

Kosmische Strahlung macht Schule

Achim Stoeßl, Basho Kaminsky, Robert Franke
DESY, Standort Zeuthen

Outline

Physikalischer Hintergrund

Kosmische Strahlung und Schulphysik

Projekte

Tabletop-Experimente

Experimente mit Webinterface

Langzeitexperimente an entfernten Orten

Software

Erfahrungen mit SchülerInnen

Ausblick

Wer sind wir?

Masterstudent (Basho) & Doktoranden (Achim & Robert)
am

DESY

- Beschleuniger
- Forschung mit Photonen
- Teilchen- & Astroteilchenphysik
 - IceCube - weltgrößtes Neutrino-teleskop in der Antarktis



Was machen wir?

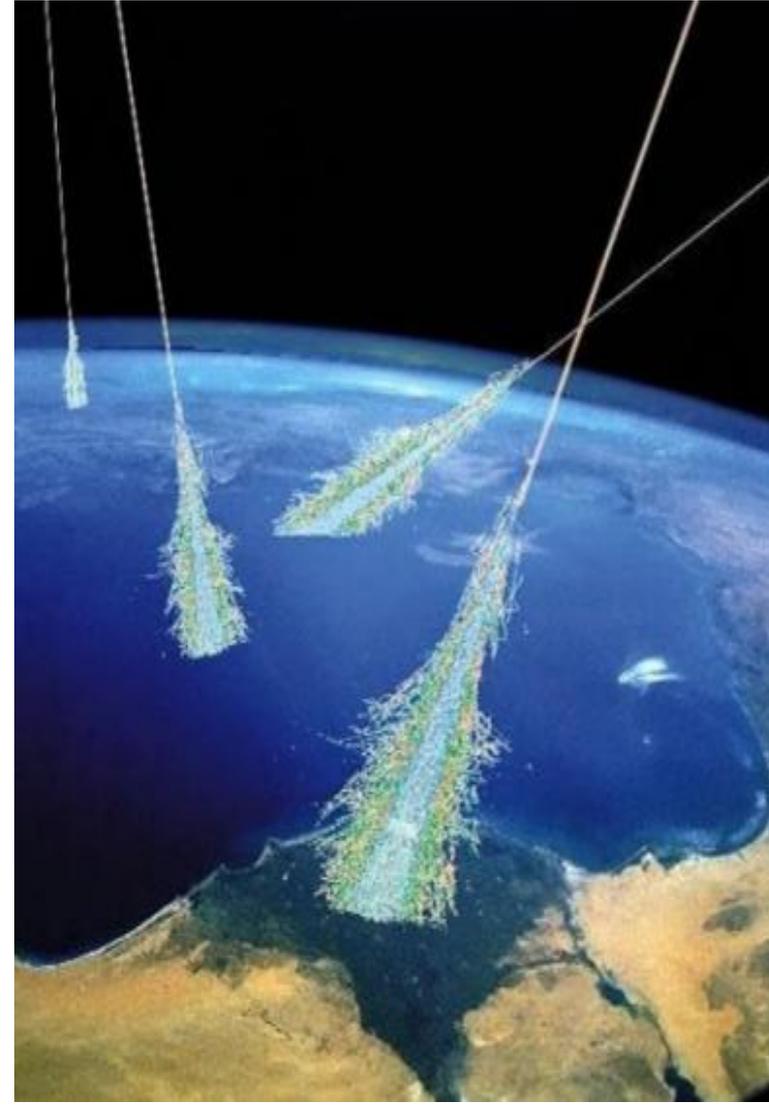
Experimente, die Schüler an moderner Grundlagenforschung teilhaben lassen.

