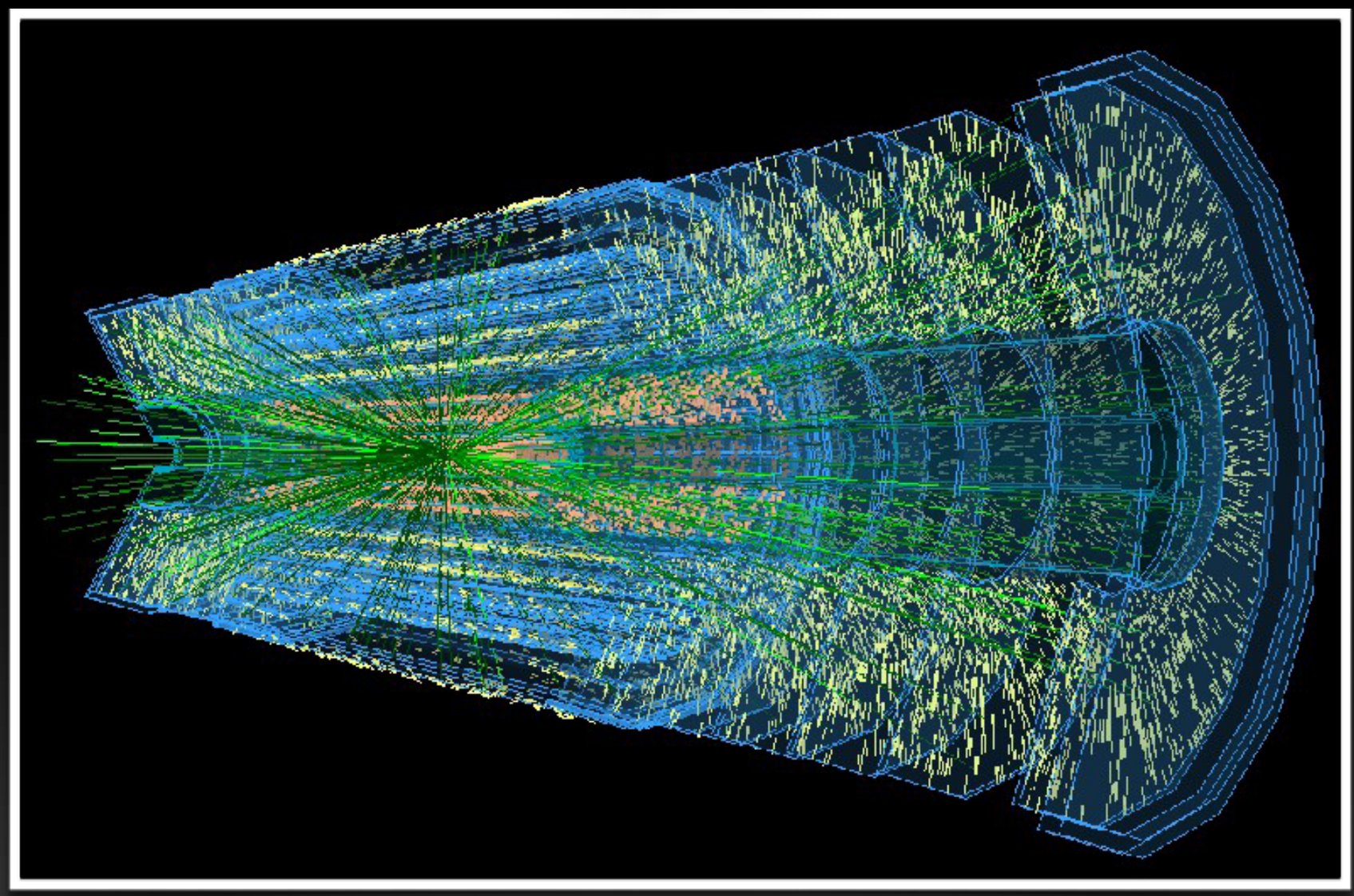
Die zukünftigen Strategien für das Computing in der Hochenergiephysik

