

# Table of Contents

<b>A.2 Troubleshooting Guide: Understanding and Solving Common Software Issues.....</b>	<b>1</b>
Compiler Errors.....	1
Error: expected `;' before.....	1
Error: expected `}' at end of input.....	2
Error: 'XXX' is not a member of 'yyy'.....	2
Error: undefined reference.....	3
Python Errors.....	4
Error: 'Process' object has no attribute 'XXX'.....	4
Error: No module named XXX.....	4
Error: XXX does not already exist.....	5
Error: more than 255 arguments.....	5
Runtime Errors.....	6
Error: Failed to open the file.....	6
Error: Unable to find plugin 'XXX'.....	7
Error: No "JetCorrectionsRecord" record found in the EventSetup.....	7
Error: Found zero products matching all criteria.....	8
Error: InvalidReference.....	9
Useful tools for debugging your code.....	10
Review status.....	10

## A.2 Troubleshooting Guide: Understanding and Solving Common Software Issues

Programming and especially using CMSSW is a complicated thing. And even though the quest of common tools such as Physics Analysis Toolkit (PAT) is to simplify things they are still far from trivial to use it. When you start using PAT, CMSSW or just start programming there is a bunch of problems you will always encounter. During a normal evolution process you start with parse and linking errors during compile time and via the mysteries of python errors finally arrive at CMSSW complaints during runtime. Gaining the experience to recognize these errors is an important but painful process. This page should give you some assistance to cope with very typical errors that always occur when beginners get started.

**Note:** By construction this page is evolving with time. We will try to add more and more typical errors with increasing experience.

### Compiler Errors

In modern software architectures not all source code, that is used is compiled into a single very heavy executable. It is rather split into a set of clearly structured pre-compiled shared library object files ( `.so`). These shared library object files will be loaded dynamically when really needed during the compilation of the actual executable. Consequently the compilation of a new `cmsRun` executable consists of two parts:

- **Parsing:**

The grammar of the source code you wrote is checked for its correctness. The compiler checks that you did not forget any key signs (like '#', or ';'), that all functions and their arguments are known and understood and their return values match to the expected types. The declaration of new functions can be found in the header files ( `.h`) files that were added in the implementation of the source code.

- **Linking:**

Some functions that were used in the source code might be implemented in different pre-compiled shared library objects ( `.so`). In this step all `.so` files needed for the compilation of the executable are searched and linked.

In CMSSW all packages are compiled into individual `.so` files. When ever you added a `.h` file in your source code to declare a function you did not implement yourself the probability is high that you will also have to make the implementation of this function known to the compiler by pointing to the corresponding `.so` file. We list a few parsing and linker errors below:

#### Error: expected `;' before...

For the error below we commented a ';' from line 104 of the `PatBasicAnalyzer.cc` source file of the `PatExamples` package:

```
...
>> Entering Package PhysicsTools/PatExamples

>> Compiling /afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/
/afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/PatExamples/p
/afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/PatExamples/p
...
```

**Note:** This is a simple error. The compiler explains you exactly what going in. The error occurs for line 107 (and not in line 104, which was modified by us) as there the next statement starts.

- **What's going on?**

The compiler found a statement that was not finished by a ';'.

- **What happened?**

You added a few lines of code to your module and forgot the ';' at the end of the input line.

- **What is the remedy?**

Add the ';' at the end of the indicated line.

In our example we uncommented the ';' in line 104 of the *PatBasicAnalyzer.cc* source file of the *PatExamples* package:

```
histContainer_["jets"]-&gt;Fill(nJets);
```

## Error: expected `}' at end of input...

For the error below we commented a '}' from line 103 of the *PatBasicAnalyzer.cc* source file of the *PatExamples* package:

```
...
>> Compiling /afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/
/afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/PatExamples/p
/afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/PatExamples/p
/afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/PatExamples/p
...
afs/cern.ch/cms/slc5_ia32_gcc434/external/gcc/4.3.4/bin/./lib/gcc/i686-pc-linux-gnu/4.3.4/./././
/afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/PatExamples/p
gmake: *** [tmp/slc5_ia32_gcc434/src/PhysicsTools/PatExamples/plugins/PhysicsToolsPatExamples_plu
...

```

**⚠ Note:** This error can be tricky as during the parsing a lost opening '{' or closing '}' brace may still lead to code that might be syntactically correct, but in the first places changes its meaning. Only after a few lines the grammar parsing fails and leads to this error. In our case of a missing '}' latest at the end a characteristic error occurred.

- **What's going on?**

The compiler found a missing '}', the following lines mit grammatically still be OK or lead to first parse errors. Latest at the end of the file the missing '}' beomes apparent.

- **What happened?**

You got mixed up with opening and closing braces. At the end you forgot one of them.

- **What is the remedy?**

This error is nasty as it is not always clear where it appears. The easiest way is to follow the indentation of your editor and in hard cases to comment and uncomment parts of the code step by step.

In our example we uncommented the '}' in line 103 of the *PatBasicAnalyzer.cc* source file of the *PatExamples* package:

## Error: 'XXX' is not a member of 'yyy'...

For the error below we commented the .h file for the declaration of pat::Muons from the *PatBasicAnalyzer.cc* in the *PatExamples* package:

Error: expected `;' before...

...

```
>> Entering Package PhysicsTools/PatExamples
```

```
>> Compiling /afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/
```

```
/afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/PatExamples/p
```

```
/afs/cern.ch/user/r/rwolf/scratch0/CMSSW_3_8_X_2010-09-12-1300/src/CMS.PhysicsTools/PatExamples/p
```

...

**Note:** This error equally holds for any other objects that you might have introduced in your code without declaring them to the compiler: reco::Muons, edm::InputTags or pat::Jets, ...

- **What's going on?**

The compiler found an object that has never been declared when parsing your source file. In our example this is the pat::Muon.

- **What happened?**

You added an object to your code which is not known a priori to the compiler. You have to make the declaration known in a .h file.

- **What is the remedy?**

Add the corresponding .h file that contains the declaration of the object you defined in your source code.

In our example we uncommented the corresponding .h file again:

```
#include "DataFormats/PatCandidates/interface/Muon.h"
```

## Error: undefined reference

All .so files needed to compile the executable are defined in the CMS.BuildFile of your package or corresponding subdirectories. The complete name of each corresponding .so file inherits from the name of the package and the subsystem the referred function is declared and implemented in. If you add the corresponding .so file of this package the error should be gone.

Some plugins in the *PatExamples* package make use of function that have been defined in the *PatExamples* package itself. For the error given below we commented the .so file of the *PatExamples* package from the CMS.BuildFile located in the *plugins* directory of the *PatExamples* package:

...

```
tmp/slc5_ia32_gcc434/src/PhysicsTools/PatExamples/plugins/PhysicsToolsPatExamples_plugins/libPhys
```

```
collect2: ld returned 1 exit status
```

```
gmake: *** [tmp/slc5_ia32_gcc434/src/PhysicsTools/PatExamples/plugins/PhysicsToolsPatExamples_plu
```

...

**Note:** This is a rare error, unless you start a project from scratch the BuildFiles you are using are pretty complete.

- **What's going on?**

The linker does not find the implementation of a function that you use in your code.

- **What happened?**

You used a function somewhere in your code and added the .h file where the declaration of the

function can be found. You forgot to make the `.so` file known to the compiler in which the implementation (located in the `.cc` file in the corresponding package) is defined.

- **What is the remedy?**

Add the corresponding `.so` file in the `CMS.BuildFile` of your package. A safe way is to check for all packages that you used in your source code to declare your objects from the start on.

In our example we uncommented the corresponding `.so` file again:

```
<use name="CMS.PhysicsTools/PatExamples"/>
```

## Python Errors

The common configuration language for `cmsRun` is `python`. Have a look at `WorkBookConfigFileIntro` to learn more about the structure of `cmsRun` configuration files. Have a look at this tutorial [to learn more about python itself](#). Before starting `cmsRun` we recommend you to parse your configuration file for its syntactical correctness using the `python` interpreter. The way to do this is shown below for the examples of a typical configuration file in the `PatExamples` package:

```
python PhysticsTools/PatExamples/test/patTuple_standard_cfg.py
```

A set of typical python errors is given below:

### Error: 'Process' object has no attribute 'XXX'

```
Traceback (most recent call last):
  File "CMS.PhysicsTools/PatExamples/test/analyzeTopSelection_cfg.py", line 137, in
    process.step7
AttributeError: 'Process' object has no attribute 'step7'
```

 **Note:** For this error we introduced a typo in line 67 to the `analyzeTopSelection_cfg.py` configuration file of the more complex analysis example in the `PatExamples` package.

- **What's going on?**

At least one module is not know to the `python` interpreter when checking the process path of the configuration file.

- **What happened?**

Most probably you introduced a typo in the definition of your modules or in the process path or you introduced a module in the path that you forgot to define before. Note that module definitions might also be hidden in imported file.

- **What is the remedy?**

The best cure might be to check all module definitions for their consistency. The linux tool `grep` might be of help for you.

In our example we corrected the typo in line 67:

```
process.step7 = countPatJets.clone(src = 'goodJets' , minNumber = 4)
```

### Error: No module named XXX

```
Traceback (most recent call last):
  File "CMS.PhysicsTools/PatExamples/test/analyzeTopSelection_cfg.py", line 35, in
    process.load("CMS.PhysicsTools.PatExamples.topObjectSelection")
  File "/afs/naf.desy.de/group/cms/sw/slc5_ia32_gcc434/cms/cmssw/CMSSW_3_8_2/python/FWCore/Parame
```

Error: undefined reference

```
module = __import__(moduleName)
ImportError: No module named topObjectSelection
```

**Note:** For this error we introduced a typo in line 35 to the `analyzeTopSelection_cfg.py` configuration file of the more complex analysis example in the *PatExamples* package. We omitted the `"_cff"` in the `load` command.

- **What's going on?**

There is a file specified an `import` or `load` statement that does not exist.

- **What happened?**

Either you introduced a file via `import` or `load` that does not exist, or you introduced a typo in the file name or path.

- **What is the remedy?**

Check the path and name of the corresponding file in your working directory. Note that *python* directories are always omitted in the configuration file.

In our example we corrected the typo in line 35:

```
process.load("CMS.PhysicsTools.PatExamples.topObjectSelection:cff")
```

## Error: XXX does not already exist

Traceback (most recent call last):

```
File "CMS.PhysicsTools/PatExamples/test/analyzeTopSelection_cfg.py", line 67, in
  process.step7 = countPatJets.clone(src2 = 'goodJets' , minNumber = 4)
File "/afs/naf.desy.de/group/cms/sw/slc5_ia32_gcc434/cms/cmssw/CMSSW_3_8_2/python/FWCore/Parameterizable__raiseBadSetAttr(key)
File "/afs/naf.desy.de/group/cms/sw/slc5_ia32_gcc434/cms/cmssw/CMSSW_3_8_2/python/FWCore/Parameterizable__raiseBadSetAttr(key)
  raise TypeError(name+" does not already exist, so it can only be set to a CMS python configuration type")
TypeError: src2 does not already exist, so it can only be set to a CMS python configuration type
```

**Note:** Again for this error we introduced a typo in line 67 to the `analyzeTopSelection_cfg.py` configuration file of the more complex analysis example in the *PatExamples* package.

- **What's going on?**

At least one parameter of which you wanted replace a value does not exist in the definition of the module.

- **What happened?**

Either you introduced a typo, when specifying the parameter name or you really try to replace a parameter that does not exist. in rare cases it happens that the module definition indeed changed under the hood from release to release.