- Physik
- Elektronik
- Detektoren
- Software
- Datenanalyse

Physikalischer Hintergrund

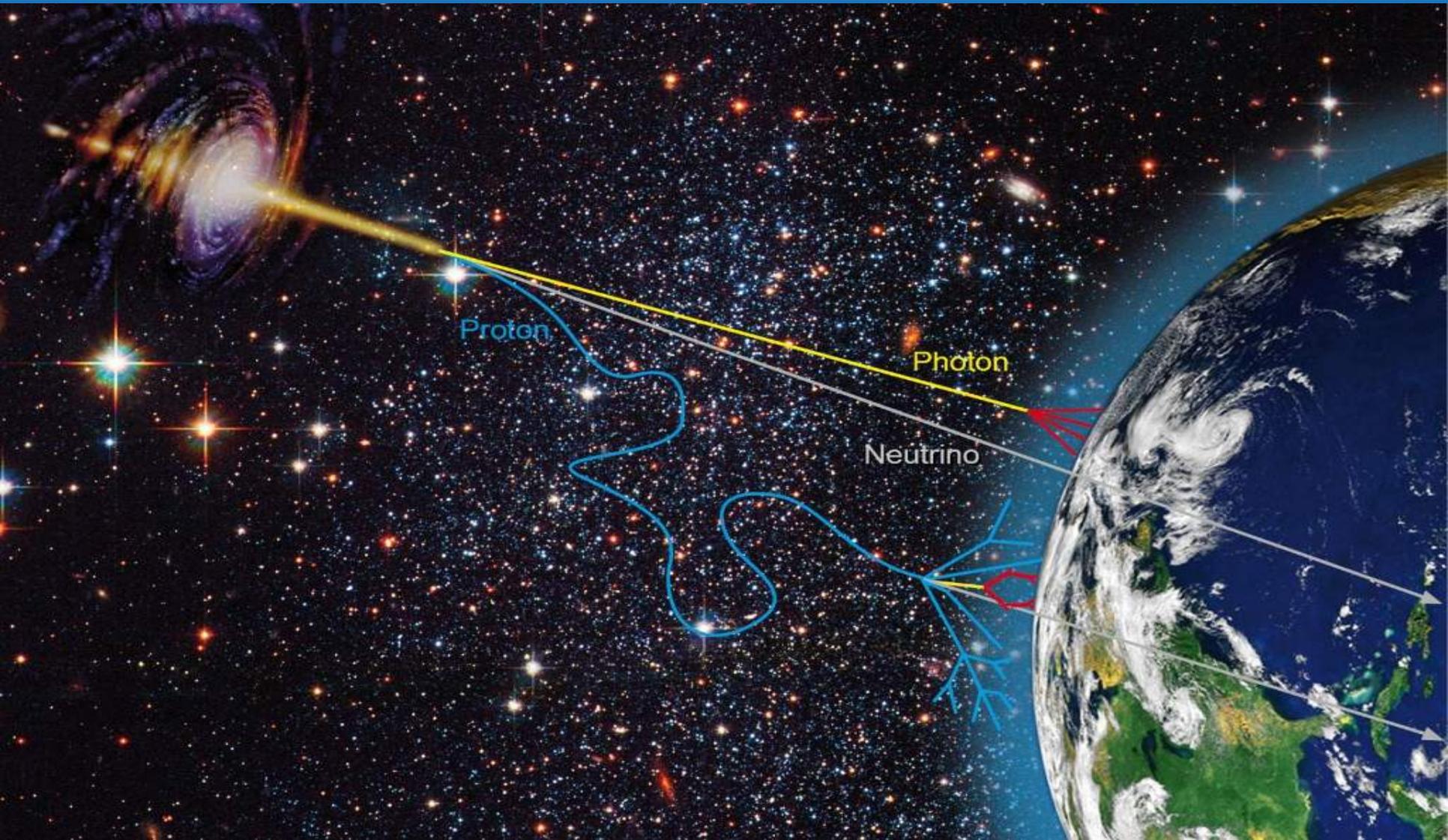
Was ist kosmische Strahlung?

- Teilchen-"Strahlung":
 - Protonen, Elektronen, Atomkerne, Neutrinos, Photonen, ...
- 1912 von Victor Hess entdeckt
- Wo kommt sie her???
- U. a. Teilchen mit den höchsten je gemessenen Energien



Physikalischer Hintergrund

Woher kommt kosmische Strahlung?