Dr. Markus Elsing

Symposium an der Bergischen Universität

27. Juni 2013

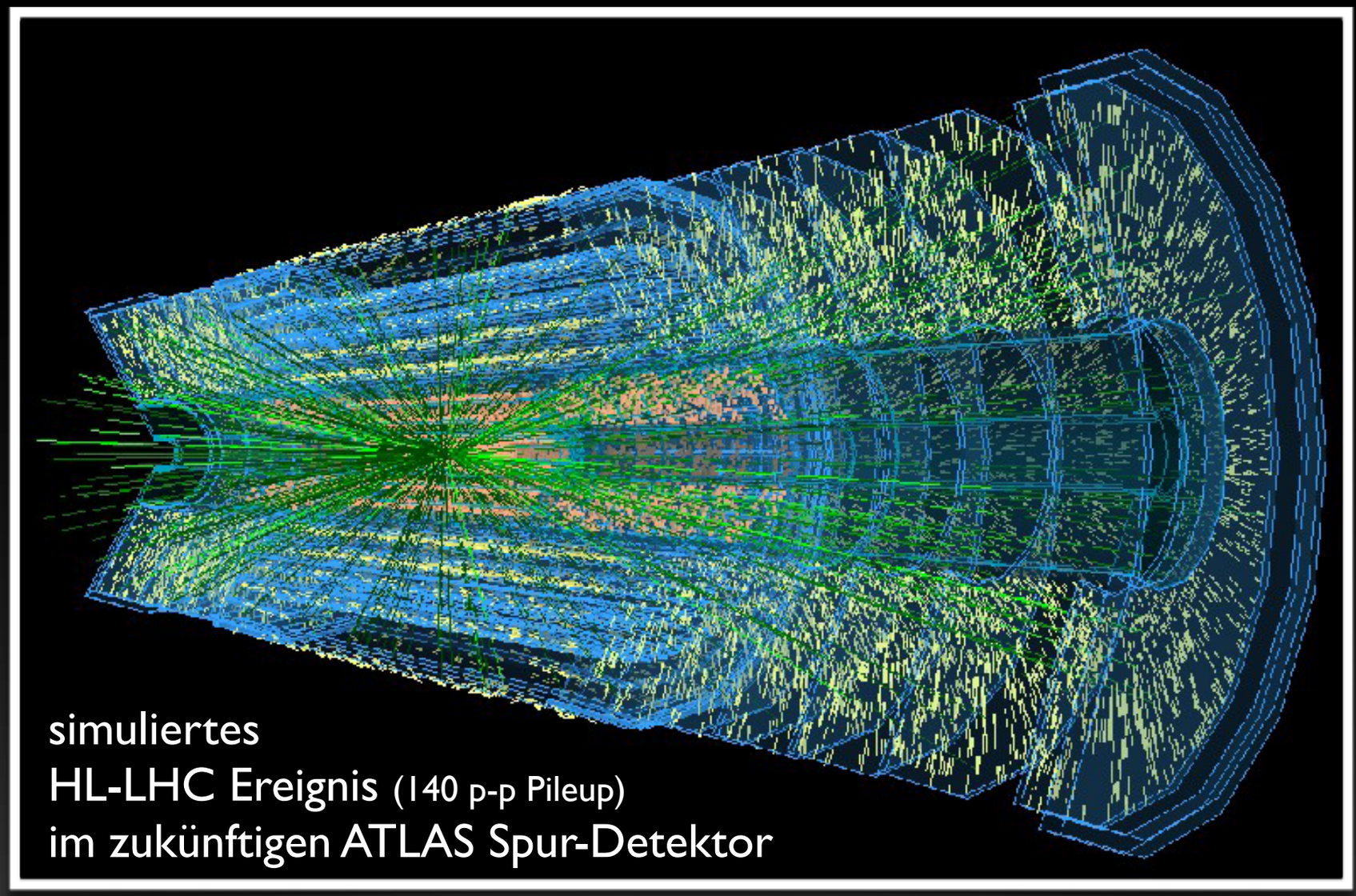


CERN

Einleitung - die Herausforderung

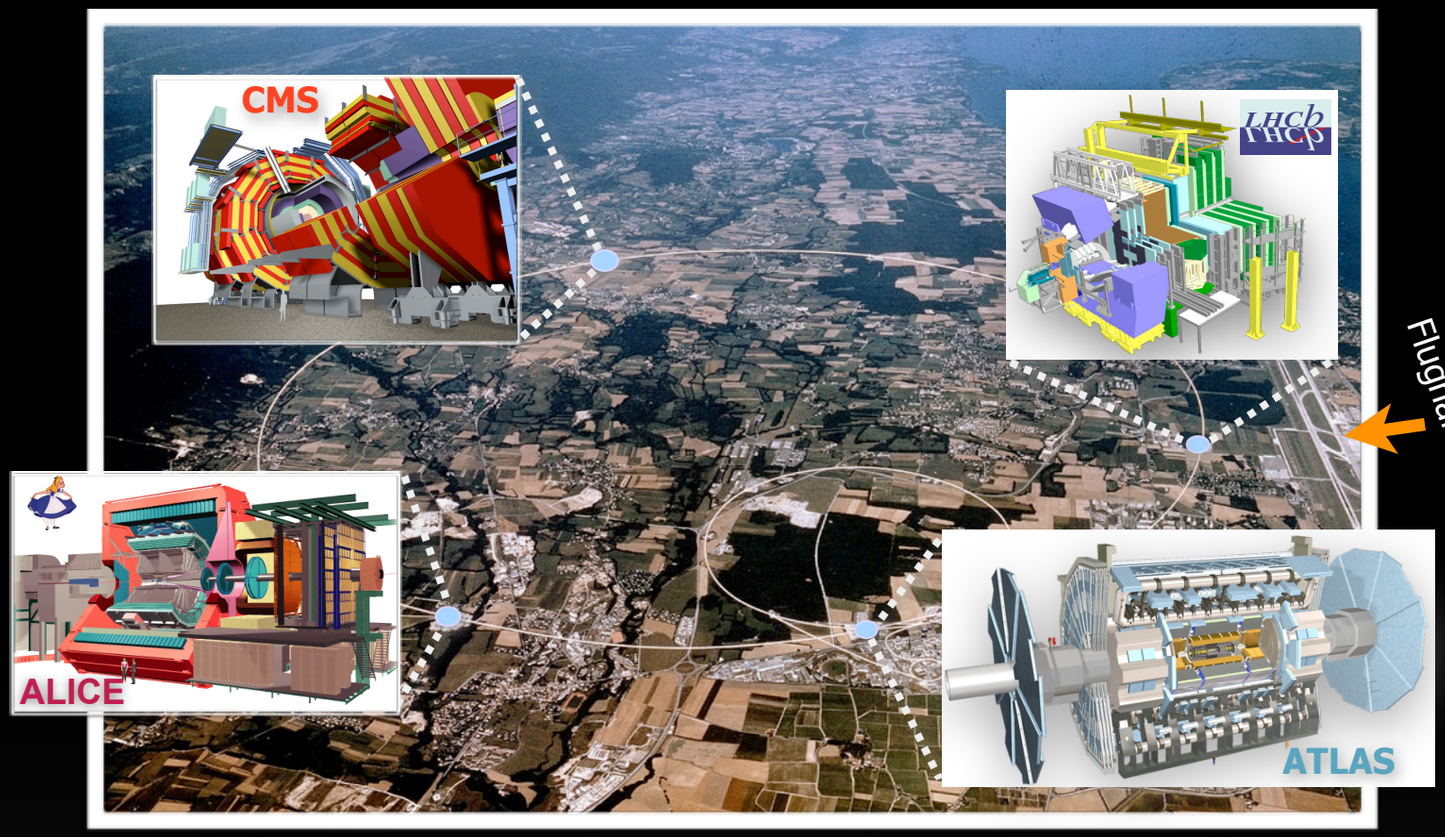
Einleitung - die Herausforderung

- Large Hadron Collider (LHC)
  - ➡ Teilchenbeschleuniger bei höchsten Energien und Intensitäten
  - ➡ 40 Millionen Ereignisse pro Sekunde, jeweils viele überlappende Proton-Proton Wechselwirkungen (Pileup)
  - ➡ auch Kollisionen schwerer Ionen mit tausenden von produzierten Teilchen

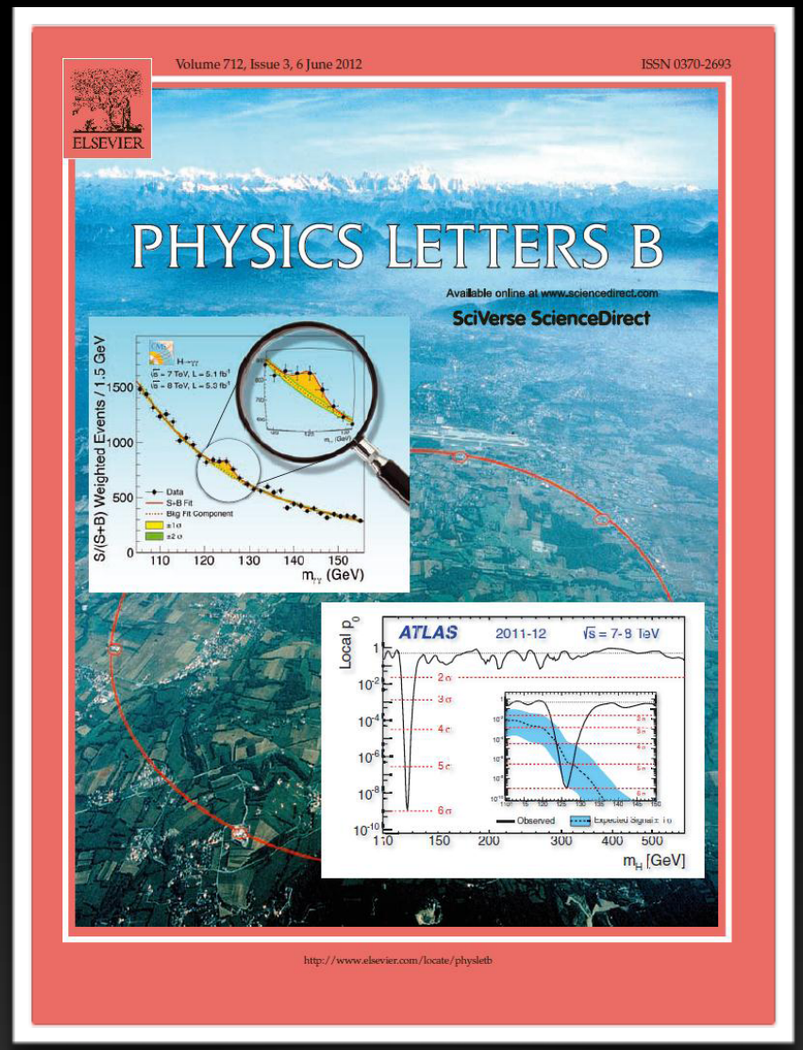


simuliertes
HL-LHC Ereignis (140 p-p Pileup)
im zukünftigen ATLAS Spur-Detektor

Erfahrungen der ersten Datenperiode



- 4 große Experimente
  - ➡ >15 Jahre der Vorbereitung bis zur ersten Datenname
  - ➡ WLCG - GRID Computing

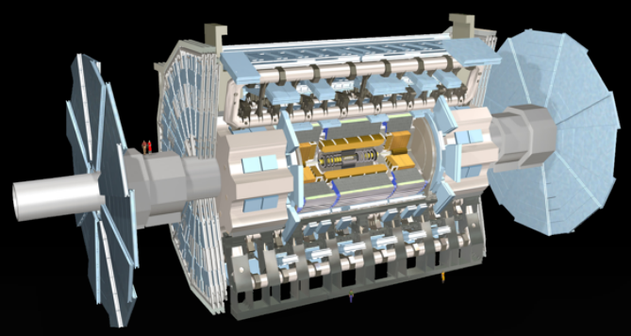


- Datennahme 2010 bis 2012
  - ➡ LHC operierte bei halber Schwerpunktenergie (7 TeV) und bei Luminositäten unterhalb des Designwertes
  - ➡ Beschleuniger, die 4 Experimente und das Computing haben fantastisch funktioniert !
- epochales Ergebnis:
  die Entdeckung des Higgs-Bosons

Das Computing-Modell der Experimente

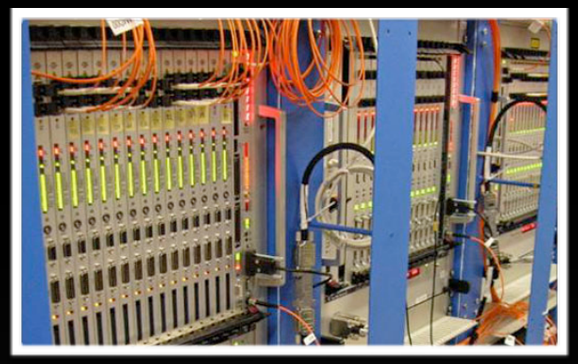
- Ereignis-Selektion bei der Datennahme (Trigger, Online)
  - ➡ Experimente produzieren ca.10 PB Rohdaten jedes Jahr

ATLAS-Detektor



40 MHz, 100 TB/s

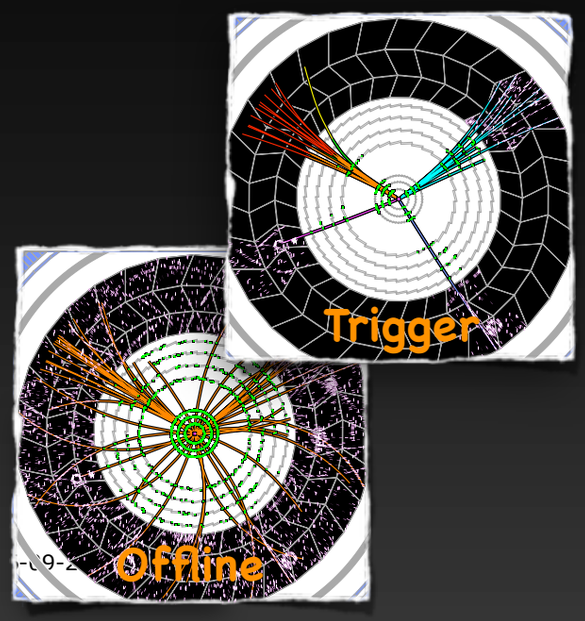
1. Stufe:
spezielle Elektronik



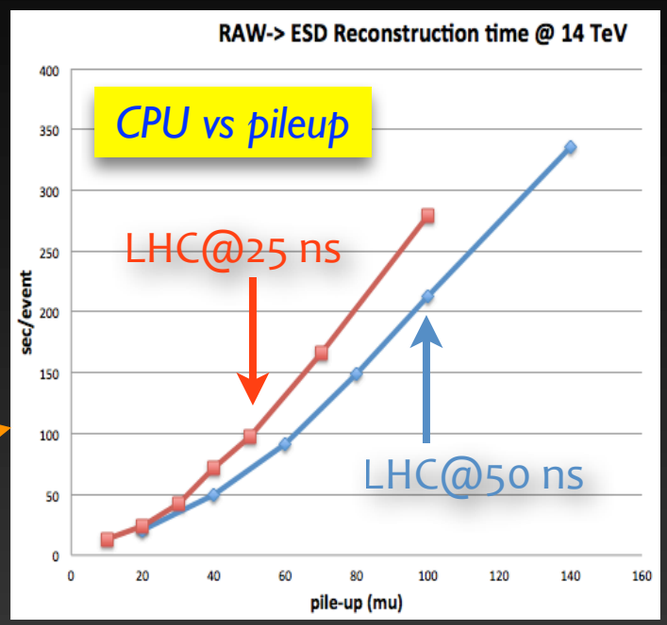
20 kHz, 50 GB/s

2.+3. Stufe:
Software, PC-Farm

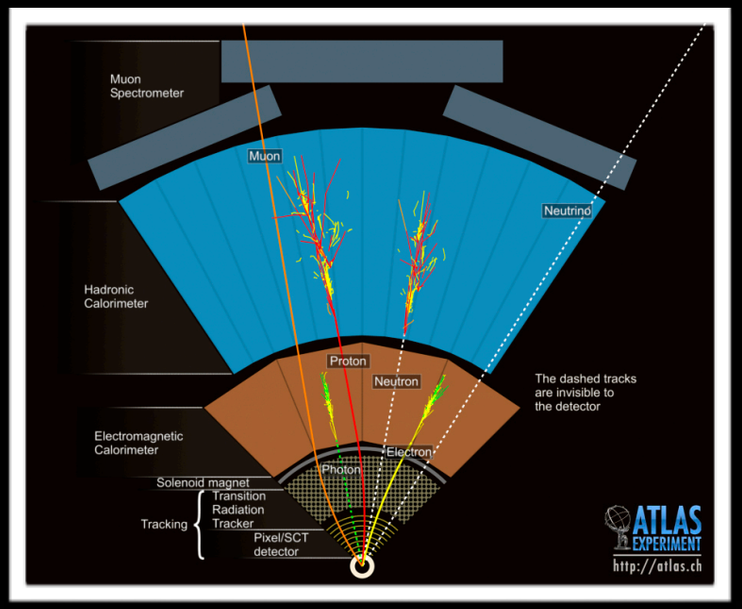
400 Hz , 1 GB/s



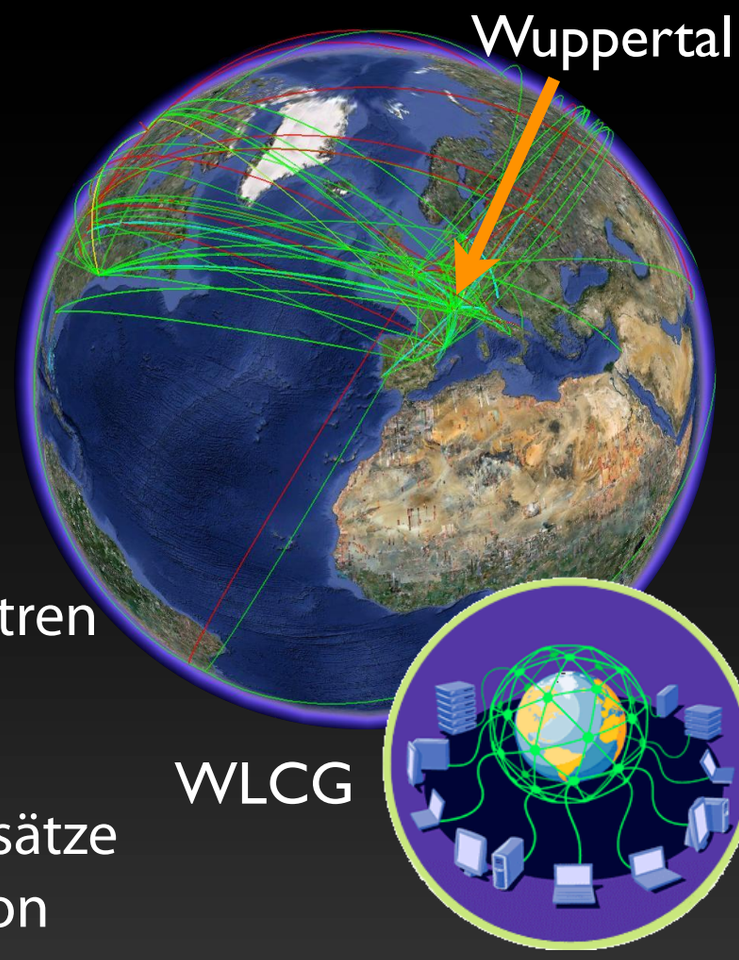
- Ereignis-Rekonstruktion
  - ➡ ausgefeilte Algorithmen zur Mustererkennung
    - insbesondere zur Spur-Rekonstruktion
  - ➡ Trigger : nur relevante Aspekte
  - ➡ Offline : volle Rekonstruktion
- CPU limitiert durch Pileup (!)



Das Computing-Modell der Experimente

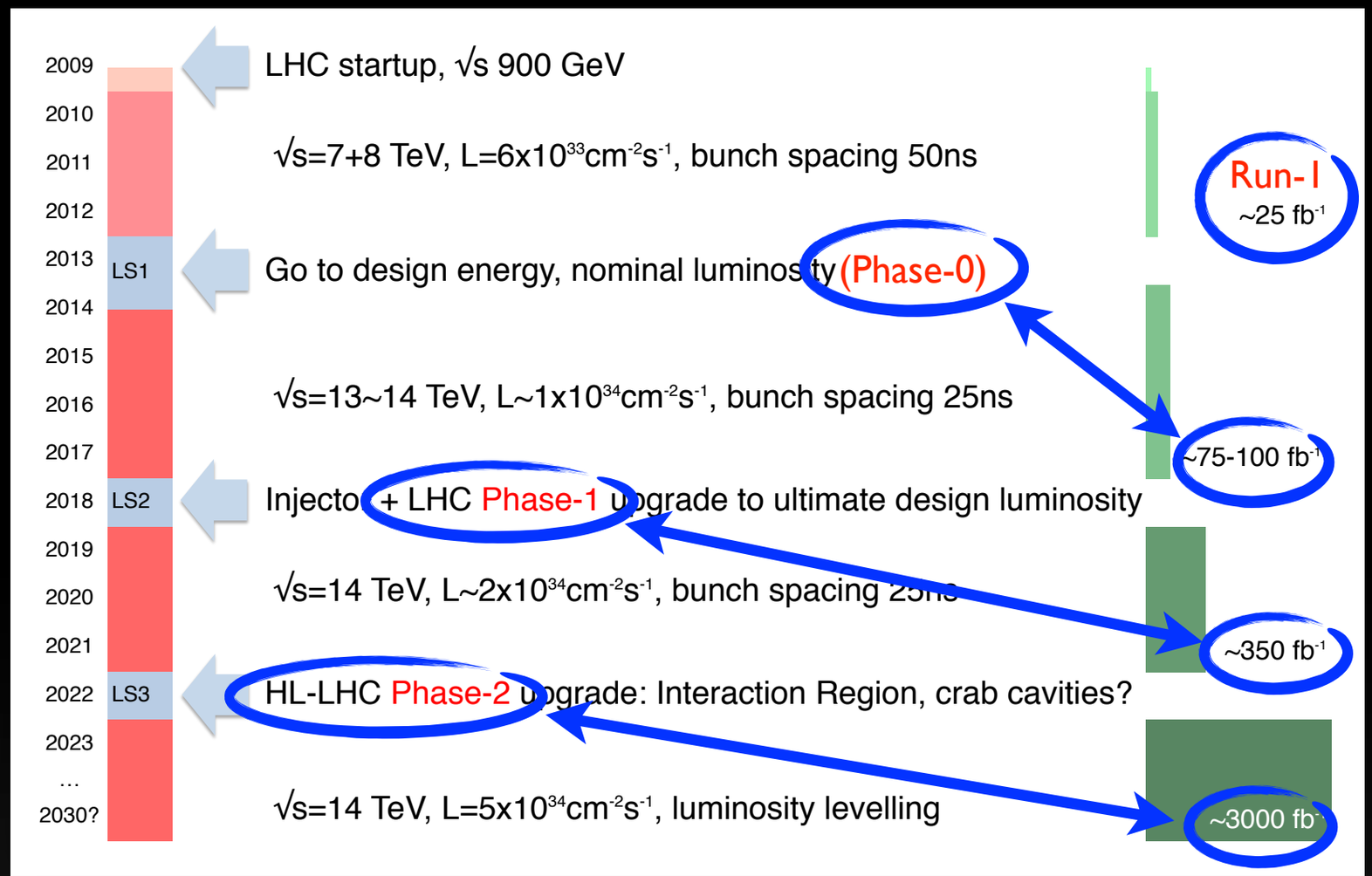


- Aufgaben des Offline-Computings:
  - ➡ Rekonstruktion der Ereignisse (hauptsächlich am Tier-0)
  - ➡ aufwendige Produktion von großen Monte-Carlo Datensätzen
    - Generatoren für Physik-Prozesse
    - sehr detaillierte Simulation der Detektoren (Geant4)
  - ➡ Physik-Analyse der Datensätze durch Forschungsgruppen
    - Datensätze der Experiment etwa 200 PB



- Worldwide LHC Computing Grid (WLCG)
  - ➡ Verbund von mehr als 150 Rechenzentren weltweit
    - GRID Middleware, Experimente haben eigene Produktionssysteme und Data-Management-Systeme
  - ➡ hierarchisch organisiert (Monarc-Modell):
    - Tier-0 am CERN (für direkte Rekonstruktion der Ereignisse)
    - Tier-1 Zentren weltweit, mit regional assoziierten Tier-2 Zentren
    - kleinere Tier-3 Farmen (lokale Datenanalyse an Instituten)
  - ➡ Daten werden vom CERN aus weltweit verteilt
    - Tier-1 sind Datenzentren und für Analyse der großen Datensätze
    - Tier-2 fungieren hauptsächlich als CPU-Farmen für Simulation

Der schrittweise Ausbau des LHC



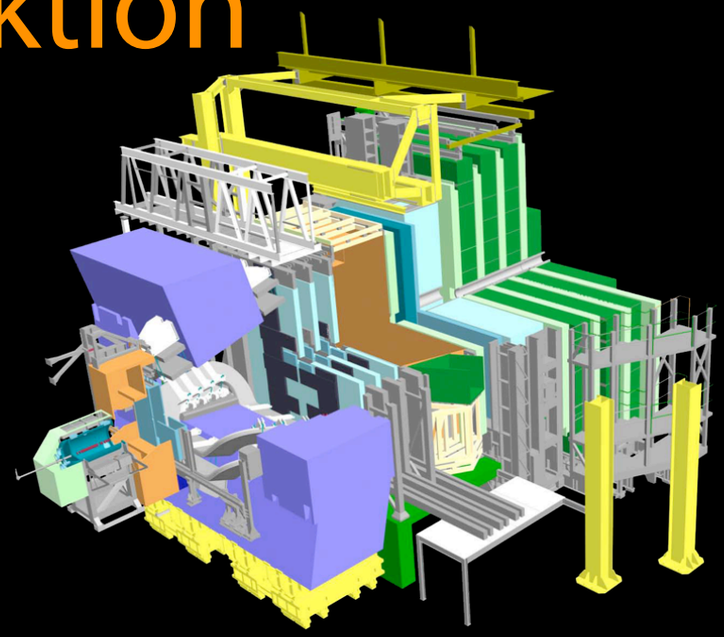
- Zeitplan
  - ➡ R.Heuer ('13)

- Verdopplung der Energie, schrittweise mehr Luminosität
  - ➡ Faktor 7 mehr Pileup als heute, Ereignisse fast um Faktor 2 größer
  - ➡ schrittweise Erhöhung der Datenrate von heute 400 Hz auf 5-10 kHz
  - ➡ CPU pro Ereignis und Speicherverbrauch der Rekonstruktion steigen rapide
- Herausforderung für Computing, Offline und Online

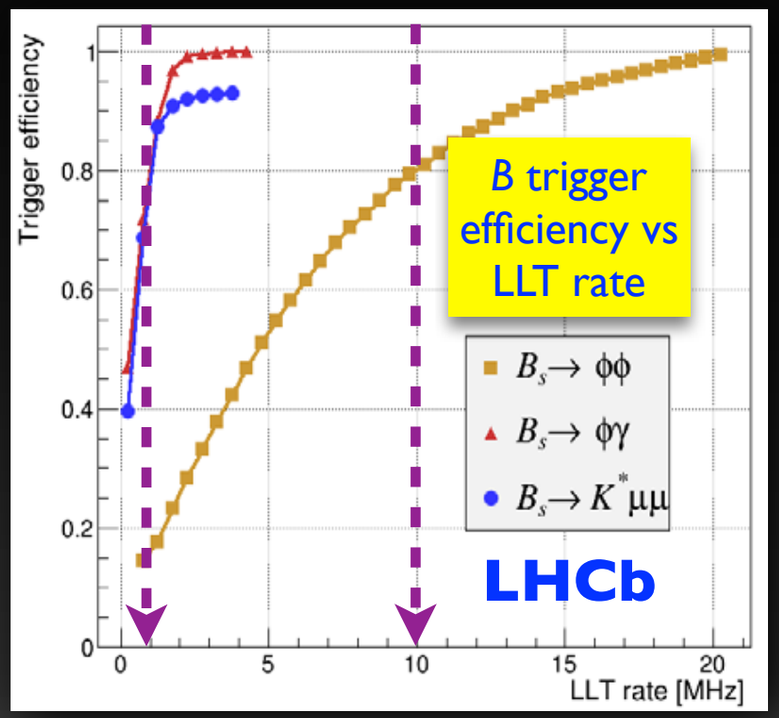
Strategien im Online-Computing

Neue Strategien der Trigger-Selektion

- LHCb-Experiment
  - ➡ Präzisionsmessungen zur indirekten Suche nach neuer Physik
  - ➡ keine Ereignisgröße ermöglicht schnelle Rekonstruktion



- neue Strategie: "Triggerless Readout"
  - ➡ neue, sehr schnelle Auslese des Detektors
    - "keine" Trigger-Selektion in 1.Stufe, spezielle Elektronik wird ersetzt
  - ➡ alle Ereignisse werden in erweiterter PC-Farm rekonstruiert
    - Selektion der interessanten Ereignisse mit voller Auflösung
  - ➡ Erhöhung der Luminosität um Faktor 5 und bessere Effizienz



40 *MHz*

ECal, HCal, Muons

LLT

p_T of h, μ, e/γ

custom electronics

1-40 *MHz*

full detector information

HLT

tracking, vertexing, inclusive/exclusive selection

CPU farm

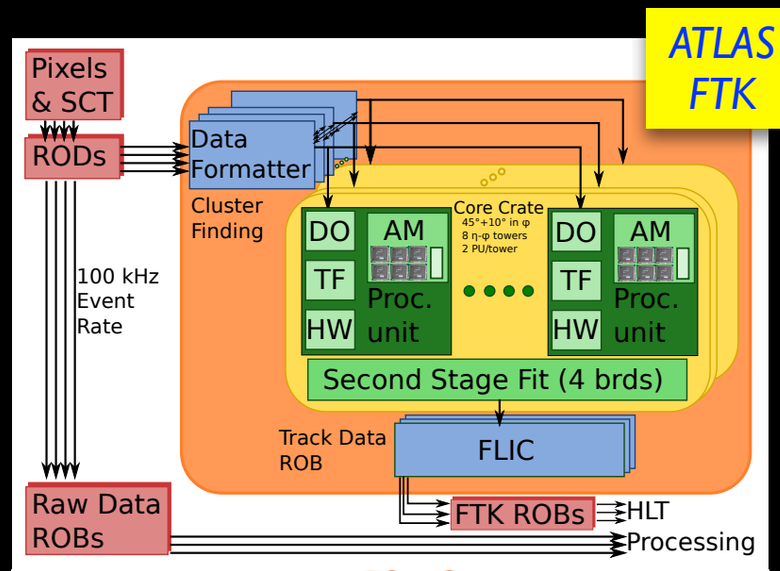
20 *kHz*

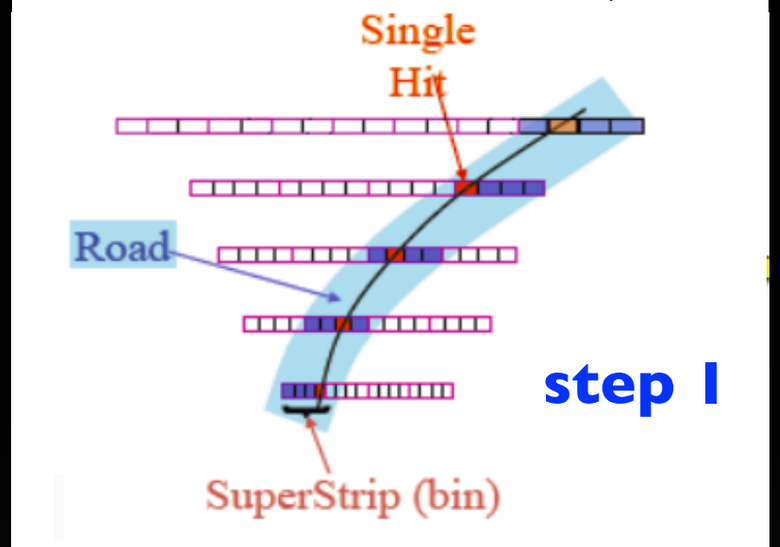
Co-Prozessoren im Trigger

- ALICE: seit 2010
  - ➡ FPGA-Karten zur Aufbereitung der Rohdaten
  - ➡ Spur-Rekonstruktion in Trigger PC-Farm mittels GPUs

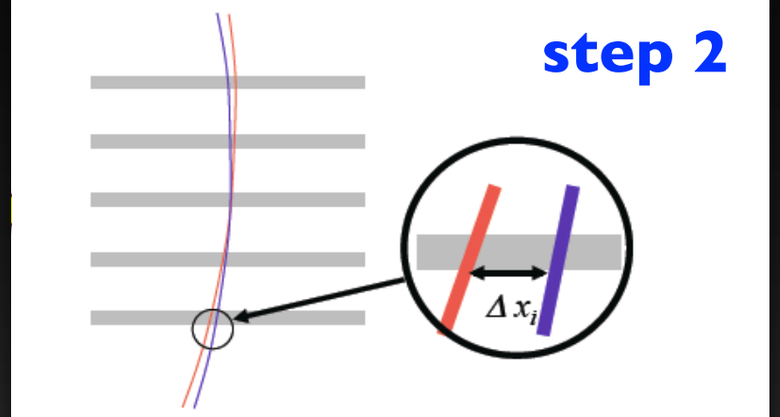
- ATLAS: Erweiterung in 2016
  - ➡ spezielle Elektronik zur schnellen Spur-Rekonstruktion der vollen Ereignisse

- ATLAS Fast TracKer (FTK)
  - ➡ Entwickelt aus CDF Spur-Trigger (SVT)
  - ➡ Rekonstruktion in 2 Stufen:
    - Suche der Spuren aus 50GB möglicher Kombinationen
    - schnelle (lineare) Abschätzung der Spur-Parameter und Selektion guter Spuren
  - ➡ Technische Umsetzung:
    - Associative Memory Karten für Suche nach Spuren
    - FPGA Karten für Berechnung der Spur-Parameter
  - ➡ volle Spur-Rekonstruktion in 25 µs
    - via spezieller Auslese des Detektors
    - Trigger PC-Farm startet von Spuren aus dem FTK





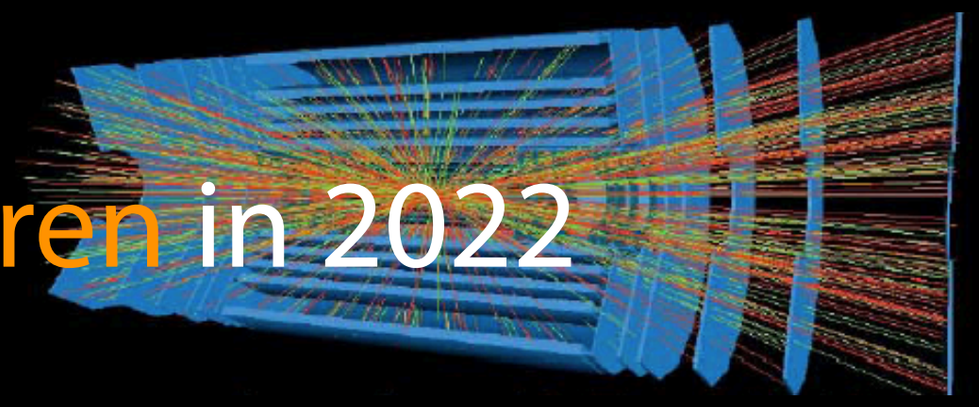
Pattern recognition in coarse resolution (superstrip➔road)



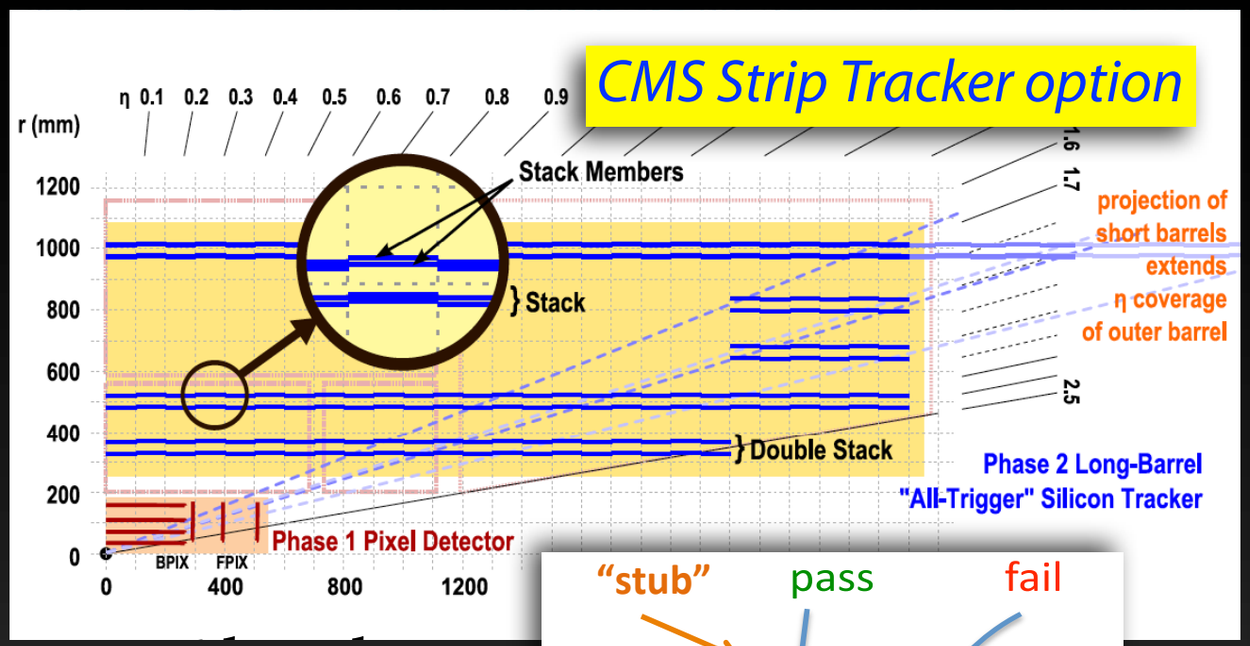
Track fit in full resolution (hits in a road)

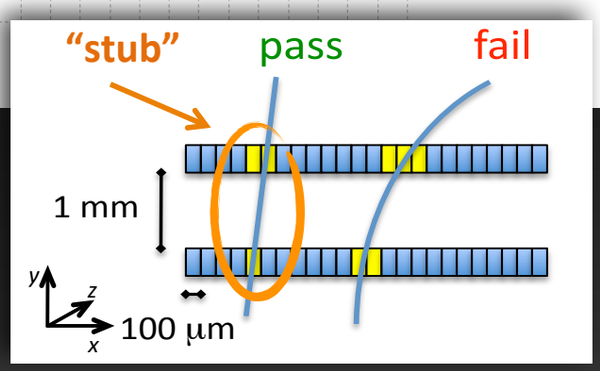
$F(x_1, x_2, x_3, \ldots) \sim a_0 + a_1\Delta x_1 + a_2\Delta x_2 + a_3\Delta x_3 + \ldots = 0$

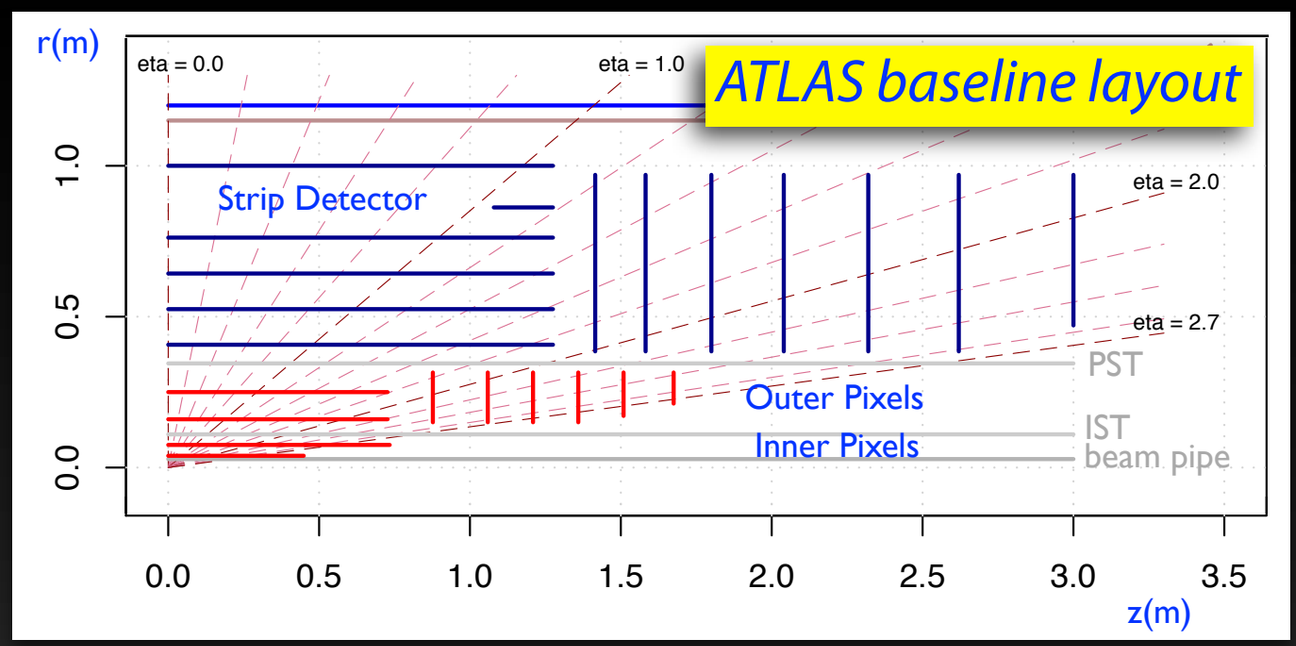
Verbesserte Spur-Detektoren in 2022



- vorgesehen für Datennahme bei sehr hoher Luminosität
  - ➡ spezielles Design, um Spur-Rekonstruktion in Elektronik zu unterstützen
    - CMS untersucht Auslese von Hit-Paaren in benachbarten Lagen
    - geometrische Unterdrückung von Untergrund aus falschen Kombinationen
  - ➡ sehr schnell, Spur-Findung vollständig Teil der 1.Stufe der Trigger-Selektion
    - Spur-Rekonstruktion in Trigger PC-Farm nur noch zur nachträglichen Verbesserung der Auflösung und Effizienz







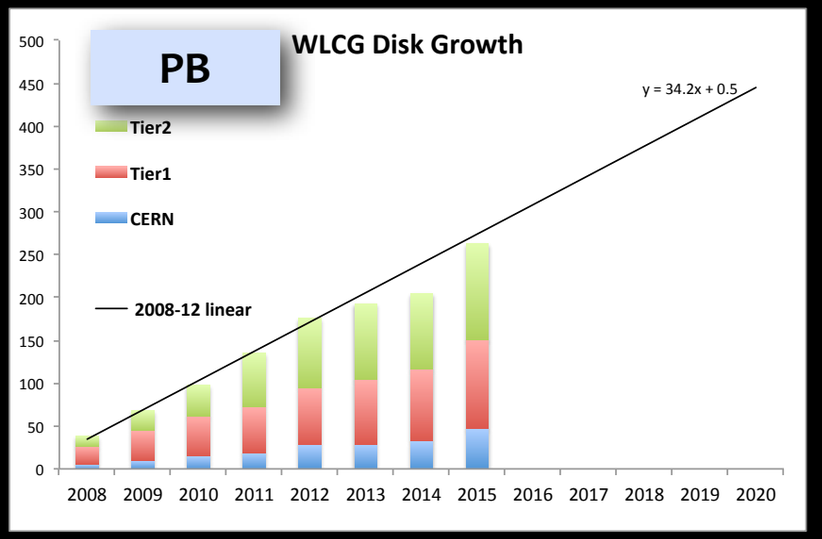
Strategien im Offline-Computing

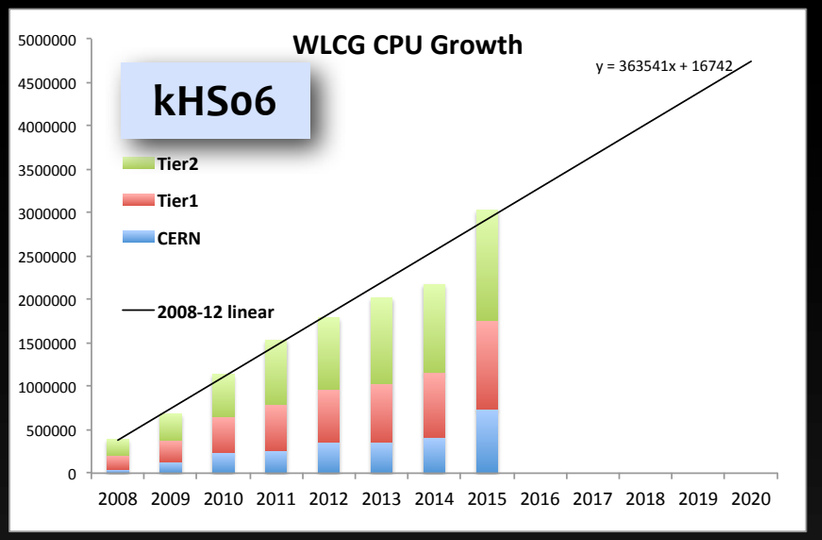
Projektion der WLCG Ressourcen

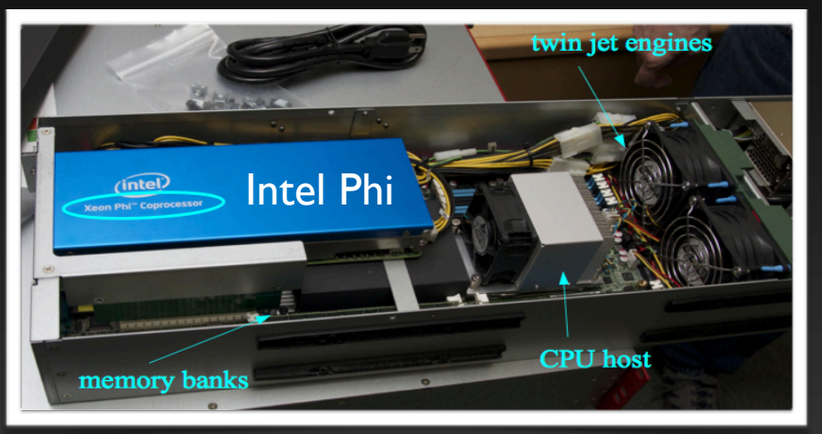
- Ausbau der bestehen Zentren
  - ➡ Wachstum hauptsächlich durch Fortschritte in der Technologie (CPU und Speichermedien)
  - ➡ wird nicht mit wachsenden Bedarf Schritt halten !

- heutiges Modell
  - ➡ X86 basierte CPU Server, 4 GB pro Prozessorkern
  - ➡ Anwendungen laufen “Ereignis”-parallel in separaten Prozessen
  - ➡ Jobs laufen nahe bei Daten, um Transfer zu vermeiden

- Entwicklung der Technologie
  - ➡ Bandbreite der Netzwerke nimmt schnell zu
    - Datentransfer kein Problem mehr
    - strenges Monarc-Modell wird flexibilisiert
  - ➡ moderne Prozessoren: Vektorisierung der Anwendung
  - ➡ “viel-Kern” Prozessoren wie Intel Phi (MIC), und GPUs
    - weniger Speicher pro Kern !



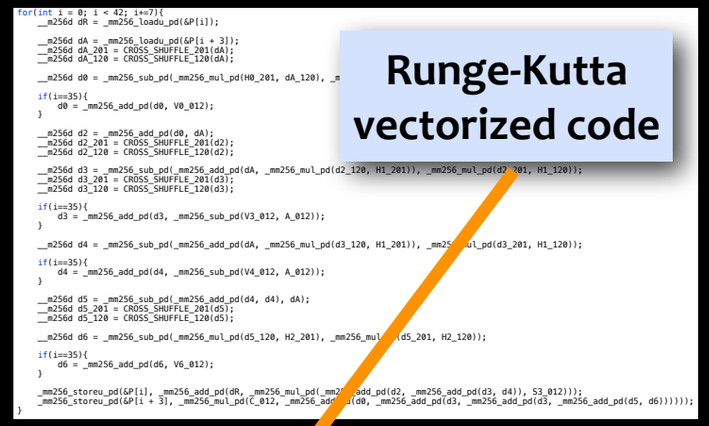


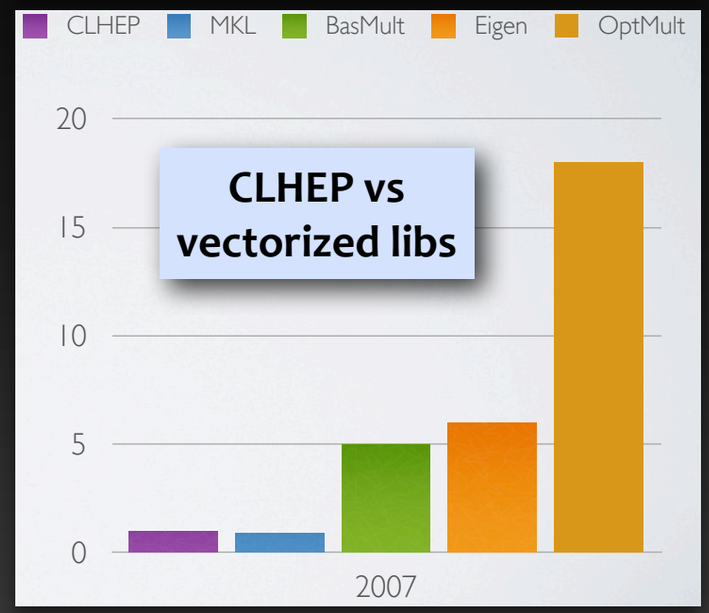


Anwendungen auf modernen Prozessoren

- Anwendung in der Hochenergiephysik
  - ➡ optimiert für Präzision der Physik
  - ➡ mehr als 10 Jahre Software-Entwicklung
    - Projekte haben viele 100 Entwickler (Studenten...)
    - komplexe Anwendungen, hoch spezialisiert

- Vektorisierung (SIMD)
  - ➡ moderne Prozessoren haben große Register
    - ausnutzen der Vektorisierung hat großes Potential
    - verlangt tieferes C++ Verständnis, nicht immer vorhanden bei Entwicklern
  - ➡ Erfahrung mit Auto-Vektorisierung im Compiler
    - HEP Software nicht einfach zu optimieren
  - ➡ dedizierte Vektorisierung kritischer Software
    - z.B. Runge-Kutta Transport geladener Spuren in Simulation und Rekonstruktion
  - ➡ Vektor-Bibliotheken für lineare Algebra und trigonometrische Funktionen
    - optimiertes Datenmodel für aufwendige Rechnungen

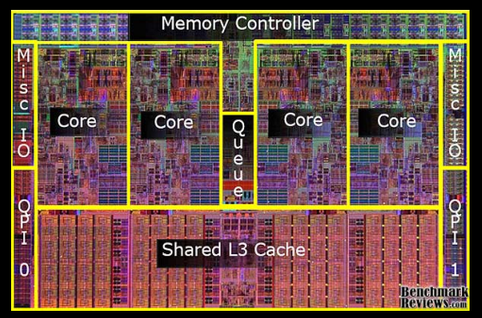




➡ siehe auch Vortrag von F. Wolfheimer, CST

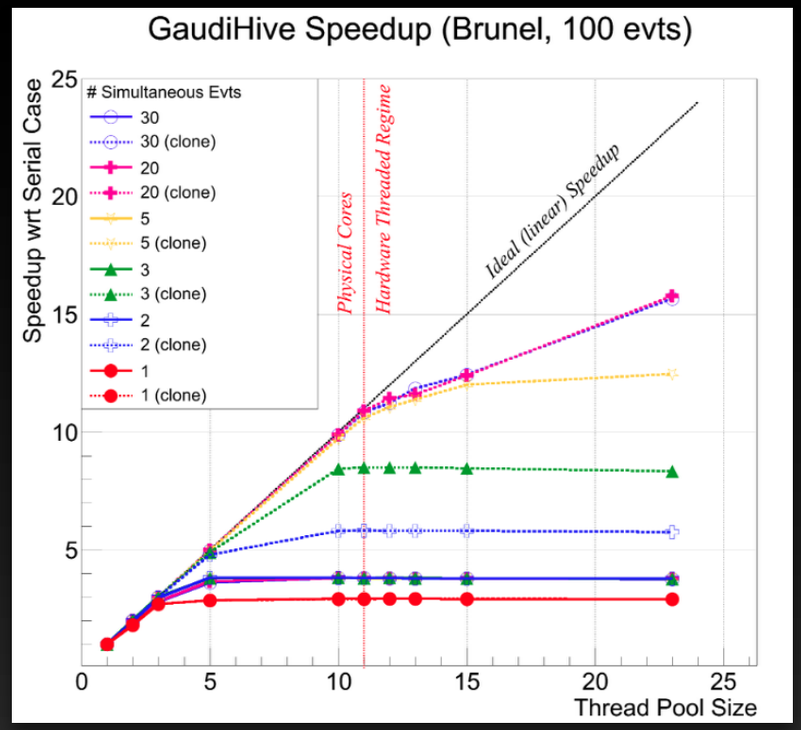
Entwicklung hin zu Multi-Threading

- heutige Anwendung sind "Ereignis"-parallel
  - ➡ typischerweise 2-4 GB an Speicher pro Kern notwenig
  - ➡ noch kein Problem auf heutigen CPU-Servern



- zukünftige "viel-Kern" Prozessoren
  - ➡ insbesondere weniger Speicher pro Kern
  - ➡ heutige Anwendung laufen nicht optimal
    - Speicherzugriffe und Bandbreite, usw.

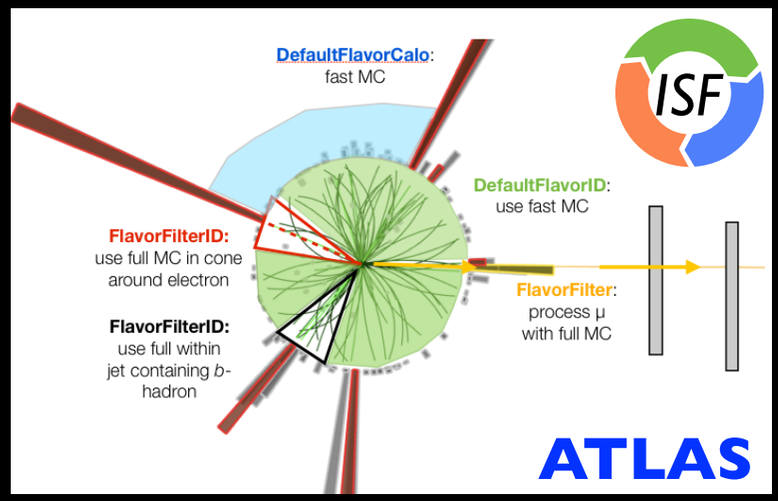
- kurzfristige Entwicklungen
  - ➡ Multi-Threading um Speicherbedarf zu optimieren
  - ➡ Verwendung von z.B. Intel TBB
    - Software-Infrastruktur unterstützt paralleles rechnen
    - Anwendungsbeispiele existieren (CMS, ...)

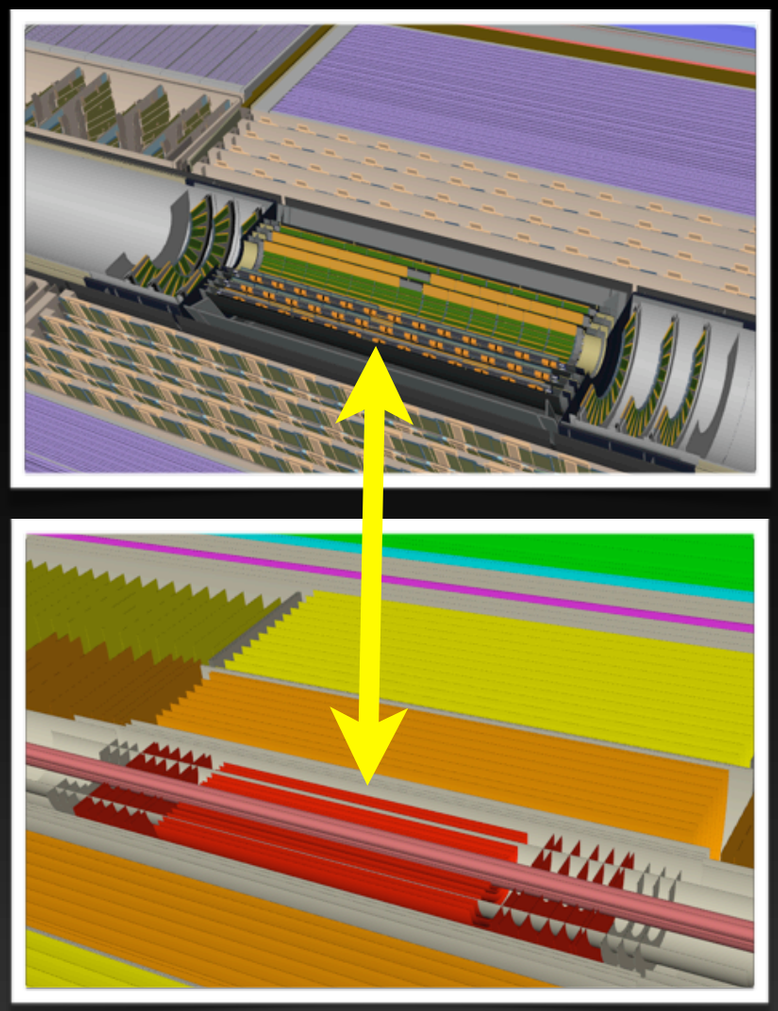


- langfristig Anpassung der Anwendung
  - ➡ Simulation: verschiedene Aufgaben auf Kerne verteilt
  - ➡ Gaudi Hive: paralleles Prozessieren einzelner Algorithmen

Optimierung der Anwendungen

- Prozessor Technologie nur ein Aspekt
  - ➡ Optimierung der Algorithmen ebenso wichtig
- Beispiel: Detektor Simulation
  - ➡ detaillierte Beschreibung gibt beste Resultate (Geant4)
  - ➡ vereinfachte Simulation ist schnell, aber weniger genau
- Integrated Simulation Framework (ISF)
  - ➡ erlaubt Kombination verschiedener Simulationsmethoden
    - interessante Teile der Ereignisse werden voll simuliert
    - andere Teile mit schneller Simulation abgehandelt
  - ➡ großes Einsparungspotential (CPU)
    - Ereignis in Geant4: 600 s
    - ATLFAST-IIF: 0.75 s
  - ➡ in der Zukunft:
    - Kombination mit schneller Rekonstruktion
    - Unterstützung von Multi-Threading
    - ggf. Möglichkeit, Co-Prozessoren einzubinden (GPUs)

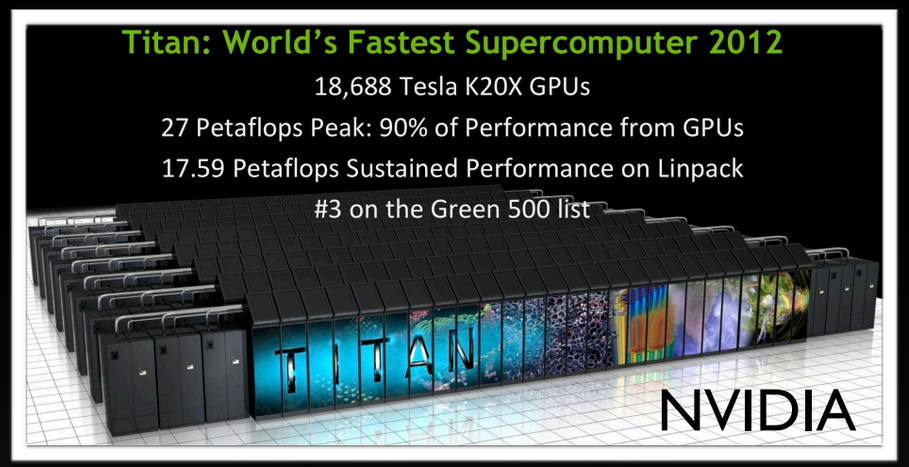




➡ siehe auch Vortrag von M.Sandhoff

High-Performance Computing in HEP

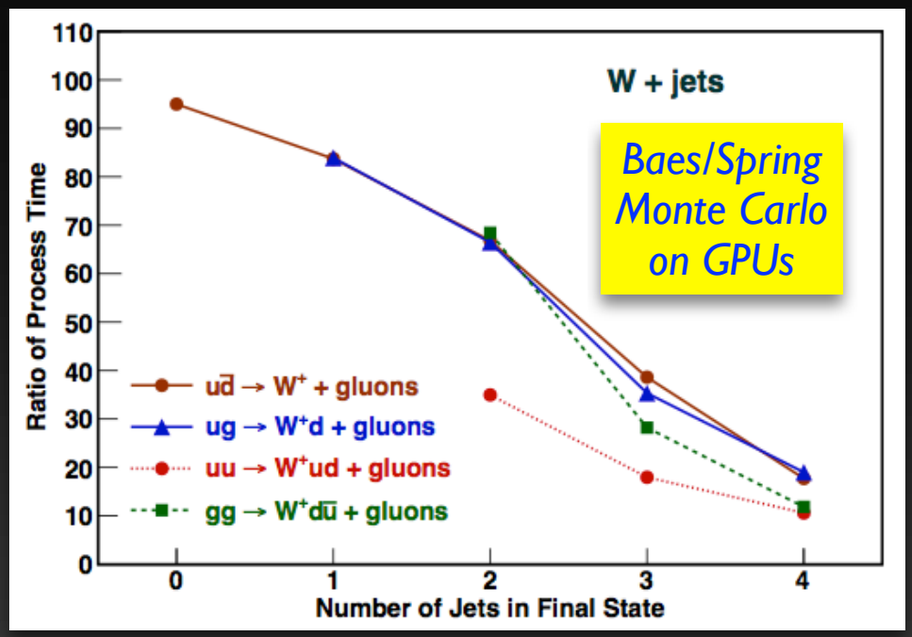
- Infrastruktur wird heterogener
  - ