e\mathbf{E}$
- L'energia guadagnata da un elettrone è:
 - $\int_0^l \mathbf{F}d\mathbf{x} = e \int_0^l \mathbf{E}d\mathbf{x} = e\Delta V$
- Se misuriamo la carica in 'elettroni' l'energia ha le dimensioni di
 - elettroni * Volt = [eV]
- Energie 1 MV - 30 MV

Acceleratori elettrostatici

La sorgente di protoni è una bombola di idrogeno!



- È il principio con cui funzionavano i televisori a tubo catodico;
 - Ho una sorgente di elettroni;
 - Gli elettroni vengono iniettati tra le armature di un condensatore
 - Il campo elettrostatico accelera gli elettroni
- La forza che agisce su un elettrone è:
 - $\mathbf{F} = e\mathbf{E}$
- L'energia guadagnata da un elettrone è:
 - $\int_0^l \mathbf{F}d\mathbf{x} = e \int_0^l \mathbf{E}d\mathbf{x} = e\Delta V$
- Se misuriamo la carica in 'elettroni' l'energia ha le dimensioni di
 - elettroni * Volt = [eV]
- Energie 1 MeV - 30 MeV


```
<EventLoop/
<EventLoop/
<EventLoop/
<EventLoop/
Infrastructure in
include "xAODCore/S
include "xAODRootAc
include "xAODRootAc
include "xAODRootAc
include "xAODEventI
include <xAODTruth/
include "xAODEgamma
include "xAODMissir
include "xAODMissir
include "xAODMissir
include "FourMomUti
include "xAODBTaggi
include "xAODBTaggi
include "MuonSelect
include "PathResolv
include "TEfficienc
include "TH1.h"
include "TH1F.h"
include "TH1D.h"
include "TTree.h"
include "TFile.h"
include "TStopwatch
include <iostream>
include <iterator>
class PisaxAODAnalysis
```

```
// put your config
// that way they c
protected:
// float cutValue;
/// This is a poin
const xAOD::EventI
/// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

The Nobel Prize in Physics 1951



Sir John Douglas
Cockcroft
Prize share: 1/2



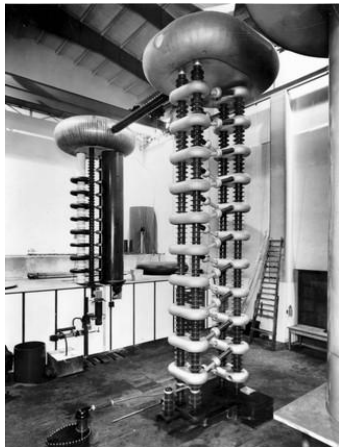
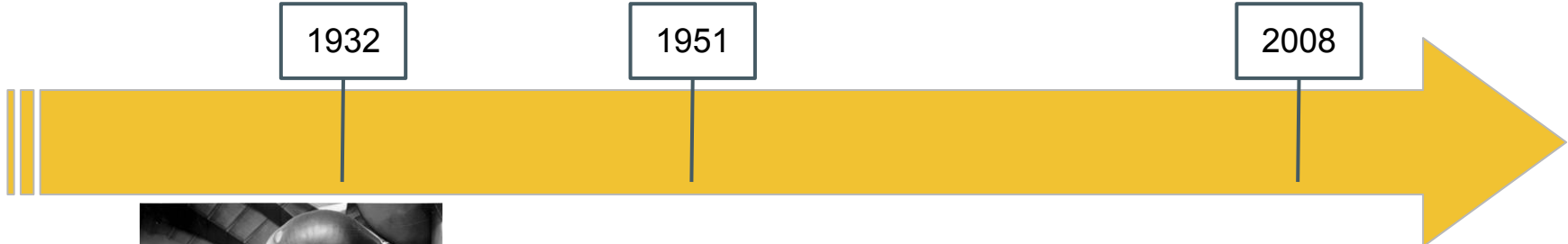
Ernest Thomas
Sinton Walton
Prize share: 1/2

The Nobel Prize in Physics 1951 was awarded jointly to Sir John Douglas Cockcroft and Ernest Thomas Sinton Walton "for their pioneer work on the transmutation of atomic nuclei by artificially accelerated atomic particles"

1932

1951

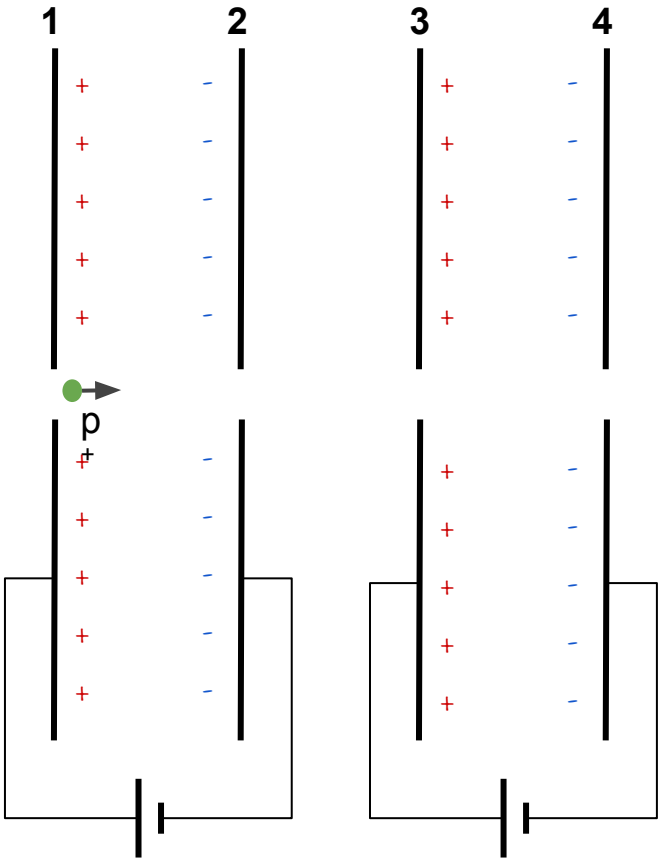
2008



Acceleratori elettrostatici?

- Cosa succederebbe se facessimo questo?
 - 1-2: è come prima

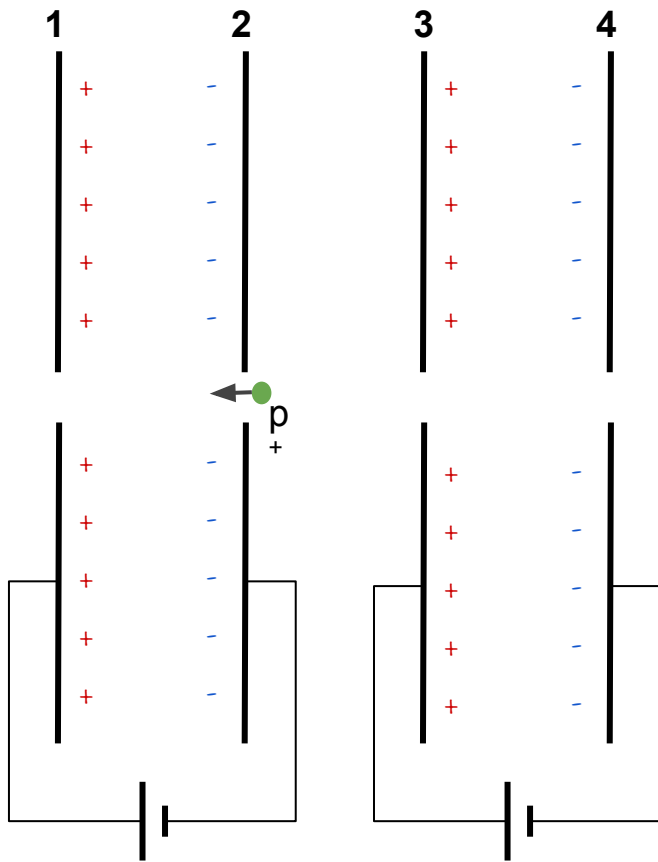
La sorgente di protoni è una bombola di idrogeno!



```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
/// This is a poin
const xAOD::EventI
/// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

Acceleratori elettrostatici?

La sorgente di protoni è una bombola di idrogeno!

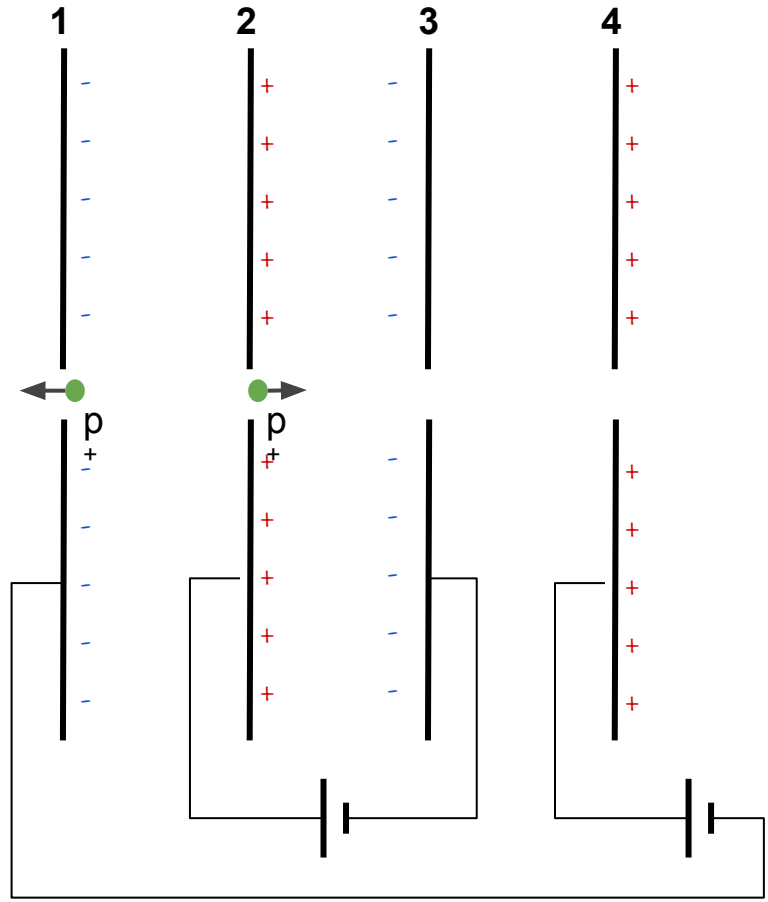


- Cosa succederebbe se facessimo questo?
 - 1-2: è come prima
 - 2-3: il protone viene frenato
 - devo invertire polarità!

```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAd
#include "xAODRootAd
#include "xAODRootAd
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

Acceleratori elettrostatici?

La sorgente di protoni è una bombola di idrogeno!

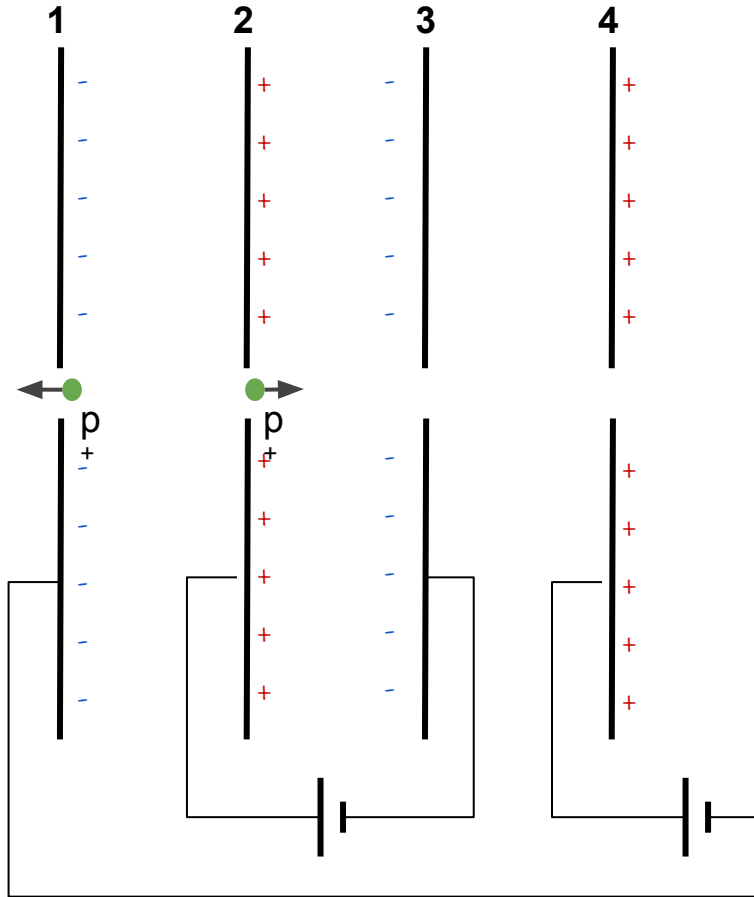


- Cosa succederebbe se facessimo questo?
 - 1-2: è come prima
 - 2-3: il protone viene frenato
 - devo invertire polarità!

