

ROOT – Installation

The image shows a browser window displaying the ROOT website. The browser's address bar shows the URL `https://root.cern`. The website's navigation menu includes links for 'About', 'Install', 'Get Started', 'Forum & Help', 'Manual', 'Blog Posts', 'Contribute', and 'For Developers'. The main content area features the heading 'ROOT: analyzing petabytes of data, scientifically.' and a sub-heading 'An open-source data analysis framework used by high energy physics and others.' Below this, there are two buttons: 'Learn more' and 'Install v6.22/02'. The 'Install v6.22/02' button is circled in red. Below the buttons are four icons representing 'Get Started', 'Reference', 'Forum & Help', and 'Gallery'. At the bottom, there are three columns of text describing ROOT's capabilities: statistical analysis, high-performance software, and integration with C++ and Python.

File Edit View History Bookmarks Tools Help

Slide 1 - Verzeigu x Slide 1 - c++_einle x Slide 1 - LinuxCor x Slide 1 - Pointer.p x Slide 1 - FileIO.pd x Slide 1 - Funktion x ROOT: analyzing | x +

https://root.cern 90% Search

CERN PIOrganization C++ / ROOT LHCb Meetings Dies/Das VMware openSUSE DE Computing Biking Bookmarks Menu Erste Schritte Meistbesucht

ROOT
Data Analysis Framework

About **Install** Get Started Forum & Help Manual Blog Posts Contribute For Developers

ROOT: analyzing petabytes of data, scientifically.

An open-source data analysis framework used by high energy physics and others.

Learn more **Install v6.22/02**

Get Started Reference Forum & Help Gallery

v-1

ROOT enables *statistically sound* scientific analyses and visualization of large amounts of data: today, more than 1 exabyte (1,000,000,000 gigabyte) are

globe

As *high-performance* software, ROOT is written mainly in C++. You can use it on Linux, macOS, or Windows; it works out of the box. ROOT is open

\$ _

ROOT comes with an incredible C++ interpreter, ideal for *fast prototyping*. Don't like C++? ROOT integrates super-smoothly with Python thanks to its

ROOT – Installation

Es gibt einen Produktionsrelease von ROOT. Dafür steht der gesamte Quellcode und binäre Bibliotheken für bestimmte Compilerversionen und Linux Distributionen zur Verfügung.

Useful links
All releases

Release 6.22/02 - 2020-08-17

Release Notes

The release notes for this release can be found here.

Source distribution

Platform	Files	Size
source	root_v6.22.02.source.tar.gz	165M

Binary distributions

Platform	Files	Size
CentOS 7	root_v6.22.02.Linux-centos7-x86_64-gcc4.8.tar.gz	186M
Fedora 30	root_v6.22.02.Linux-fedora30-x86_64-gcc9.3.tar.gz	225M
Fedora 31	root_v6.22.02.Linux-fedora31-x86_64-gcc9.3.tar.gz	225M
Fedora 32	root_v6.22.02.Linux-fedora32-x86_64-gcc10.2.tar.gz	227M
Ubuntu 16	root_v6.22.02.Linux-ubuntu16-x86_64-gcc5.4.tar.gz	200M
Ubuntu 18	root_v6.22.02.Linux-ubuntu18-x86_64-gcc7.5.tar.gz	218M
Ubuntu 19	root_v6.22.02.Linux-ubuntu19-x86_64-gcc9.2.tar.gz	223M
Ubuntu 20	root_v6.22.02.Linux-ubuntu20-x86_64-gcc9.3.tar.gz	224M
macOS 10.13 Xcode 10	root_v6.22.02.macosx64-10.13-clang100.pkg	315M
macOS 10.13 Xcode 10	root_v6.22.02.macosx64-10.13-clang100.tar.gz	200M
macOS 10.14 Xcode 10	root_v6.22.02.macosx64-10.14-clang100.pkg	314M
macOS 10.14 Xcode 10	root_v6.22.02.macosx64-10.14-clang100.tar.gz	200M

Zum Kompilieren und Installieren von ROOT verwenden

Benutzbar falls Sie die gleiche Linux Distribution installiert haben

ROOT – Binäre Installation

Die ROOT Dateien werden als gepacktes und komprimiertes File verteilt, als tar Archiv. Beim Entpacken wird die directory Struktur zum Zeitpunkt des Packens erhalten.

- **tar Beispiel**

alle Directories und Files in `myDirectory` werden in ein File mit dem Namen `myTarFile.tar` geschrieben und anschließend mit `gzip` komprimiert.

```
$> tar -cvf myTarFile.tar myDirectory
$> gzip myTarFile.tar
```

Das komprimierte tar File mit dem Namen `myTarFile.tar.gz` wird entpackt und in die ursprüngliche Struktur expandiert.

```
$> tar -zxvf myTarFile.tar.gz
```

Beim Entpacken der binären ROOT Distribution wird die Struktur von ROOT erhalten.

```
$> cd MeinROOT
$> tar -zxvf root_v6.22.02.Linux-centos7-x86_64-gcc4.8.tar.gz
$> . MeinROOT/root/bin/thisroot.sh
$> root
```

ROOT – Quellcode Installation

Zum Kompilieren von ROOT benötigen wir die Header der Programmpakete, die ROOT benötigt. Diese sind in PaketName-devel oder PaketName-dev zu finden. Die Liste der benötigten Pakete für die verschiedenen Linux Distributionen ist in den Installationsinstruktionen zu finden.

<https://root.cern/install/dependencies/>

- **Quellcode Installation**

die notwendigen Pakete können unter OpenSuSE 15 entweder mit `yast2` oder `zypper` installiert werden:

```
$> zypper install git bash cmake gcc-c++ gcc binutils \  
$> xorg-x11-libX11-ccache libXpm-devel xorg-x11-devel \  
$> xorg-x11-util-devel libXext-devel libX11-devel
```

optionale Pakete:

```
$> zypper install gcc-fortran libopenssl-devel \  
$> pcre-devel Mesa Mesa-devel glew-devel pkg-config \  
$> libmysqld-devel fftw3-devel fftw3-threads-devel \  
$> fftw3-mpi-devel libcfitsio6 graphviz-devel \  
$> libdns169_sd avahi-compat-mDNSResponder-devel \  
$> openldap2-devel patterns-devel-python-devel_python3 \  
$> libxml2-devel krb5-devel gsl-devel libqt4-devel libpythia8 \  
$> pythia-devel libjpeg62-devel libtiff-devel giflib-devel
```

die Verwendung der python Version hängt von der installierten Version ab.

ROOT – Quellcode Installation

- ROOT soll im directory `/local/cern` installiert werden

Bitte zum superuser root wechseln und das komprimierte tar File mit dem Namen der letzten Version **hier** herunterladen `root_v6.26.06.source.tar.gz` entpacken und in die ROOT Quellstruktur expandieren.

```
$> tar -zxvf root_v6.26.06.source.tar.gz
$> less root-6.26.06/README/INSTALL
```

Im folgenden nehmen wir eine Installation für alle user des Linuxsystems vor. Die Dateien werden in `/local/cern/root-6.26.06` zu finden sein.

```
$> mkdir root-6.26.06_build; cd root-6.26.06_build
$> cmake /local/cern/root-6.26.06 -Droofit=ON -Dminuit2=ON
$> make -j4
$> ln -s root_v6.26.06_build root
$> source /local/cern/root/bin/thisroot.sh
```

Optionen hängen von den installierten Software-Headern ab

erzeuge link zum dir root

Setzt die Pfad Variablen, die zur Benutzung von ROOT notwendig sind
Alternativ kann folgendes in das `.bashrc` geschrieben werden

```
export ROOTSYS=/cern/root/
export PYTHONDIR=/usr/bin/python3
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$PYTHONDIR/lib
export PYTHONPATH=$ROOTSYS/lib:$PYTHONPATH
```