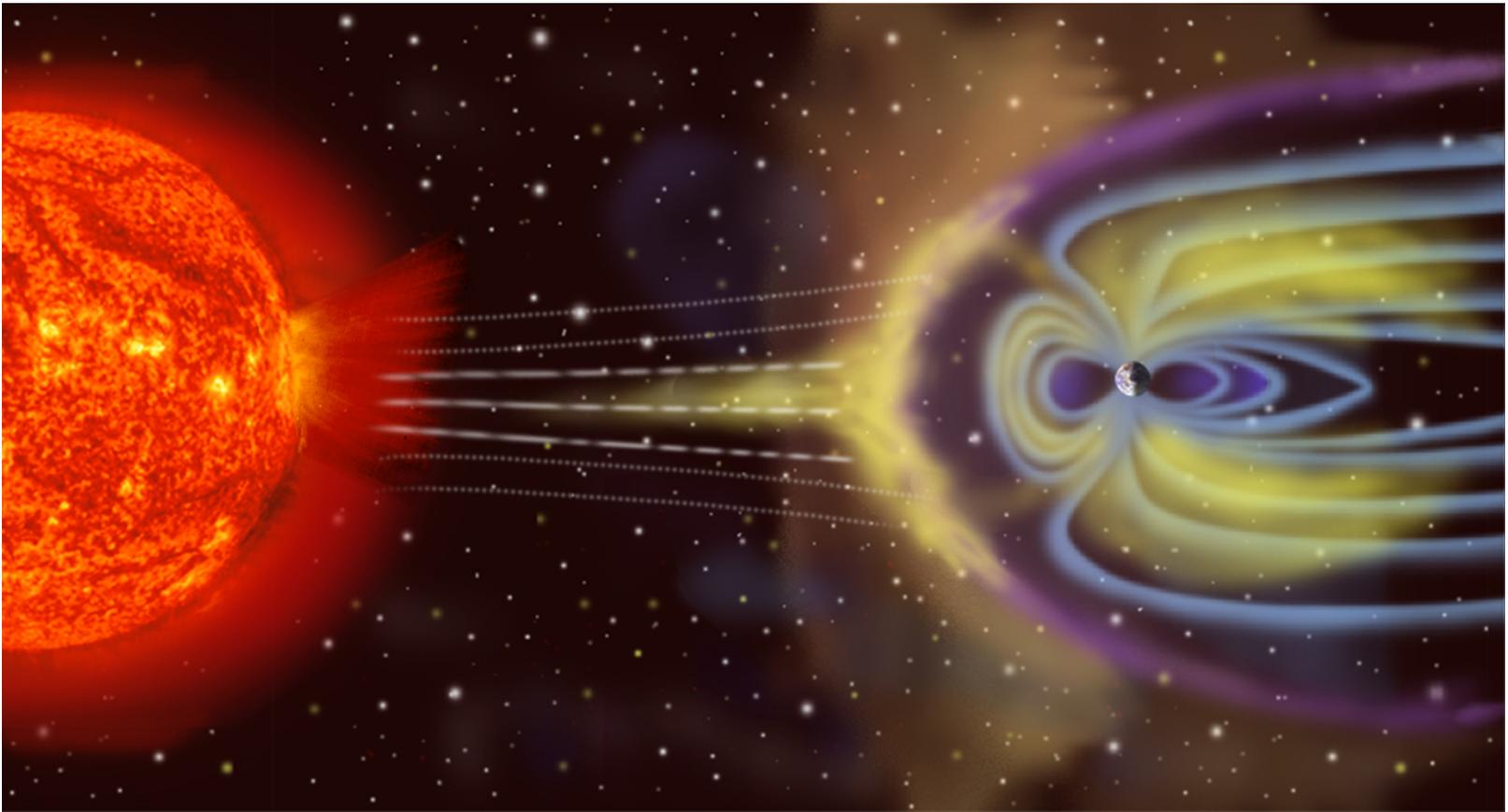


Physikalischer Hintergrund

Woher kommt kosmische Strahlung?

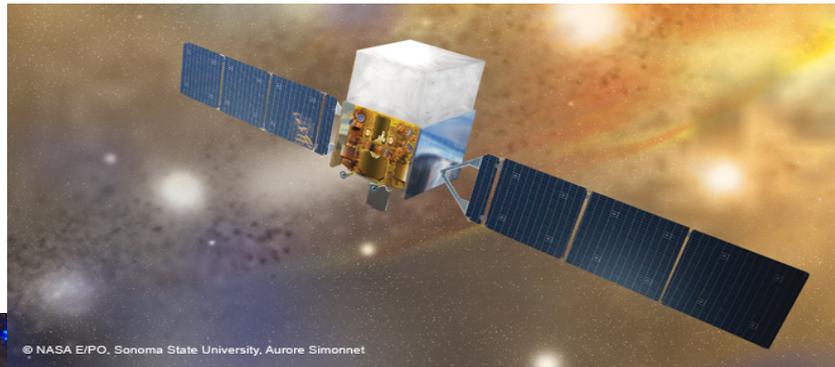
Sonnenwind:

Quelle niederenergetischer kosmischer Strahlung



Physikalischer Hintergrund *Aktuelle Experimente.*

FERMI (Weltraum)



HESS, Namibia



Veritas, Arizona



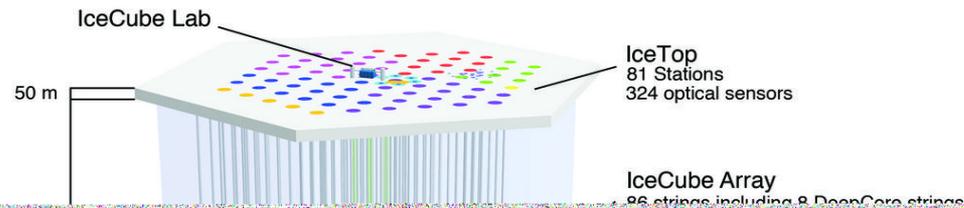
© NASA E/PO, Sonoma State University, Aurore Simonnet

© F. Acero & H. Gast

Physikalischer Hintergrund

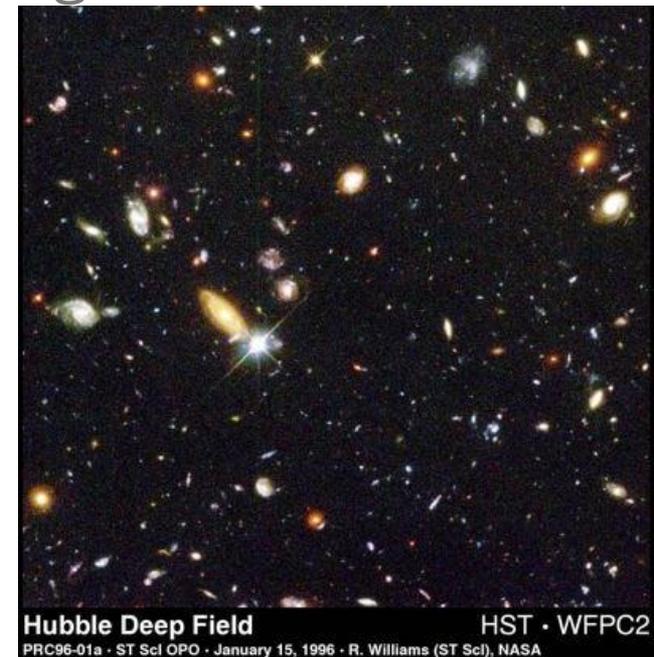
Aktuelle Experimente.

IceCube, Südpol



Physikalischer Hintergrund *Schulphysik?*

- Astroteilchenphysik: Schnittstelle zwischen Teilchenphysik, Astronomie, Kosmologie und Informatik
- Schul-Experimente möglich?



Physikalischer Hintergrund

Was kann erlernt werden?

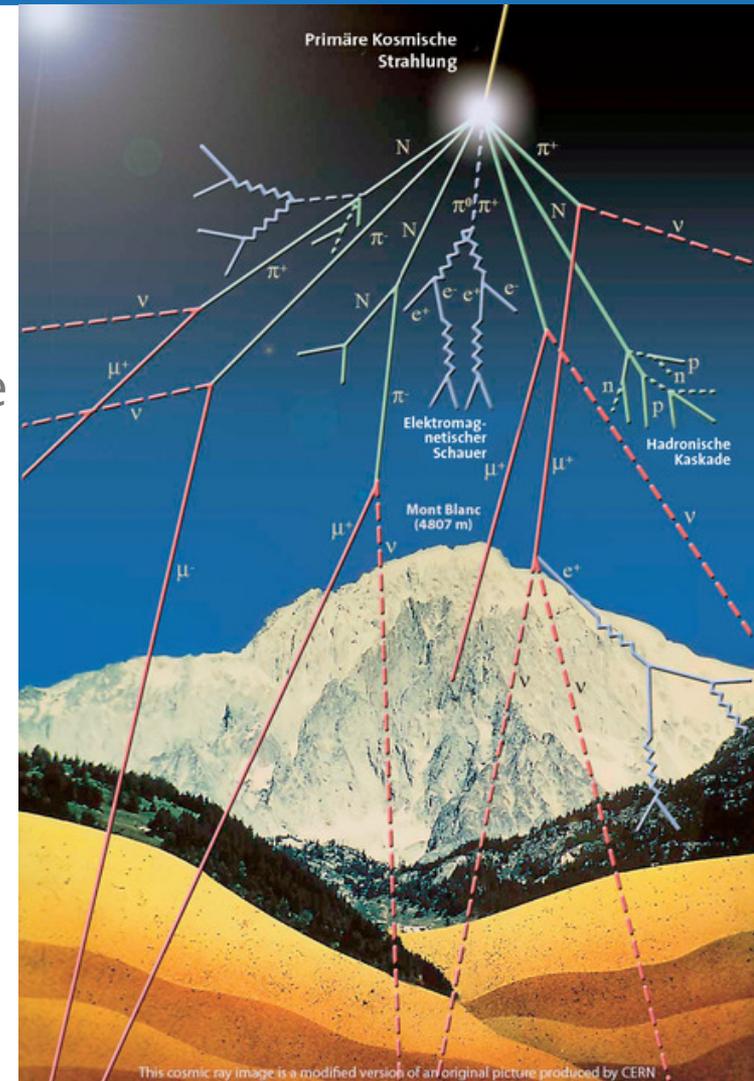
- Einblick in Detektortechnik (Koinzidenzmessung: Nobelpreis 1954 für Bothe)
- mathematische Gesetzmässigkeiten in der Natur (Zerfallsgesetz, Statistik...)
- Kennenlernen von moderner Datenverarbeitung



Physikalischer Hintergrund

Was lässt sich messen?

- Myonen entstehen bei Kollision kosmischer Strahlung mit Atomen in der Erdatmosphäre
- negative Ladung, Masse = $206 m_e$
- Fluss an der Erdoberfläche: 100 pro m^2 und s
- Myonen sind instabil und zerfallen



Projekte

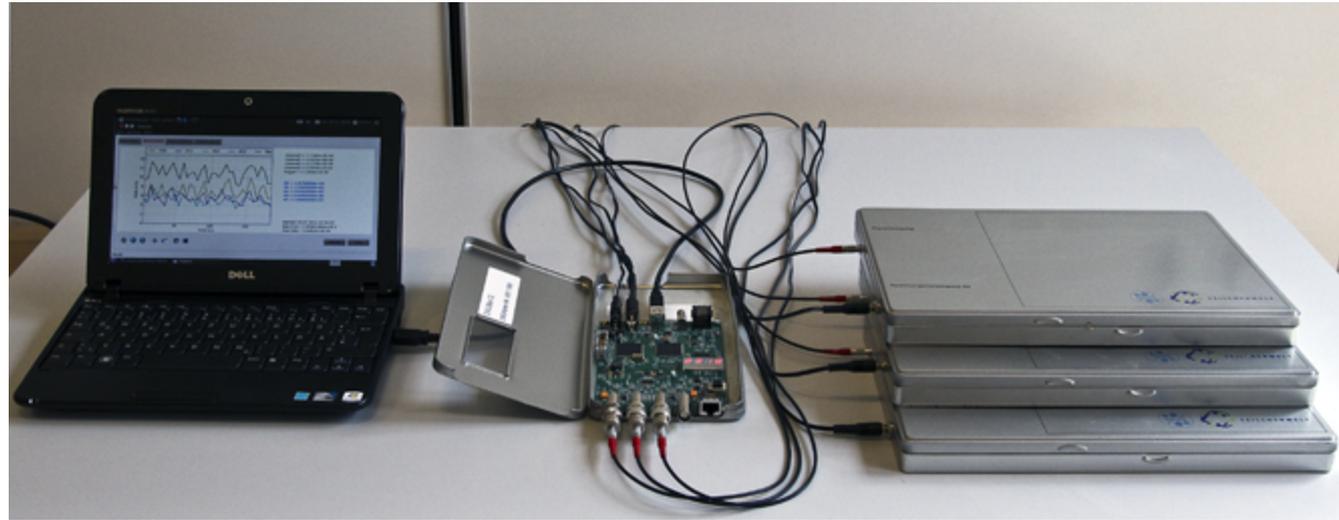
Kamiokannen



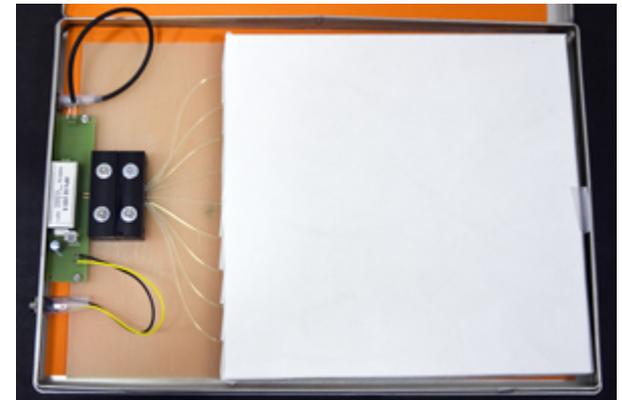
- Entwicklung aus Göttingen
- Normale Thermoskanne mit Photomultiplier

Projekte

Szintilationszähler



- DAQ entwickelt am Fermilab, quasi offen
- International genutzt
- CPLD + Mikrocontroller
- Zeitpräzision ca. 1,25 ns relativ, 10 ns absolut via GPS!



Software

Muonic

muonic - für Schüler leicht zu bedienende Software zur Datennahme und Konfiguration der Experimente

- graphische Oberfläche (PyQt4)
- Speichern der Daten in verschiedenen Formaten
- graphische Aufbereitung der Messungen
- mit Szintillationszählern an Schulen

Projekte

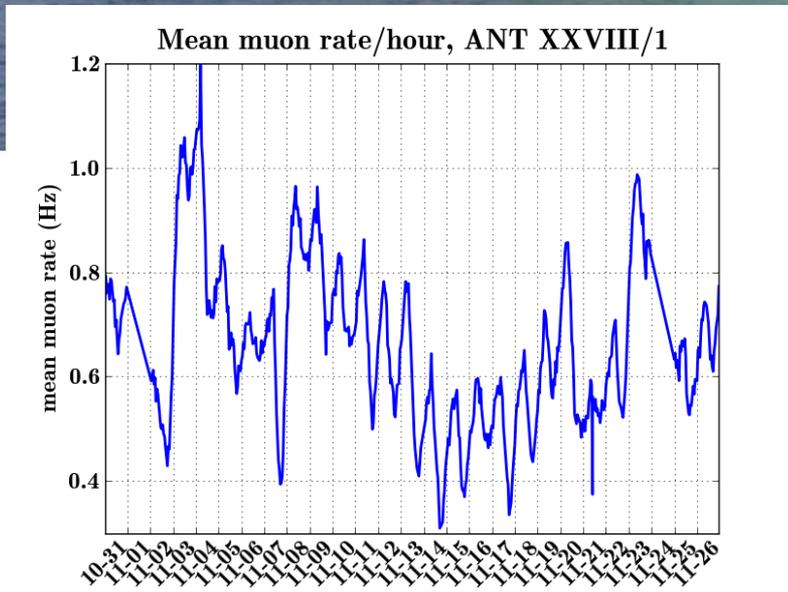
Trigger-Hodoskop

- <http://schlab.ifh.de/trigger/>
- Apache, Perl, Javascript, C++ (ROOT)
- Läuft (mit kleinen Unterbrechungen) seit 2004



Langzeitprojekte

Polarstern & Neumayer Projekt



Langzeitprojekte

Polarstern & Neumayer Projekt

- Myonen-Ratenmessung in Abhängigkeit des Erdmagnetfeldes & besonderen Wetterbedingungen
- Suche nach besonderen Ereignissen (z.B. Sonneneruptionen)
- Daten für interessierte Schüler - z.B. Jugend forscht



Software *icebreaker*

icebreaker - robuste, vollautomatische Software zur Datennahme der Experimente ohne direkten Zugang

- vollautomatische, kontinuierliche Datennahme
- automatisches Versenden von Daten und/oder Statusmails möglich
- erweiterbar mit zusätzlichen Komponenten (Barometer, Neigungssensor, etc.)

Software

Anforderungen und Ziele

Generelle Anforderungen

- leicht wartbarer und gut lesbarer Code
- anpassbar an neue Entwicklungen
- niedrige Kosten
- breite Einsetzbarkeit
- lauffähig auch auf alter oder niedrig performanter Hardware

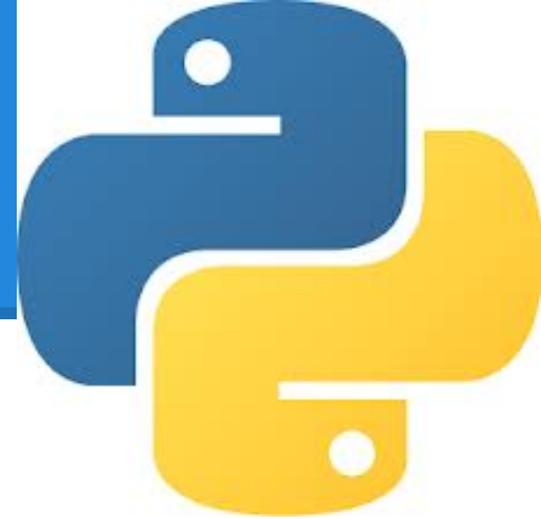
—————> erfüllt durch ***muonic & icebreaker***

Software *Umgebungen*

- Linux (Ubuntu 10.04 LTS)
 - "schwache" Hardware (Dell Latitude 2100/Inspiron Mini 1018 Netbooks)
 - (Lizenz-)Kosten
 - Erfahrung
 - Stabilität
 - Fernwartbarkeit über langsame Verbindungen
- *Python, C*



Warum Python?



- Erfahrung aus IceCube
- Performant genug
- prinzipiell gute Cross-Plattform GUI
- Bibliotheken (Pyserial, matplotlib, Cython)
- Wartbarkeit
- einfache Integration von C/C++ Bibliotheken
- Einsteigerliteratur für Schüler verfügbar

Software *Bibliotheken*

Eingesetzte Pakete:

pyqt4 - Graphische Oberfläche, Cross Platform

pyserial - Kommunikation mit serieller Schnittstelle

matplotlib - Histogramme, Liniendiagramme

numpy - Performante Arrays

scipy - Fits von Funktionen an Daten

virtualenv - isolierte, portierbare Umgebung

Softwarebeispiel

Muonic

Architektur

- 2 Threads sorgen für DAQ I/O
- QTimer arbeitet I/O alle 200 ms ab
- Timer updated Ratenplot alle 5s
- Ratenplot mit *matplotlib*
- Benutzerinterface mit verschiedenen Widgets: *PyQt4*

Kommunikation mit DAQ-Karte

Einfaches Textprotokoll über serielle Schnittstelle (via USB)

7CAB0C26	00	00	00	00	2B	00	00	00	7C66EC3C	040013.008	020811	A	09	0	+0058	^M
7CAB0C26	00	37	00	00	00	31	00	00	7C66EC3C	040013.008	020811	A	09	0	+0058	^M
7EE5E1A9	BF	00	00	00	00	00	00	00	7DE4647C	040014.016	020811	A	09	0	+0066	^M
7EE5E1AA	00	00	25	00	22	00	00	00	7DE4647C	040014.016	020811	A	09	0	+0066	^M
7EE5E1AA	00	00	00	2D	00	00	00	00	7DE4647C	040014.016	020811	A	09	0	+0066	^M
7EE5E1AA	00	34	00	00	00	34	00	00	7DE4647C	040014.016	020811	A	09	0	+0066	^M
803D6DFA	80	00	00	00	33	00	00	00	7F61DCBC	040015.008	020811	A	09	0	+0058	^M
803D6DFA	00	00	38	00	00	00	00	00	7F61DCBC	040015.008	020811	A	09	0	+0058	^M
803D6DFB	00	00	00	21	00	00	00	00	7F61DCBC	040015.008	020811	A	09	0	+0058	^M
803D6DFB	00	00	00	00	00	2A	00	00	7F61DCBC	040015.008	020811	A	09	0	+0058	^M
808BBA7F	AB	00	2F	00	00	00	00	00	7F61DCBC	040015.008	020811	A	09	0	+0058	^M
808BBA7F	00	00	00	3E	00	00	00	00	7F61DCBC	040015.008	020811	A	09	0	+0058	^M
808BBA80	00	23	00	00	00	00	00	00	7F61DCBC	040015.008	020811	A	09	0	+0058	^M
81934993	B4	00	00	00	00	00	00	00	80DF54FC	040016.016	020811	A	09	0	+0066	^M
81934993	00	00	3B	00	00	00	00	00	80DF54FC	040016.016	020811	A	09	0	+0066	^M
81934994	00	00	00	21	00	00	00	00	80DF54FC	040016.016	020811	A	09	0	+0066	^M

Software

Muonic - Codebeispiel I - pyserial

```
def get_port():
    connected = False
    while not connected:
        which_tty_daq = os.path.split(os.path.abspath(__file__))[0] + os.sep + "which_tty_daq"
        dev = subprocess.Popen([which_tty_daq], stdout=subprocess.PIPE).communicate([0])
        dev = "/dev/" + dev
        dev = dev.rstrip('\n')
        self.logger.info("Daq connected to %s",dev)
        try:
            port = serial.Serial(port=dev, baudrate=115200,
                                  bytesize=8,parity='N',stopbits=1,
                                  timeout=1,xonxoff=xonxoff)

            connected = True
        except serial.SerialException, e:
            logger.error(e)
            logger.error("Waiting 10 seconds")
            sleep(10)

    self.logger.info("Successfully connected to serial port")
    return port
```

Software

Muonic - Codebeispiel II - pyqt

```
def create_widgets(self):
```

```
    self.tabwidget = TabWidget(self, self.options.timewindow, self.logger)
    self.setCentralWidget(self.tabwidget)
```

```
    # provide buttons to exit the application
```

```
    exit = QtGui.QAction(QtGui.QIcon('/usr/share/icons/gnome/24x24/actions/exit.png')
    exit.setShortcut('Ctrl+Q')
    exit.setStatusTip(tr('MainWindow', 'Exit application'))
```

```
    self.connect(exit, QtCore.SIGNAL('triggered()'), self.exit_program)
    self.connect(self, QtCore.SIGNAL('closeEmitApp()'), QtCore.SLOT('close()'))
```

```
    # prepare the config menu
```

```
    config = QtGui.QAction(QtGui.QIcon(''), 'Channel Configuration', self)
    config.setStatusTip(tr('MainWindow', 'Configurer the Coincidences and channels'))
    self.connect(config, QtCore.SIGNAL('triggered()'), self.config_menu)
```

```
    # prepare the threshold menu
```

```
    thresholds = QtGui.QAction(QtGui.QIcon(''), 'Thresholds', self)
    thresholds.setStatusTip(tr('MainWindow', 'Set trigger thresholds'))
    self.connect(thresholds, QtCore.SIGNAL('triggered()'), self.threshold_menu)
```

```
    # the options menu
```

```
    options = QtGui.QAction(QtGui.QIcon(''), 'Options', self)
    options.setStatusTip(tr('MainWindow', 'Set program options'))
    self.connect(options, QtCore.SIGNAL('triggered()'), self.options_menu)
```

Erfahrungen

- Unterschiedliches Vorwissen
- Meist zu wenig Zeit (auch von unserer Seite...)
- Einblick in verwendete Software möglich
 - Python ist einfach(er) zu erlernen
 - Visueller Einstieg sehr hilfreich (z.B. pygame-Tutorial)
- Mehr Dokumentation...



Zusammenfassung / Entwicklungen

- Erfahrung des Phänomens kosmischer Strahlung
- Physik zum "Anfassen"
- Zugang zu aktueller Forschung
- Interdisziplinär

- Mehr Webexperimente (nicht nur Auswertung, auch Steuerung von Experimenten)
- Bessere Didaktik

Weitere Informationen / Links

schlab.ifh.de/trigger

physik-begreifen-zeuthen.desy.de

www.teilchenwelt.de

www.teilchenphysik.de

www.hap-astroparticle.org

astro.desy.de

quarknet.fnal.gov