```
<EventLoop/
<EventLoop/
<EventLoop/
Infrastructure in
"xAODCore/S
"xADDRootAc
"xADDRootAc
"xADDRootAc
"xAODEventI
<xAODTruth/
"xAODEgamma
"xAODMissir
"xAODMissir
"xAODMissir
"FourMomUti
"xAODBTagg
"xAODBTagg
"MuonSelect
"PathResolv
"TEfficienc
"TH1.h"
"TH1F.h"
"TH1D.h"
"Tree.h"
"File.h"
"Stopwatch
<iostream>
<iterator>
class PisaxAODAnalysis
// put your config
// that way they c
protected:
// float cutValue;
/// This is a poin
// This is a poin
const xAOD::EventI
/// This is a Missir
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

Acceleratori elettrostatici?

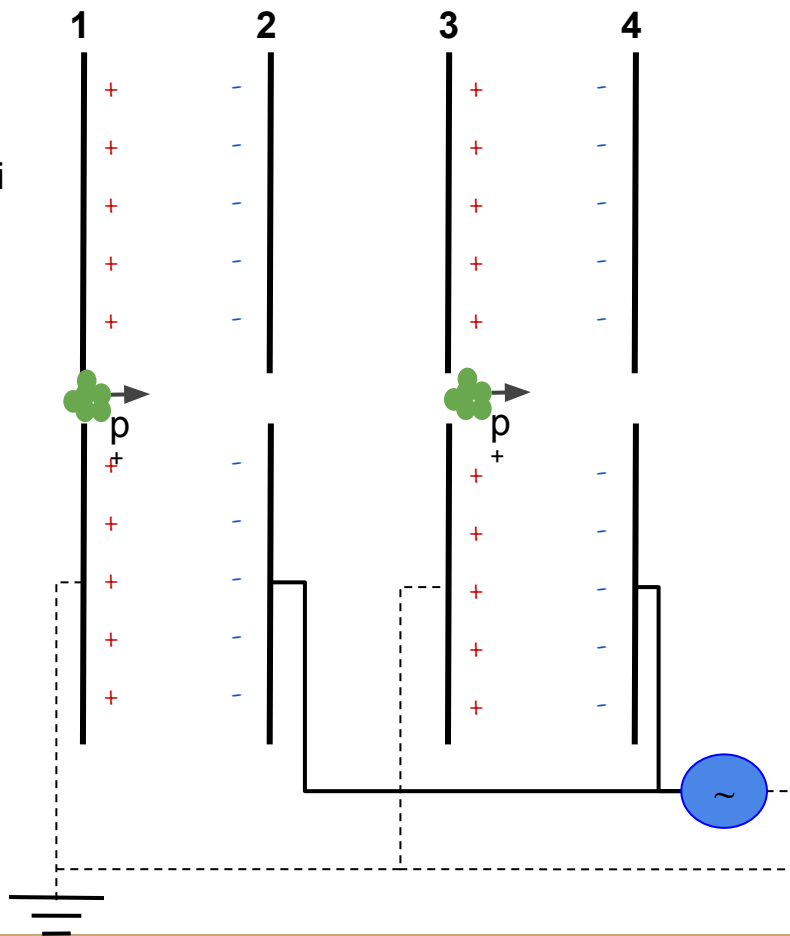
La sorgente di protoni è una bombola di idrogeno!



- Cosa succederebbe se facessimo questo?
 - 1-2: è come prima
 - 2-3: il protone viene frenato
 - devo invertire polarità!
- Cosa succede se metto un secondo protone in 1?
 - Verrebbe frenato
 - **I protoni non vengono accelerati in modo continuo**

Acceleratori elettrostatici?

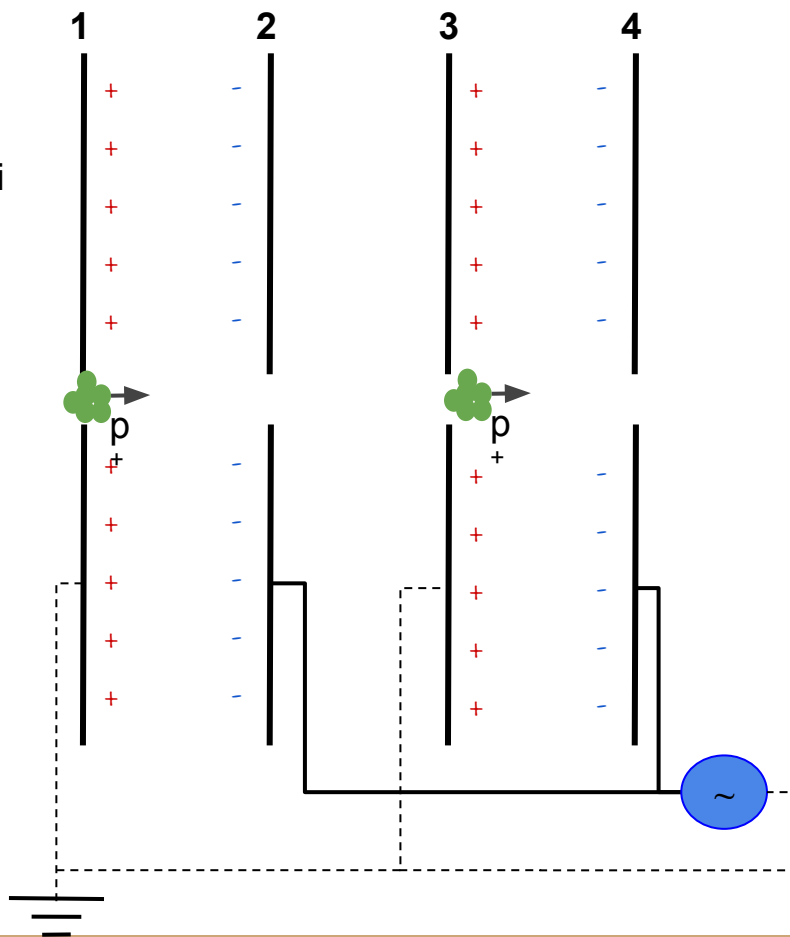
La sorgente di protoni è una bombola di idrogeno!



- Cosa succederebbe se facessimo questo?
 - 1-2: è come prima
 - 2-3: il protone viene frenato
 - devo invertire polarità!
- Cosa succede se metto un secondo protone in 1?
 - Verrebbe frenato
 - **I protoni vengono accelerati in pacchetti (“bunch”)**
 - Ogni buch contiene 10^{19} p^+

Acceleratori elettrostatici?

La sorgente di protoni è una bombola di idrogeno!



- Cosa succederebbe se facessimo questo?
 - 1-2: è come prima
 - 2-3: il protone viene frenato
 - devo invertire polarità!
- Cosa succede se metto un secondo protone in 1?
 - Verrebbe frenato
 - **I protoni vengono accelerati in pacchetti (“bunch”)**
 - Ogni buch contiene 10^{19} p^+
- Il tempo impiegato da un protone per muoversi tra due piastre dipende dalla sua velocità
 - più è alta, più le piastre saranno lontane, più sarò grande l’acceleratore

LINAC2: 50 MeV

```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTagg
#include "xAODBTagg
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnalysis
```



GUSTAF ISING

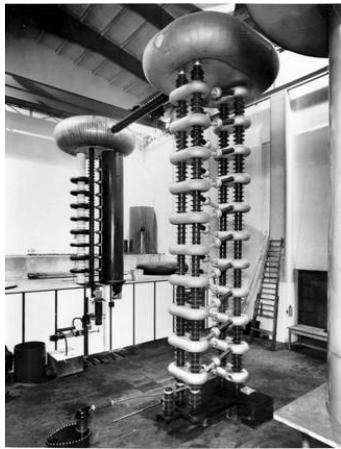
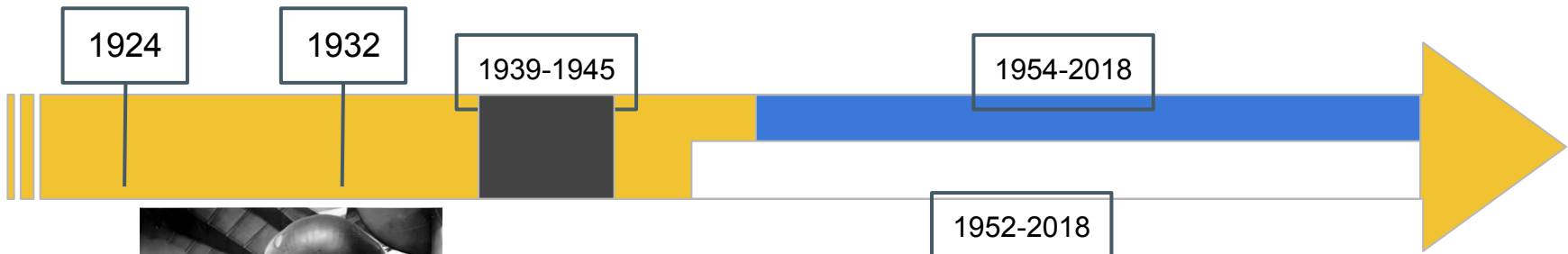
1924

1932

1939-1945

1954-2018

1952-2018



```
// put your config
// that way they c
protected:
// float cutValue;
/// This is a poin
const xAOD::EventI
/// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```



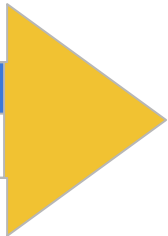

GUSTAF ISING

1924

1932

1978

2008



```

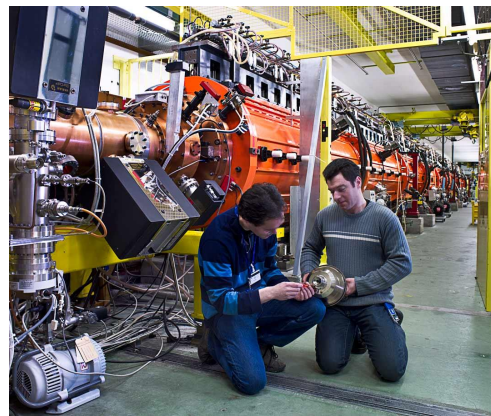
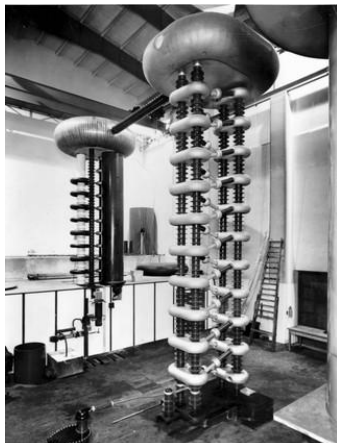
<EventLoop/
#include <EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnalysis

```

```

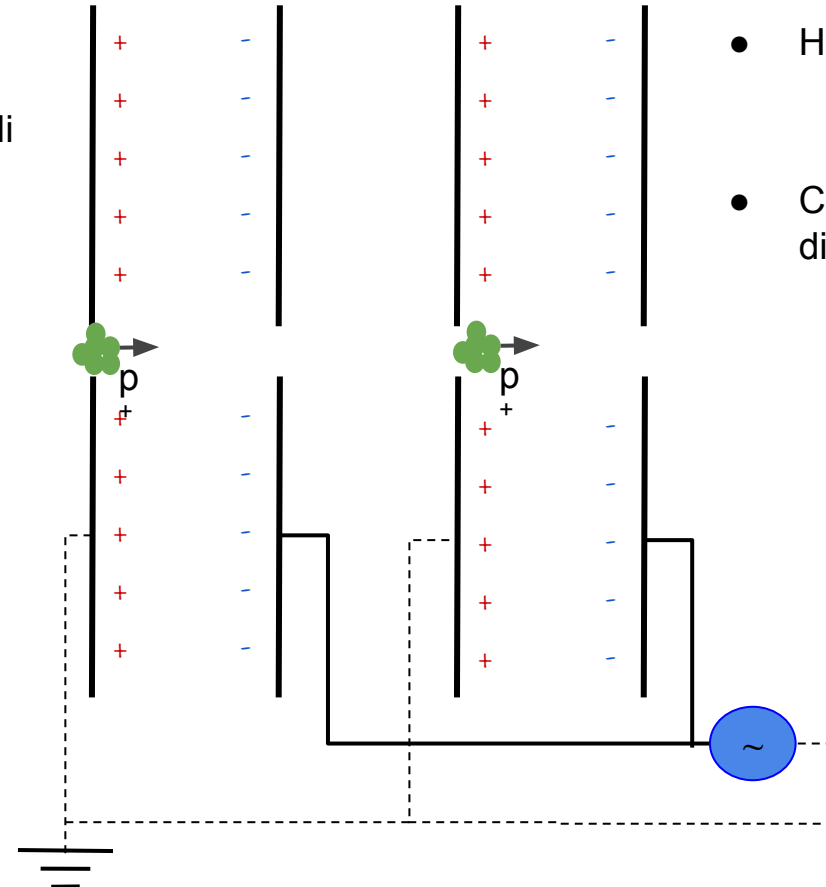
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (

```



Acceleratori elettromagnetici: cavità risonanti

La sorgente di protoni è una bombola di idrogeno!



- Ho una sorgente a radiofrequenza
 - genera un'onda elettromagnetica
 - che carica un conduttore.
- Cosa succederebbe se usassi direttamente l'onda elettromagnetica?

```
<!-- EventLoop -->
<!-- EventLoop -->
<!-- EventLoop -->
<!-- Infrastructure in -->
<!-- xAODCore/S -->
<!-- xAODRootAd -->
<!-- xAODRootAd -->
<!-- xAODRootAd -->
<!-- xAODEventI -->
<!-- xAODTruth/ -->
<!-- xAODEgamma -->
<!-- xAODMissir -->
<!-- xAODMissir -->
<!-- xAODMissir -->
<!-- FourMomUti -->
<!-- xAODBTaggi -->
<!-- xAODBTaggi -->
<!-- MuonSelect -->
<!-- PathResolv -->
<!-- TEfficienc -->
<!-- TH1.h -->
<!-- TH1F.h -->
<!-- TH1D.h -->
<!-- TTree.h -->
<!-- TFile.h -->
<!-- TStopwatch -->
<!-- iostream -->
<!-- iterator -->
class PisaxAODAnalysis
{
public:
    // put your configuration here
    // that way they can be protected:
    // float cutValue;
    // This is a pointer to the event
    const xAOD::Event* m_event;
    // This is a pointer to the event
    const xAOD::Missir* m_missir;
    BTaggingSelection* m_btagging;
    BTaggingEfficiency* m_btagging_efficiency;
    std::unique_ptr<CP
};

// variables that are protected from
// protected from
// node (done by the user)
public:
    // This is the event
    xAOD::Event* m_event;

// this is a standard
PisaxAODAnalysis (
```

Acceleratori elettromagnetici: cavità risonanti

La sorgente di protoni è una bombola di idrogeno!



- Ho una sorgente a radiofrequenza
 - genera un'onda elettromagnetica
 - che carica un conduttore.
- Cosa succederebbe se usassi direttamente l'onda elettromagnetica?
 - Uso **cavità risonanti a radio frequenza (400 MHz)**
- Genero potenziali di 2MV per cavità
- LHC ha 8 cavità per fascio
 - **16 MV orbita**

```
<EventLoop/
<EventLoop/
<EventLoop/
<EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
include "FourMomUti
include "xAODBTaggi
include "xAODBTaggi
include "MuonSelect
include "PathResolv
include "TEfficienc
include "TH1.h"
include "TH1F.h"
include "TH1D.h"
include "TTree.h"
include "TFile.h"
include "TStopwatch
include <iostream>
include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
// This is a pair
const xAOD::EventI
// This is a pair
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

Acceleratori elettromagnetici: cavità risonanti



```
<EventLoop/
<EventLoop/
<EventLoop/
<EventLoop/
Infrastructure in
"xAODCore/S
"xAODRootAc
"xAODRootAc
"xAODRootAc
"xAODEventI
<xAODTruth/
"xAODEgamma
"xAODMissir
"xAODMissir
"xAODMissir
"FourMomUti
"xAODBTagg
"xAODBTagg
"MuonSelect
"PathResolv
"TEfficienc
"TH1.h"
"TH1F.h"
"TH1D.h"
"Tree.h"
"File.h"
"Stopwatch
<iostream>
<iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
// This is a pair
const xAOD::EventI
// This is a pair
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

La sorgente di protoni è una bombola di idrogeno!

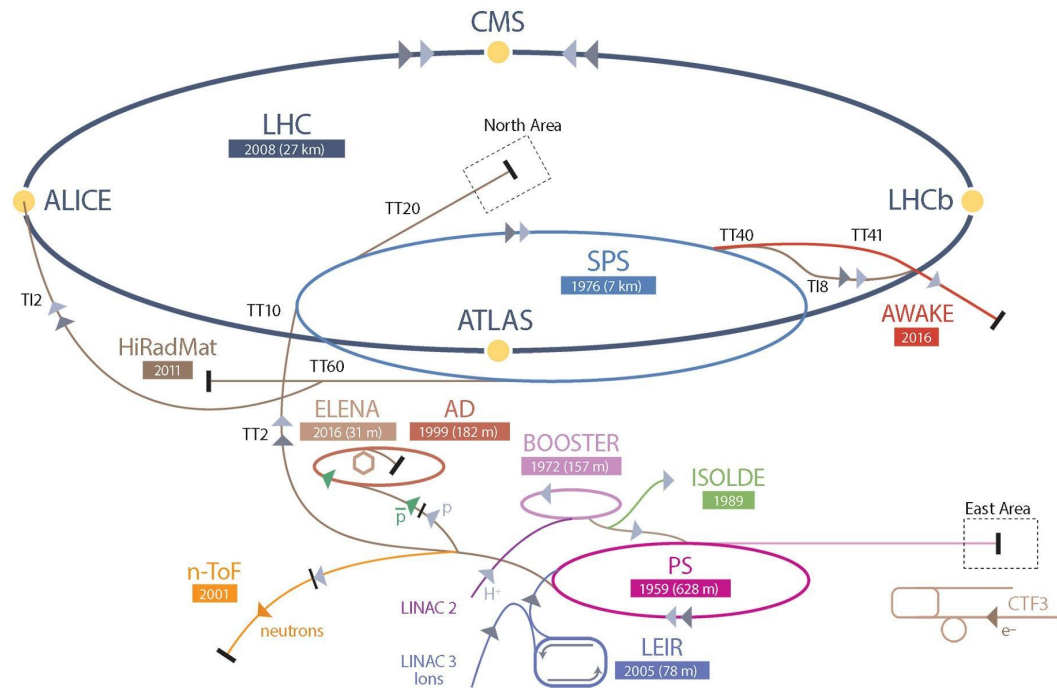


- Ho una sorgente a radiofrequenza
 - genera un'onda elettromagnetica
 - che carica un conduttore.
- Cosa succederebbe se usassi direttamente l'onda elettromagnetica?
 - Uso **cavità risonanti a radio frequenza (400 MHz)**
- Genero potenziali di 2MV per cavità
- LHC ha 8 cavità per fascio
 - **16 MV orbita**

Perchè ho delle orbite?

Cosa impareremo oggi?

CERN's Accelerator Complex



▶ p (proton) ▶ ion ▶ neutrons ▶ \bar{p} (antiproton) ▶ electron ▶ \leftrightarrow proton/antiproton conversion

Cosa rappresentano tutte quelle linee?

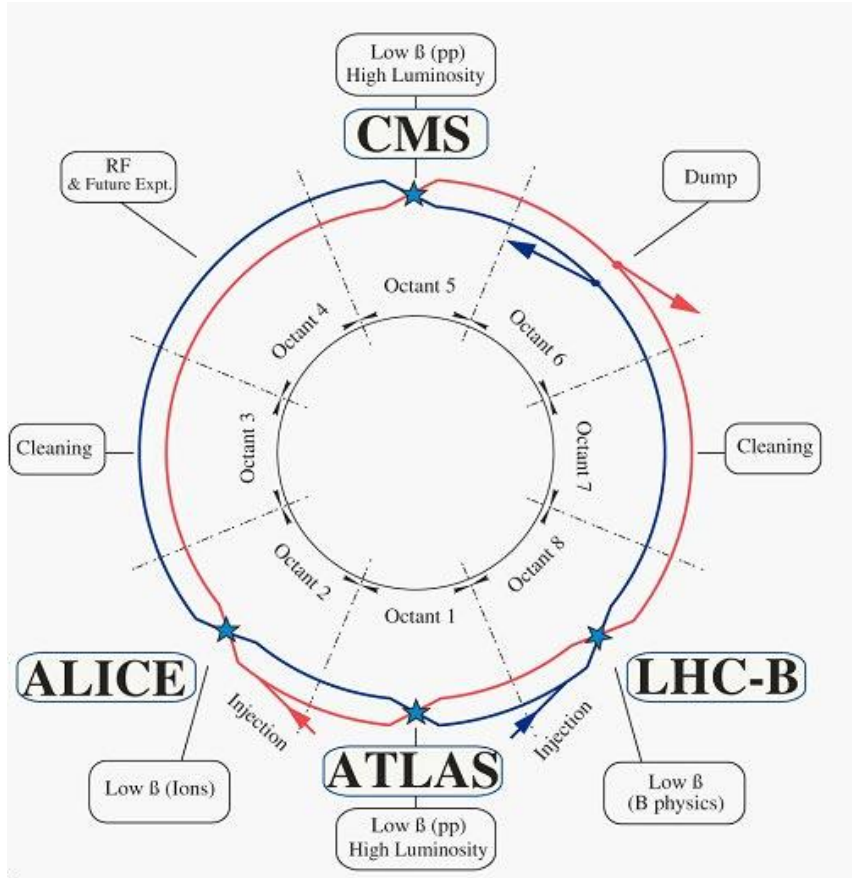
- Acceleratori
- Usano tecnologie differenti
- lineari
- circolari

Cosa sono i 'puntini gialli'?

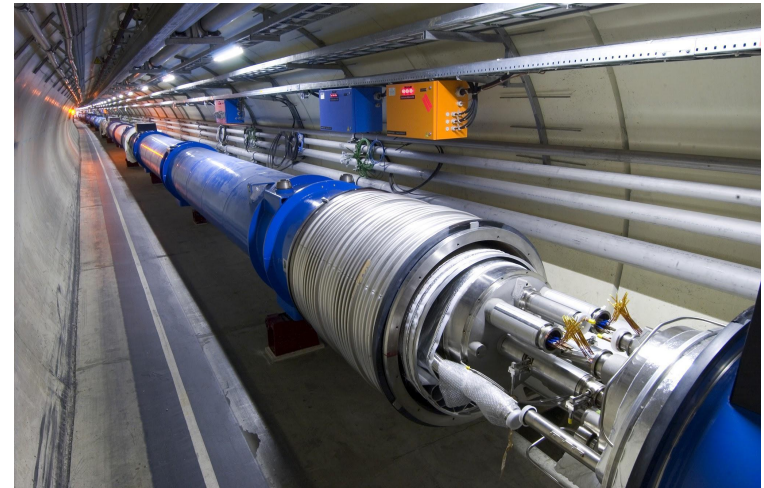
```

EventLoop/
include <EventLoop/
include <EventLoop/
include <EventLoop/
Infrastructure in
include "xAODCore/S
include "xAODRootAd
include "xAODRootAd
include "xAODRootAd
include "xAODEventI
include <xAODTruth/
include "xAODEgamma
include "xAODMissir
include "xAODMissir
include "xAODMissir
include "FourMomUti
include "xAODBTagg
include "xAODBTagg
include "xAODBTagg
include "MuonSelect
include "PathResolv
include "TEfficien
include "TH1.h"
include "TH1F.h"
include "TH1D.h"
include "TTree.h"
include "TFile.h"
include "TStopwatch
include <iostream>
include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
    
```

Acceleratori circolari

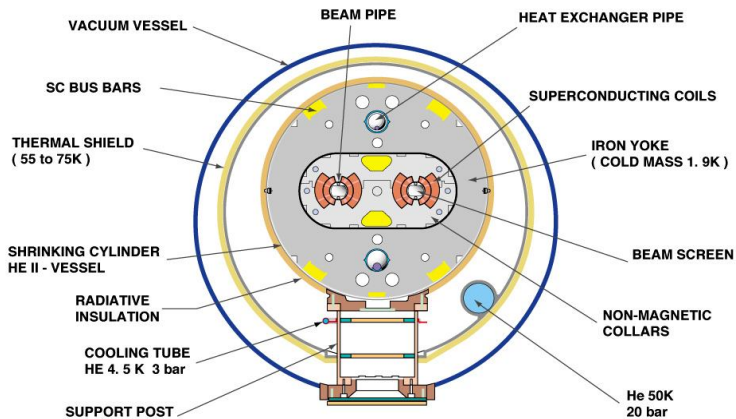


- Ad ogni giro le particelle vengono spinte.
- Ho Grandi range di energia:
 - **Booster:** 50 MeV-1.45 GeV
 - **Proton Synchrotron:** 1.45 GeV - 5.9 GeV
 - **Super Proton Synchrotron:** 5.9 GeV - 450 GeV
 - **Large Hadron Collider:** 450 GeV - 7 TeV
- Ho bisogno di un campo magnetico per far girare i protoni:

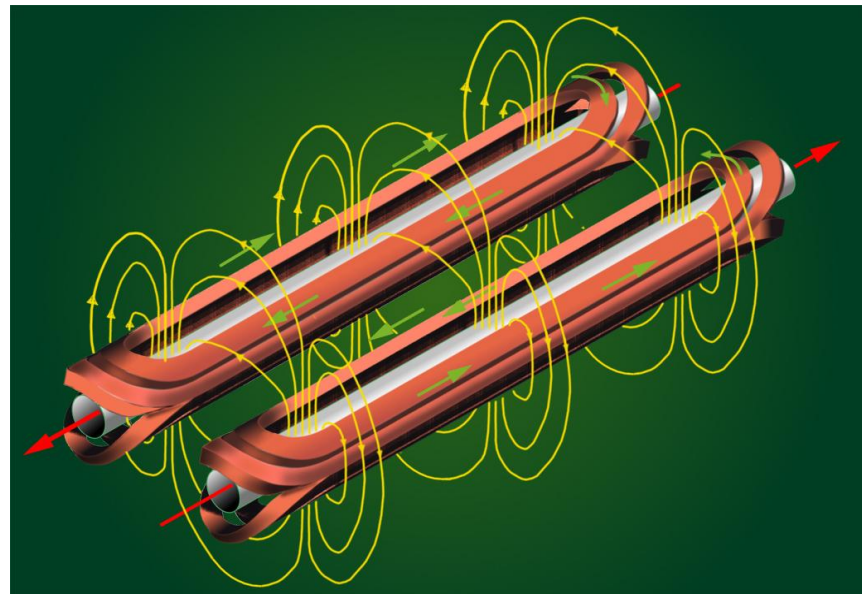


Dipoli magnetici

CROSS SECTION OF LHC DIPOLE

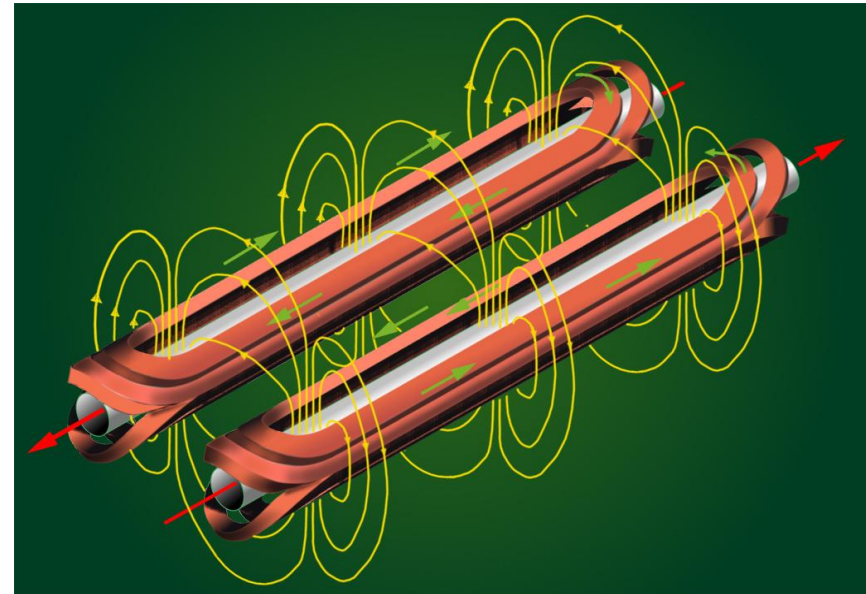
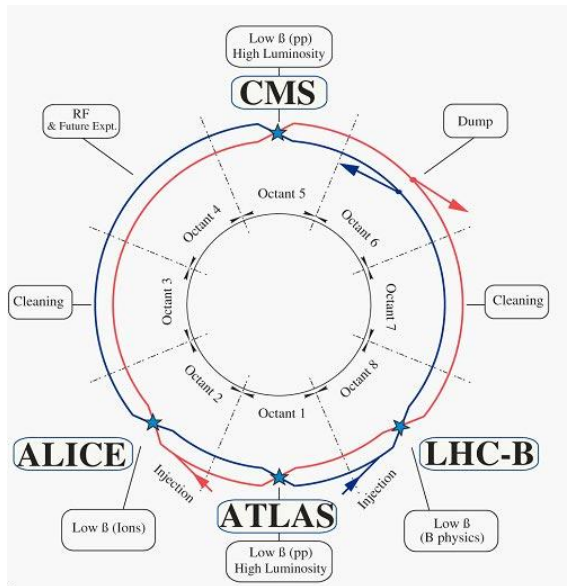


CERN AC_HE107A_V02/02/98



- 1.600 magneti superconduttori in lega di niobio e titanio
 - raffreddati alla temperatura di 1,9 K (-271,25 °C) da elio liquido superfluido
 - campo magnetico di 8.38 tesla
 - è il limite della tecnologia esistente
 - fissa le dimensioni/energia dell'acceleratore.
- Consumo di LHC ~ 80 MW
 - se i magneti non fossero superconduttivi ~ 200 GW

Dipoli magnetici



- Ogni magnete ha 2 “beam pipe”
 - Ho due fasci di protoni che girano in senso opposto
 - I due fasci si incrociano solo in 4 punti di interazione dove collidono
 - **Questi acceleratori prendono il nome di Collider**
 - Esistono anche esperimenti con bersaglio fisso (fixed target)
- **L’energia nel centro di massa è doppia rispetto a quella dei fasci!**

Dipoli magnetici... sappiamo fare i conti?

$$\mathbf{F} = q(\mathbf{v} \times \mathbf{B})$$

$$a = \frac{v^2}{R} = \frac{F}{m_p} = \frac{q(vB)}{m_p}$$

$$R = \frac{m_p}{q} \frac{v}{B}$$

Parto dalla forza di Lorentz in assenza di campo elettrico

I protoni si muovono di moto circolare uniforme, v e B sono ortogonali

Semplifico

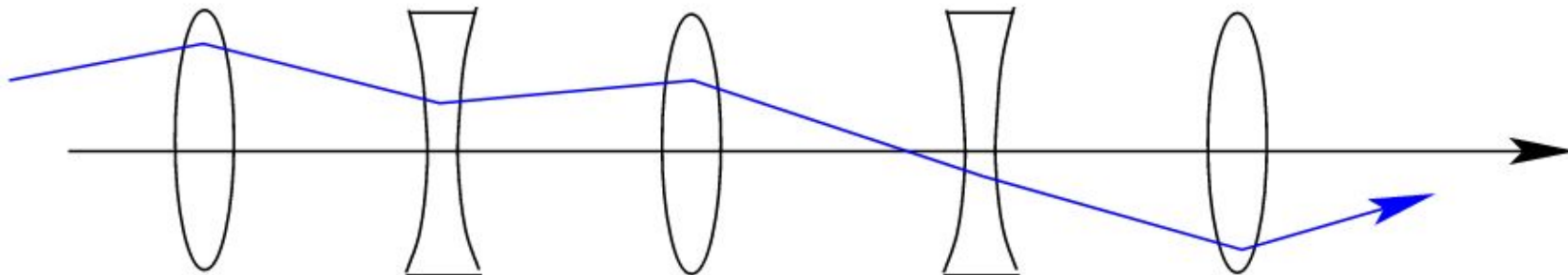
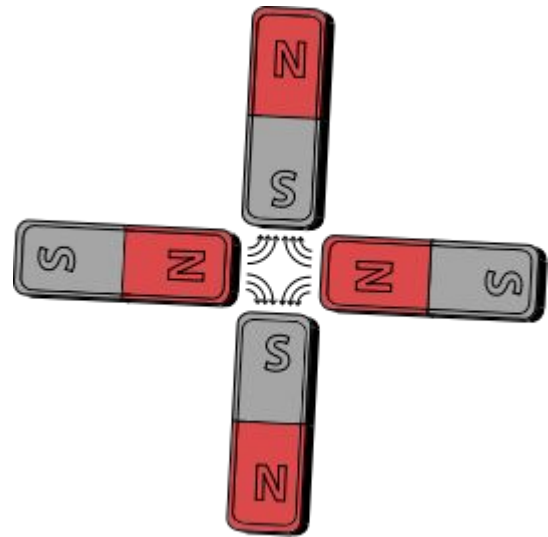
Non è vero!!



Posso fare un acceleratore senza magneti?

NO!

- I bunch di protoni tendono ad aprirsi per via della repulsione elettromagnetica tra i protoni stessi
- Ho bisogno di focalizzare i fasci!
 - Uso dei quadrupoli magnetici
 - “schiacciano” il bunch in una direzione
 - “allargano” il bunch nella direzione opposta



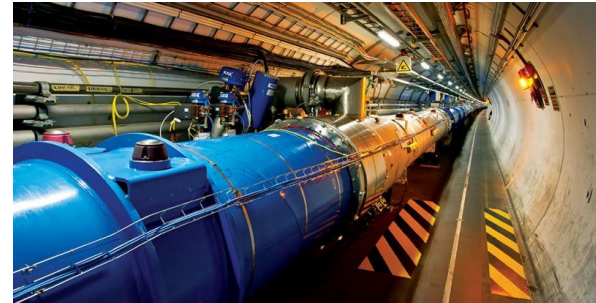
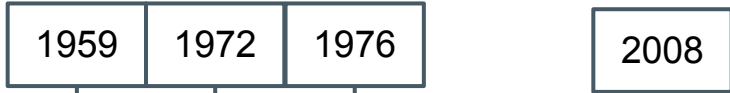
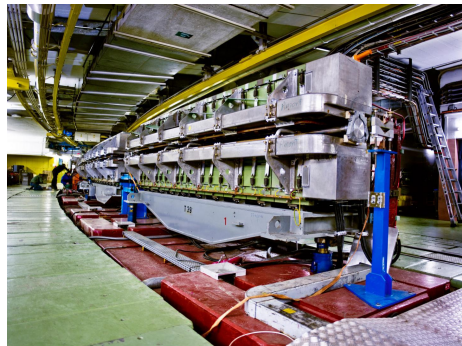
```
<EventLoop/
<EventLoop/
<EventLoop/
Infrastructure in
<Include "xAODCore/S
<Include "xAODRootAc
<Include "xAODRootAc
<Include "xAODRootAc
<Include "xAODEventI
<Include <xAODTruth/
<Include "xAODEgamma
<Include "xAODMissir
<Include "xAODMissir
<Include "xAODMissir
<Include "FourMomUti
<Include "xAODBTaggi
<Include "xAODBTaggi
<Include "MuonSelect
<Include "PathResolv
<Include "TEfficienc
<Include "TH1.h"
<Include "TH1F.h"
<Include "TH1D.h"
<Include "TTree.h"
<Include "TFile.h"
<Include "TStopwatch
<Include <iostream>
<Include <iterator>
class PisaxAODAnalysis
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

```
<EventLoop/
<EventLoop/
<EventLoop/
<EventLoop/
Infrastructure in
<Include "xAODCore/S
<Include "xAODRootAc
<Include "xAODRootAc
<Include "xAODRootAc
<Include "xAODEventI
<Include <xAODTruth/
<Include "xAODEgamma
<Include "xAODMissir
<Include "xAODMissir
<Include "xAODMissir
<Include "FourMomUti
<Include "xAODBTaggi
<Include "xAODBTaggi
<Include "MuonSelect
<Include "PathResolv
<Include "TEfficienc
<Include "TH1.h"
<Include "TH1F.h"
<Include "TH1D.h"
<Include "TTree.h"
<Include "TFile.h"
<Include "TStopwatch
<include <iostream>
<include <iterator>
class PisaxAODAnaly
```

```
// put your config
// that way they c
protected:
// float cutValue;
/// This is a poin
const xAOD::EventI
/// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP

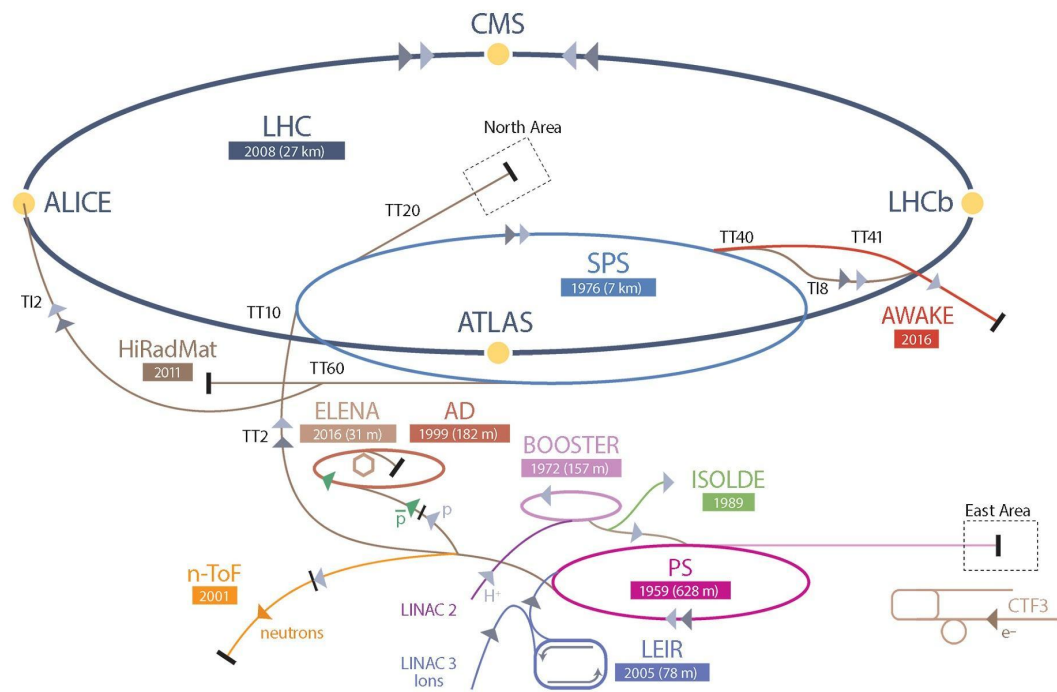
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev

// this is a stand
PisaxAODAnalysis (
```



Cosa Impareremo oggi?

CERN's Accelerator Complex



Cosa rappresentano tutte quelle linee?

- Acceleratori
- Usano tecnologie differenti
- Possono essere lineari o circolari
- Tutti usano magneti
 - dipoli
 - quadrupoli

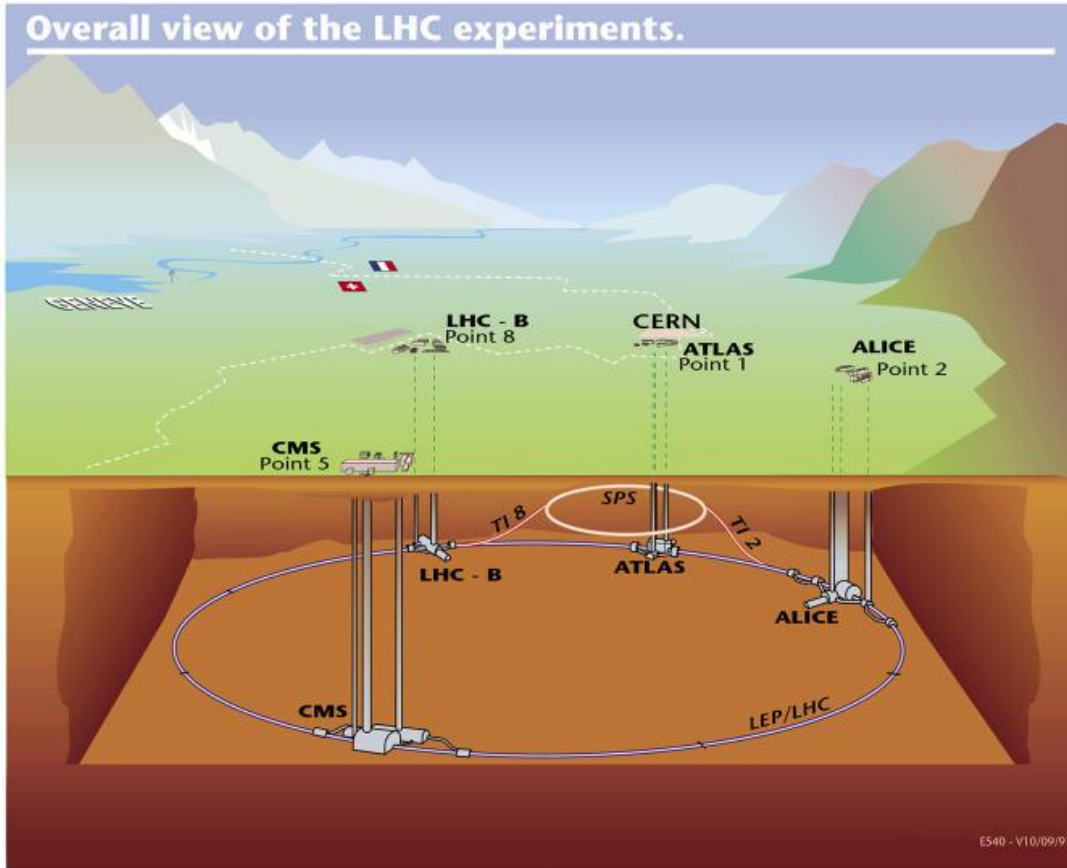
Cosa sono i 'puntini gialli'?

▶ p (proton) ▶ ion ▶ neutrons ▶ \bar{p} (antiproton) ▶ electron ▶ $\rightarrow\rightarrow$ proton/antiproton conversion

```

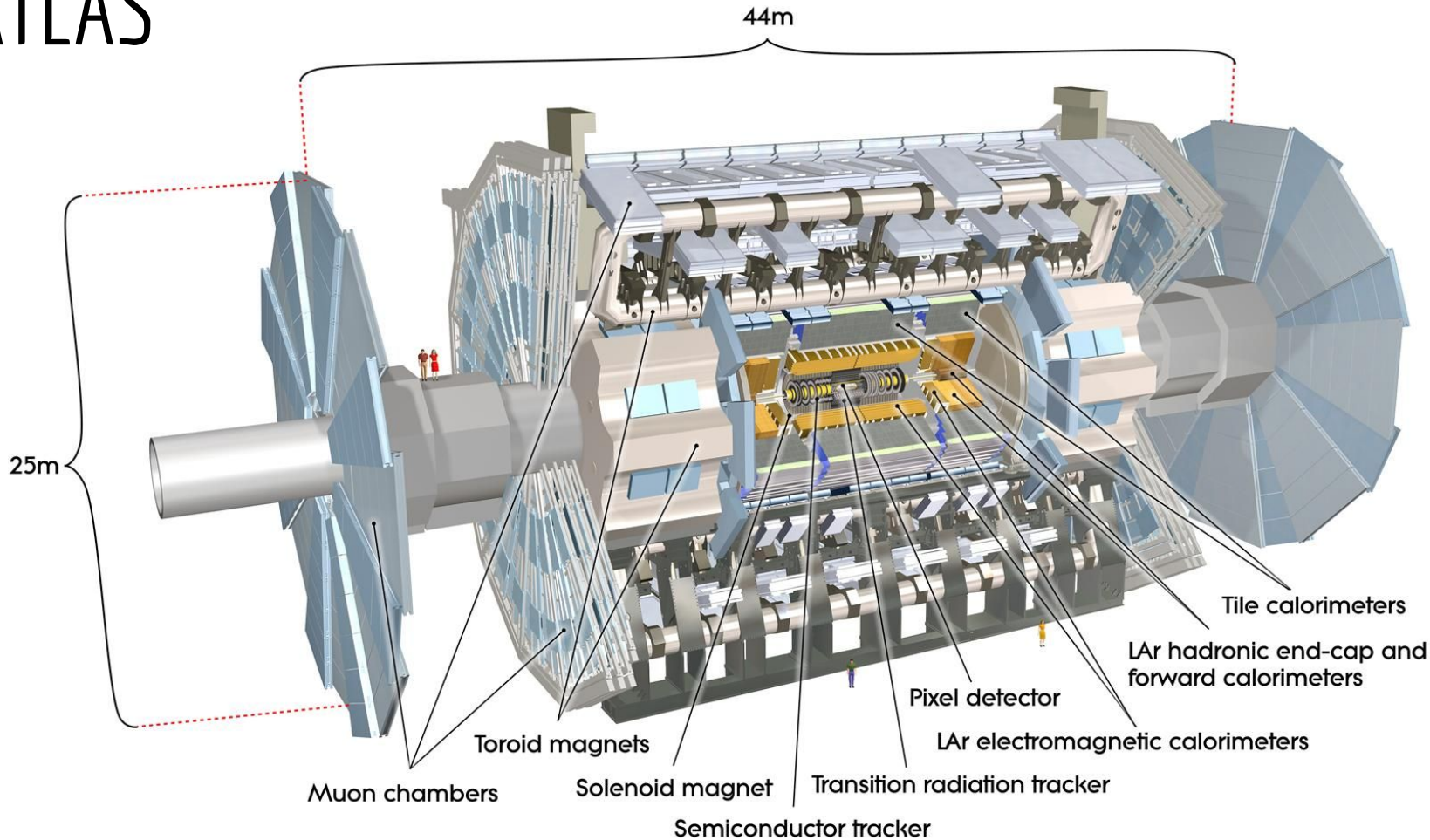
#include <EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAd
#include "xAODRootAd
#include "xAODRootAd
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTagg
#include "xAODBTagg
#include "MuonSelect
#include "PathResolv
#include "TEfficien
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include <TStopwatch
#include <iostream>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
    
```

Rivelatori di particelle



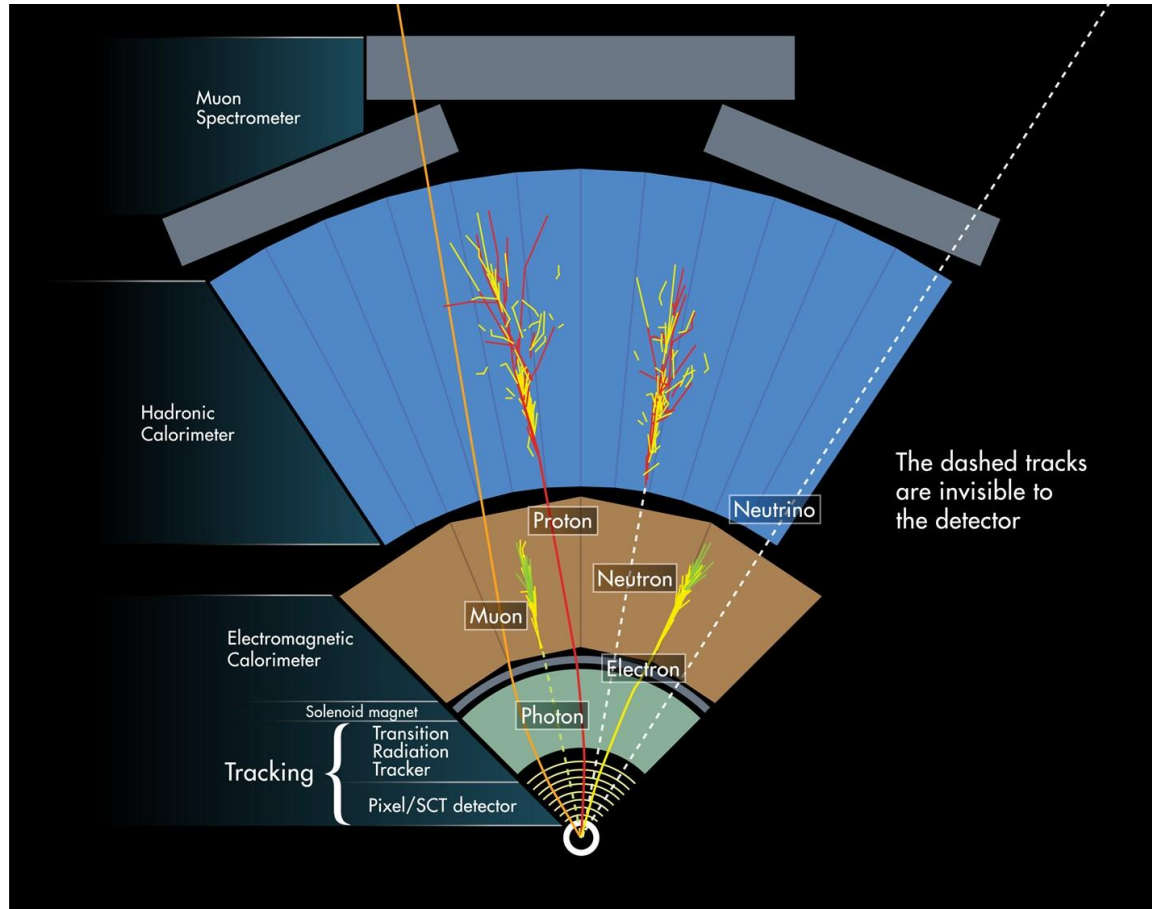
- I principali detector che operano lungo l'anello di LHC sono 4:
 - ALICE (A Large Ion Collider Experiment)
 - ATLAS (A Toroidal LHC Apparatus)
 - CMS (Compact Muon Solenoid)
 - LHCb (LHC b)
- Si trovano nei 4 punti di interazione
- Gli esperimenti sono molto diversi
- Analizziamo le caratteristiche comuni descrivendo **ATLAS**

ATLAS



Quali particelle vedo?

```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```



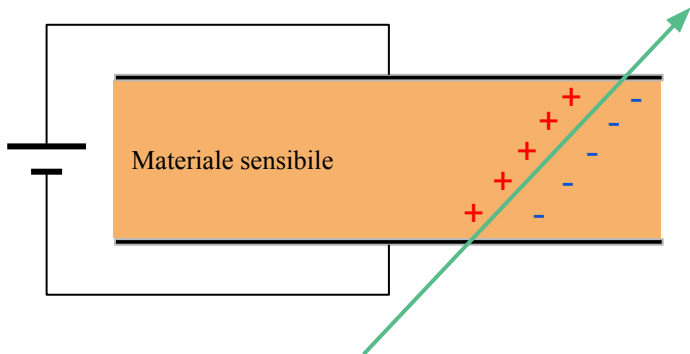
The dashed tracks are invisible to the detector

Terminologia di base... in 30"

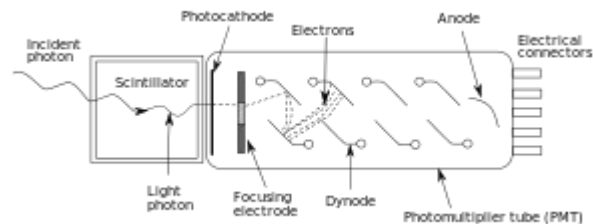
Misuriamo:

- La posizione di una particella (solitamente basati sulla ionizzazione)
 - tracciatori
- L'energia della particella (basati su entrambi i principi)
 - Calorimetri

Ionizzazione
(genero/raccolgo cariche elettriche)



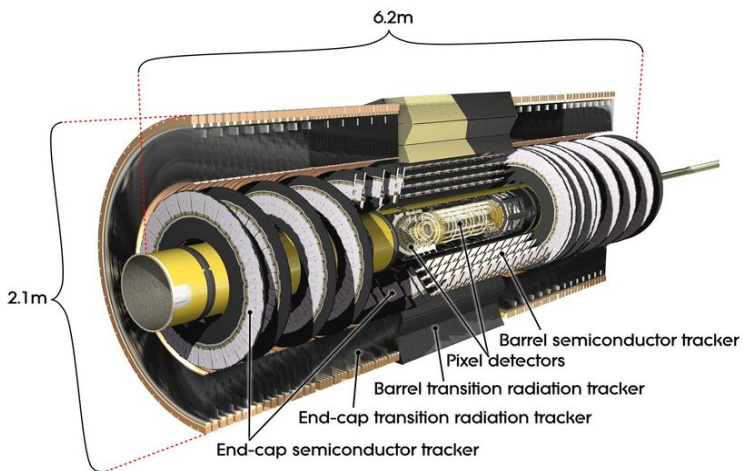
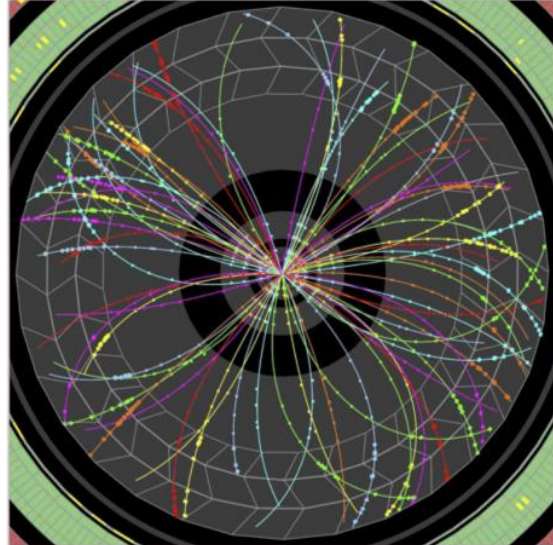
Scintillazione
(genero luce che converti in una carica elettrica)



Tracciatori

Misuriamo la posizione di una particella

- Vicini alla regione di interazione
 - ho meno disturbi dovuti all' interazione con il rivelatore
- Immersi in un campo magnetico
 - ne misuro il momento: $R = \frac{m_p}{q} \frac{v}{B}$



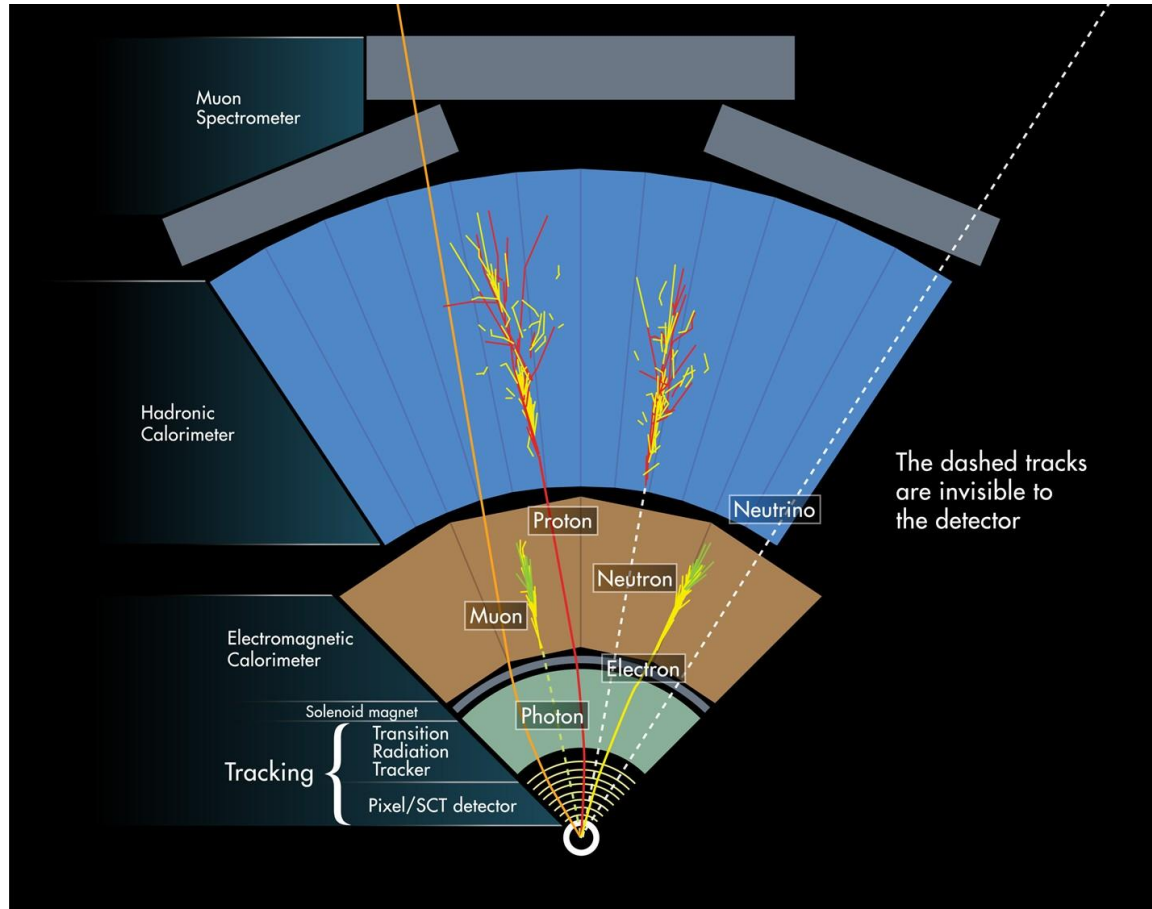
- Spesso più di due protoni interagiscono tra loro.



- Danno origine a quello che chiamiamo pile-up
- Due parametri importanti sono
 - il numero di interazione per bunch crossing
 - la luminosità

Quali particelle lasciano una traccia?

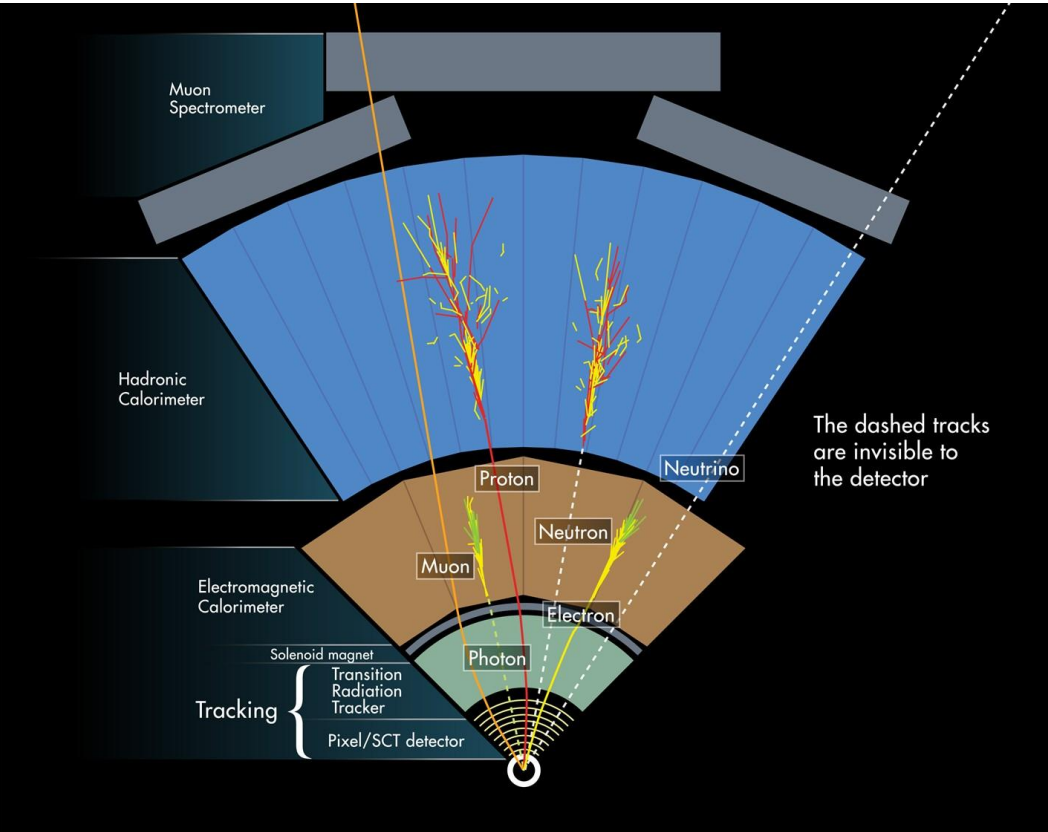
```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAd
#include "xAODRootAd
#include "xAODRootAd
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```



The dashed tracks are invisible to the detector

Cosa succede dopo il tracciatore?

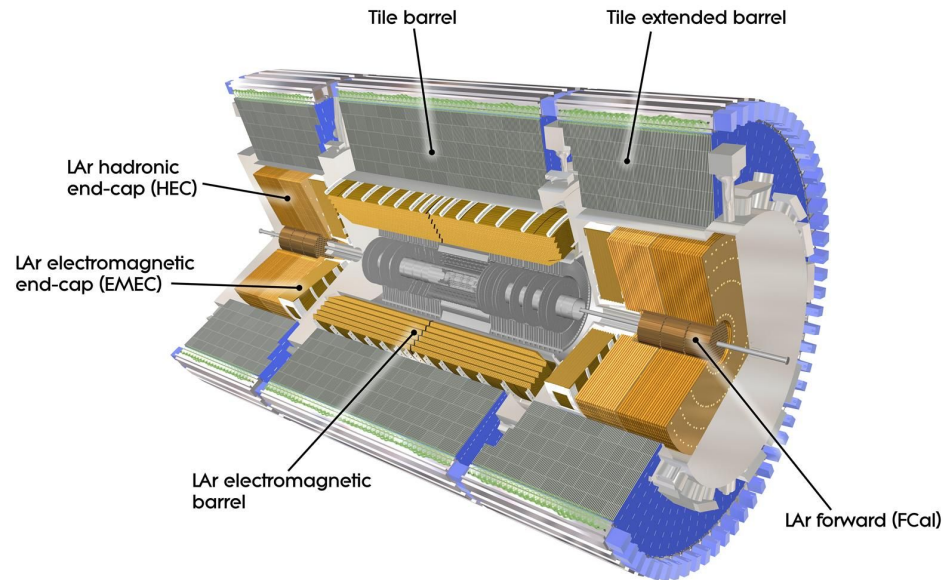
```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAd
#include "xAODRootAd
#include "xAODRootAd
#include "xAODEventT
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnalysis
// put your config
// that way they c
protected:
// float cutValue;
/// This is a pair
const xAOD::EventI
/// This is a pair
const xAOD::Missir
BTaggingSelectionI
BTaggingEfficiency
std::unique_ptr<CF
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```



- Le interazioni tra le particelle ed il rivelatore diventano dominanti
 - producono delle particelle secondarie
 - queste a loro volta producono secondari
- Ho una **shower!**
 - shower elettromagnetiche (elettroni, fotoni)
 - shower adroniche (protoni, neutroni, ...)
- **Larghezza, lunghezza e forma della shower dipendono dalla particella che le ha originate**

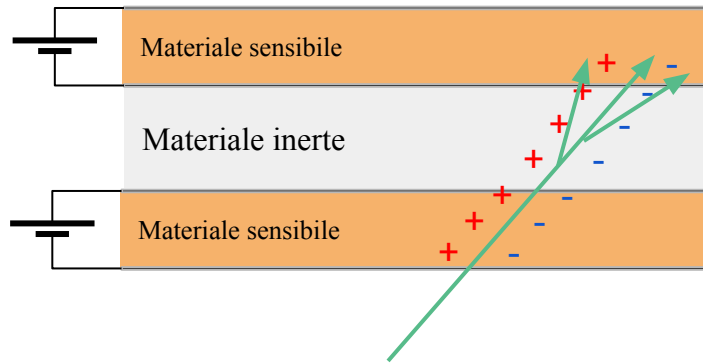
Come misuro l'energia di una particella?

- Devo misurare l'energia della shower... con un calorimetro!
- Cerco di avere una shower molto "concentrata"
- Richiede materiali diversi per
 - shower elettromagnetiche
 - shower adroniche
- Ho due calorimetri:
 - **calorimetro elettromagnetico**
 - alto Z
 - Piombo
 - **calorimetro adronico**
 - Alto A
 - Ferro

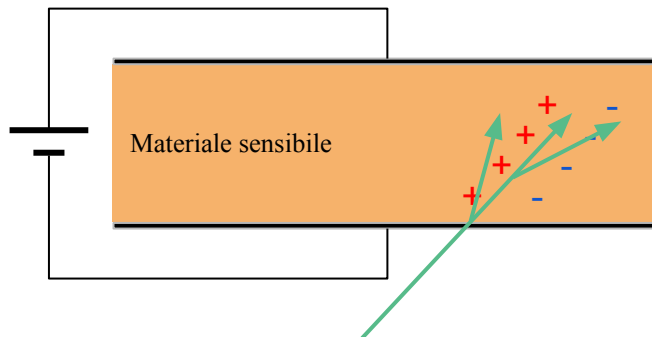


Come misuro l'energia di una particella?

- I calorimetri di ATLAS sono a “campionamento”



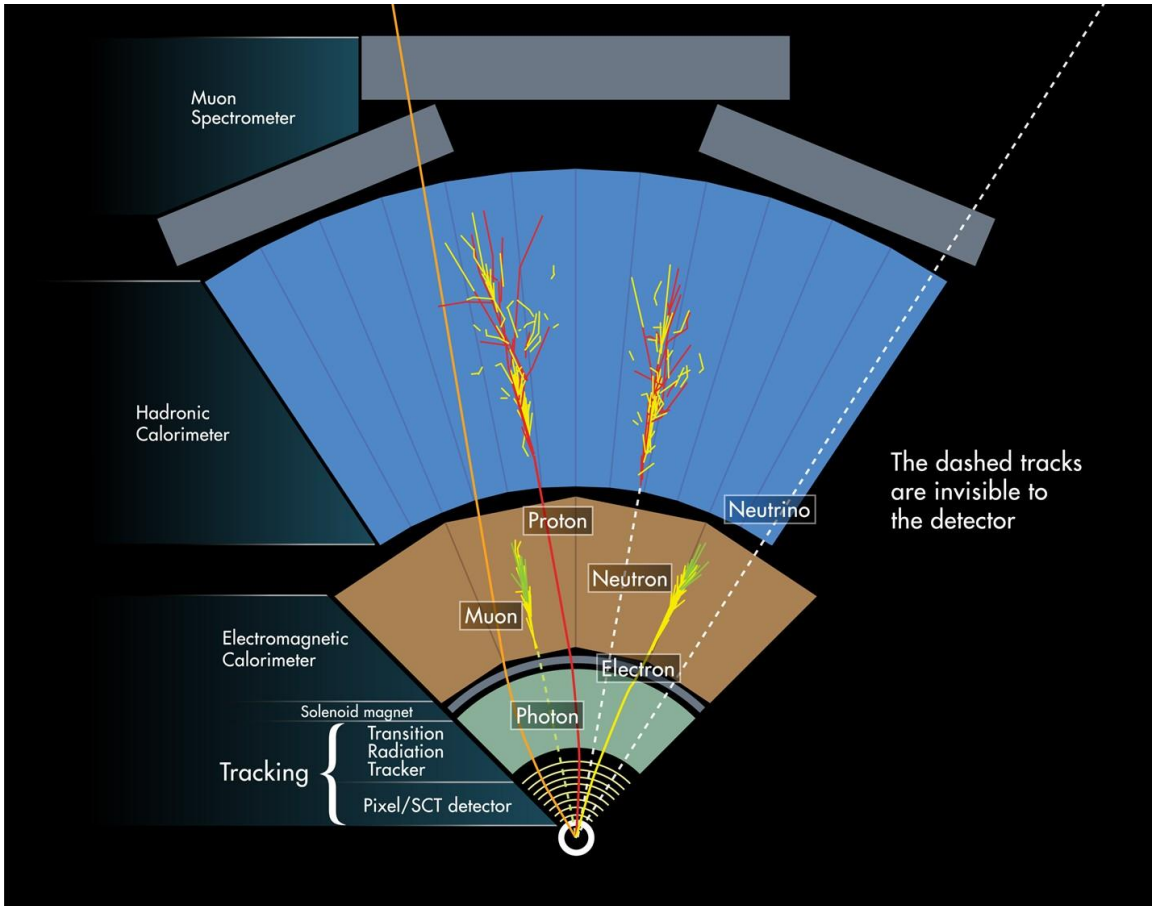
- Esistono anche calorimetri uniformi



```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAd
#include "xAODRootAd
#include "xAODRootAd
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnaly
// put your config
// that way they c
protected:
// float cutValue;
/// This is a pair
const xAOD::EventI
/// This is a pair
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```

Ho misurato tutte le particelle?

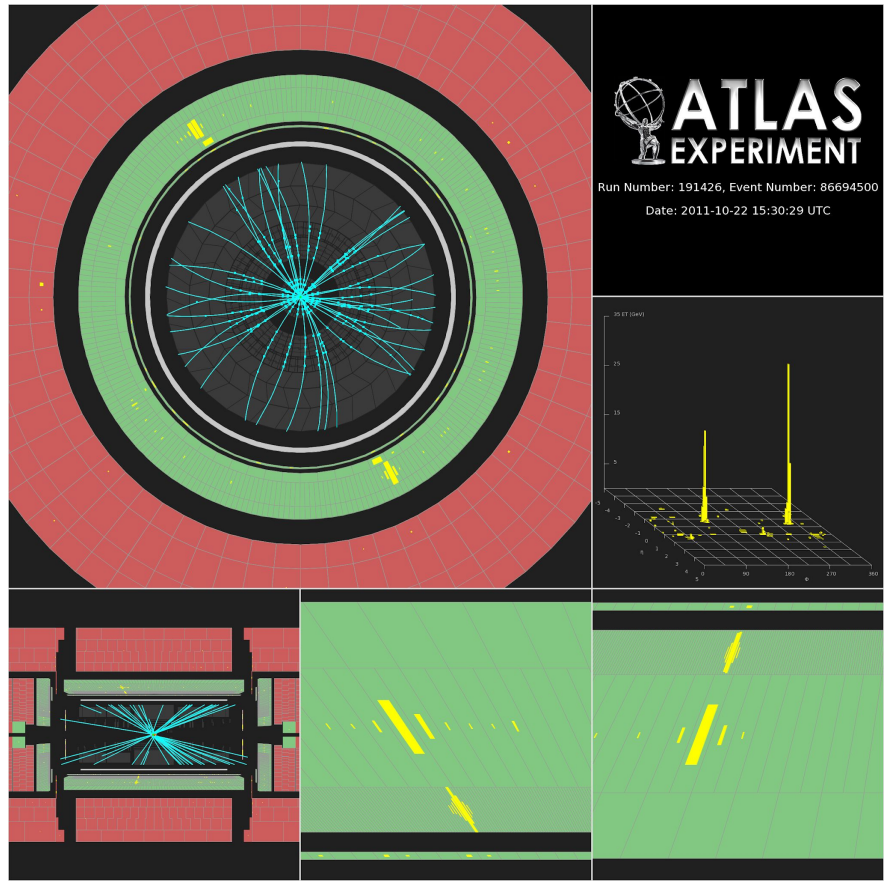
```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
Infrastructure in
#include "xAODCore/S
#include "xAODRootAc
#include "xAODRootAc
#include "xAODRootAc
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnalyser
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```



The dashed tracks are invisible to the detector

Come è fatto un evento di Higgs in due fotoni?

```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
/ Infrastructure in
#include "xAODCore/S
#include "xAODRootAd
#include "xAODRootAd
#include "xAODRootAd
#include "xAODEventI
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTaggi
#include "xAODBTaggi
#include "MuonSelect
#include "PathResolv
#include "TEfficienc
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnalysis
// put your config
// that way they c
protected:
// float cutValue;
// This is a poin
const xAOD::EventI
// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP
// variables that
// protected from
// node (done by t
public:
// This is the ev
xAOD::TEvent* m_ev
// this is a stand
PisaxAODAnalysis (
```



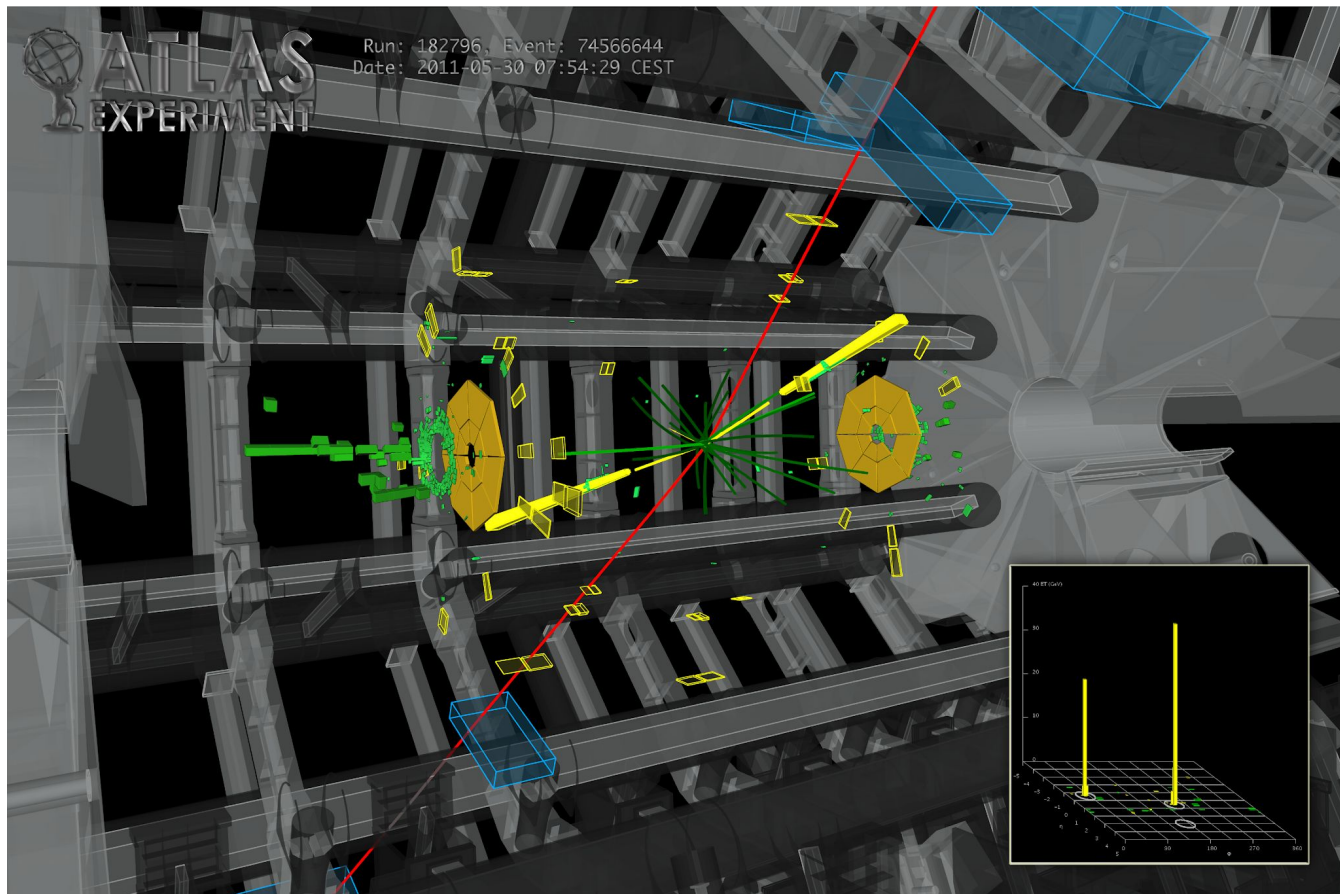
Come è fatto un evento di Higgs in 4 leptoni?

```
<EventLoop/
#include <EventLoop/
#include <EventLoop/
/ Infrastructure in
#include "xAODCore/S
#include "xAODRootAd
#include "xAODRootAd
#include "xAODRootAd
#include "xAODEventT
#include <xAODTruth/
#include "xAODEgamma
#include "xAODMissir
#include "xAODMissir
#include "xAODMissir
#include "FourMomUti
#include "xAODBTagg
#include "xAODBTagg
#include "MuonSelect
#include "PathResolv
#include "TEfficien
#include "TH1.h"
#include "TH1F.h"
#include "TH1D.h"
#include "TTree.h"
#include "TFile.h"
#include "TStopwatch
#include <iostream>
#include <iterator>
class PisaxAODAnlys

// put your config
// that way they c
protected:
// float cutValue;
/// This is a poin
const xAOD::EventI
/// This is a poin
const xAOD::Missir
BTaggingSelectionT
BTaggingEfficiency
std::unique_ptr<CP

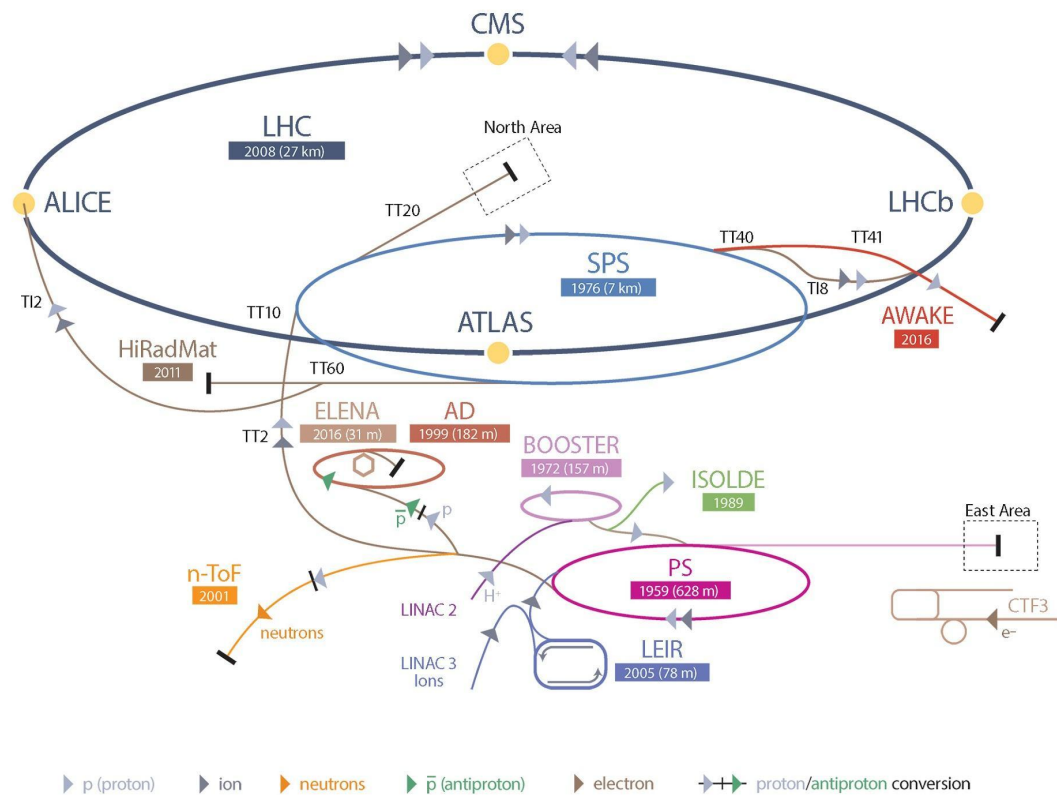
// variables that
// protected from
// node (done by t
public:
/// This is the ev
xAOD::TEvent* m_ev

// this is a stand
PisaxAODAnalysis (
```



Cosa abbiamo imparato oggi?

CERN's Accelerator Complex



Cosa rappresentano tutte quelle linee?

- Acceleratori
- Usano tecnologie differenti
- Possono essere lineari o circolari
- Tutti usano magneti
 - dipoli
 - quadrupoli
- Cosa si intende per luminosità

Cosa sono i 'puntini gialli'?

- Rivelatori di particelle
- Usano tecnologie differenti
- Usano molti sotto-detector per misurare
 - tracce
 - energia
- Che cos'è il pile-up