- **What is the remedy?**

Check with the original definition of the module in the corresponding *cfi* file.

In our example we corrected the typo in line 67:

```
process.step7 = countPatJets.clone(src = 'goodJets' , minNumber = 4)
```

## Error: more than 255 arguments

```
...
Problem with configuration file analyser_cfg.py ---- Configuration BEGIN
python encountered the error: more than
```

Error: No module named XXX

```
255 arguments (analyser_cfg.py, line 11) ---- Configuration END
...
```

**Note:** This is a typical error that occurs if you have more than 255 input files for your source module.

- **What's going on?**

The python interpreter encountered a vector with more than 255 input arguments.

- **What happened?**

You introduced a vector or list in your python configuration file which has more than 255 arguments.

- **What is the remedy?**

this is known short coming of python. You can circumvent it though following some tricks as indicated on SWGuidePythonTips#Running\_on\_more\_than\_255\_files.

## Runtime Errors

Runtime errors only occur when you execute *cmsRun*. In the following we are dealing with errors that occur in the use of *cmsRun* and in the interplay of different module in the communication among each other or with the event content. These are errors that can not be traced by syntactical checks of the compiler or *python* interpreter. we summarise a few very typical errors and their remedy below:

### Error: Failed to open the file

```
%MSG-s CMSException: AfterFile 15-Sep-2010 13:06:41 CEST pre-events
cms::Exception caught in cmsRun
---- FileOpenError BEGIN
---- StorageFactory::open() BEGIN
Failed to open the file 'wjets/patTuple_0_wjets10_madAODs.root' because:
---- File::sysopen() BEGIN
open() failed with system error 'No such file or directory' (error code 2)
---- File::sysopen() END
---- StorageFactory::open() END
```

RootInputFileSequence::initFile(): Input file file:wjets/patTuple\_0\_wjets10\_madAODs.root was not found or could not be opened.

Error occurred while creating source PoolSource ---- FileOpenError END %MSG

**Note:** For this error we specified an input file which does not exist.

- **What's going on?**

The *cmsRun* executable cannot open the specified input file.

- **What happened?**

Either the input file you specified does not exist, or you introduced a typo. Also take care that *cmsRun* needs the keyword `file:` to specify that the given file is to be specified locally.

- **What is the remedy?**

Check the file name and file location.

We give a typical example for a locally stored input file

```
process.source = cms.Source("PoolSource", fileNames = cms.untracked.vstring( 'file:wjets/patTuple
```

## Error: Unable to find plugin 'XXX'

```
%MSG-s CMSException: AfterModConstruction 15-Sep-2010 13:30:37 CEST pre-events
cms::Exception caught in cmsRun
---- PluginNotFound BEGIN
Unable to find plugin 'PatTopSelectionAnalyser'. Please check spelling of name.
---- PluginNotFound END
%MSG
```

**⚠ Note:** For this error we introduced a typo in the module definition of the `PatTopSelectionAnalyzer_cfi.py` file of the more complex analysis example in the *PatExamples* package. We replaced the 'z' by an 's' in the first argument, which specifies the corresponding plugin "*PatTopSelectionAnalyzer*".

- **What's going on?**

In the configuration file a module is called for which no plugin exists.

- **What happened?**

Either you introduced a module in your configuration which really has no correspondence in the list of plugins (this can happen if you forgot to compile the package you are working with) or you introduced a typo what specifying it.

- **What is the remedy?**

Check the plugin name you were really out for. The first step would be to `grep` in what module definition the plugin is specified. Next you can check whether this module really has no correspondence in the list of plugins (in our example typing `edmPluginDump | grep PatTopSelectionAnalyser` would do the trick). Finally you can check the correctness when looking into the plugin registry step (in our example indicated in the last line of `PatTopSelectionAnalyzer.cc`)

We summarise again the steps to trace this error down:

```
grep -r PatTopSelectionAnalyser PhysicsTools/PatExamples
CMS.PhysicsTools/PatExamples/python/PatTopSelectionAnalyzer_cfi.py:analyzePatTopSelection = cms.E
Binary file PhysicsTools/PatExamples/python/PatTopSelectionAnalyzer_cfi.pyc matches
edmPluginDump | grep PatTopSelectionAnalyser
emacs PhysicsTools/PatExamples/plugins/PatTopSelectionAnalyzer.cc
```

## Error: No "JetCorrectionsRecord" record found in the EventSetup

```
cms::Exception caught in cmsRun
---- EventProcessorFailure BEGIN
EventProcessingStopped
---- ScheduleExecutionFailure BEGIN
ProcessingStopped
---- NoRecord BEGIN
No "JetCorrectionsRecord" record found in the EventSetup.
Please add an ESSource or ESProducer that delivers such a record.
```

**⚠ Note:** This error can occur in two ways. The most direct way is indeed that you want to make use of a jet energy corrections service in you code, but did not specify it in the configuration file. In the transition from `CMSSW_3_6_X` to `CMSSW_3_8_X` it might also happen that you specified the proper jet energy correction service but did not use the correct `CMS.GlobalTag`. As a transition of the jet energy correction constants from bare *txt* files to the conditions data base took place between these two release cycles even when you specified the proper jet energy correction service you will get the same error in this special case.

- **What's going on?**

In one of your modules you try to use a jet energy correction service, which is not known to *cmsRun*.

- **What happened?**

Either you added a jet correction service in on of your modules but did not specify it in your configuration file or (in rare cases) you did everything right but do not use the proper *CMS.GlobalTag*.

- **What is the remedy?**

Specify the jet energy correction service in you configuration file or move to the porper *CMS.GlobalTag*.

We give the example of specifying the correct jet energy correction service in your configuration file, as this might be the most common source of this error. Therefore add the following lines to you configuration file:

```
## load jet corrections process.load("JetMETCorrections.Configuration.JetCorrectionServicesAllAlgo")
```

The exact choice of the correction service might depend on the jet collections you are going to use. Please have a look to SWGuideJetMet, or contact the corresponding experts for more details.

## Error: Found zero products matching all criteria

```
cms::Exception caught in cmsRun
---- EventProcessorFailure BEGIN
EventProcessingStopped
---- ScheduleExecutionFailure BEGIN
ProcessingStopped
---- ProductNotFound BEGIN
getByLabel: Found zero products matching all criteria
Looking for sequence of type: reco::Candidate
Looking for module label: goodJet
Looking for productInstanceName:
cms::Exception going through module PATCandViewCountFilter/step6a run: 1 lumi: 2 event: 1369557
If you wish to continue processing events after a ProductNotFound exception,
add "SkipEvent = cms.untracked.vstring('ProductNotFound')" to the "options" PSet in the configura
---- ProductNotFound END
Exception going through path looseEventSelection
---- ScheduleExecutionFailure END
an exception occurred during current event processing
cms::Exception caught in CMS.EventProcessor and rethrown
---- EventProcessorFailure END
```

**Note:** For this error we introduced a typo in line 64 to the [analyzeTopSelection\\_cfg.py](#) configuration file of the more complex analysis example in the *PatExamples* package. We introduced a collection which is not part of the event content and has not been produced before.

- **What's going on?**

One of the modules that is scheduled in the process path of your configuration file is looking for a object collection which is not part of the event content.

- **What happened?**

Either the object type, module label or instance label of the collection your module is trying to load is not correct or you forgot to produce the corresponding collection beforehand.

- **What is the remedy?**

Make sure that the collection your module is trying to load really exists.

There is two ways to check the event content of the file you are processing. With the *edmDumpEventContent* tool you will get an overview of all persistent objects, which are part of the event content of the input file you are using. We give an example for a special PAT tuple below:

## WorkbookTroubleshooting < CMSPublic < TWiki

```
[rwolf@tcx061]~/scratch/CMSSW_3_8_2/src% edmDumpEventContent top/patTuple_top_madgraph.root_ttbar
edm::TriggerResults      "TriggerResults"      ""      "HLT8E29."
edm::TriggerResults      "TriggerResults"      ""      "HLT."
edm::TriggerResults      "TriggerResults"      ""      "REDIGI."
edm::TriggerResults      "TriggerResults"      ""      "RECO."
vector                   "offlinePrimaryVertices"  ""      "RECO."
edm::OwnVector >        "selectedPatJets"      "tagInfos"  "PAT."
edm::TriggerResults      "TriggerResults"      ""      "PAT."
vector                   "selectedPatJets"      "caloTowers"  "PAT."
vector                   "selectedPatElectrons"  ""      "PAT."
vector                   "selectedPatJets"      ""      "PAT."
vector                   "patMETs"              ""      "PAT."
vector                   "selectedPatMuons"      ""      "PAT."
vector                   "selectedPatJets"      "genJets"    "PAT."
vector                   "selectedPatJets"      "pfCandidates"  "PAT."
```

Alternatively you can define the *EventContentAnalyzer* module and place it at an arbitrary position in front of the module which causes you trouble:

```
## check the event content process.content = cms.EDAnalyzer("EventContentAnalyzer")

process.p = cms.Path( ... process.content * ... )
```

### Error: InvalidReference

```
%MSG-s CMSEException: AfterFile 02-Jun-2010 14:31:43 CEST PostEndRun
cms::Exception caught in cmsRun
---- EventProcessorFailure BEGIN
EventProcessingStopped
---- ScheduleExecutionFailure BEGIN
ProcessingStopped
---- InvalidReference BEGIN
BadRefCore Attempt to dereference a RefCore containing an invalid
ProductID has been detected. Please modify the calling
code to test validity before dereferencing.
cms::Exception going through module PatMCMatching/analyzePatMCMatching run: 1 lumi: 666672 event:
---- InvalidReference END
Exception going through path p
---- ScheduleExecutionFailure END
an exception occurred during current event processing
cms::Exception caught in CMS.EventProcessor and rethrown
---- EventProcessorFailure END
```

**⚠ Note:** In CMSSW composed (or high analysis) objects very often internally refer to the lower level analysis objects they are composed of via pointer relations. A typical example is a *reco::Muon* pointing to the inner track it was created from. The above problem occurs if you try to dereference such a object pointer to a lower level analysis object which is not part of the event content any more. In our example the *reco::Muon* would still point to an inner track which has been dropped from the event content at some earlier processing step.

- **What's going on?**

There is an object in your analysis code that is referring to another object which is not in the event content any more.

- **What happened?**

You try to read out some information via a pointer to a lower analysis object. You did not check whether this pointer still refers to a valid object or not.

- **What is the remedy?**

Make sure that each pointer you are calling refers to a valid object via a NULL pointer request. The information you wanted to address does not exist anymore on the input file that you are using.

In our example add the following line to your analysis code:

```
if(!muon.innerTrack().isNull()){ //now go on if the innerTrack pointer really corresponds to a va
```

Unfortunately the information you wanted to have is most probably lost from the input file you are using. Due to the 'hidden' reference structure of the high analysis objects it is non trivial to reduce the file size of an AOD/RECO input file without losing analysis relevant information. The Physics Analysis Toolkit (PAT) tries to assist you on this quest. Have a look at SWGuidePATEmbeddingExercise to learn more about this.

## Useful tools for debugging your code

Useful tools for debugging your code are documented here

## Review status

[Show](#) [Hide](#)

Reviewer/Editor and Date (copy from screen)	Comments
JohnStupak - 16-September-2013	Review
RogerWolf - 15 Sept 2010	Complete overhaul

Responsible: SudhirMalik and MargueriteTonjes

This topic: CMSPublic > WorkBookTroubleShooting

Topic revision: r57 - 2018-02-27 - SudhirMalik



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.  
or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback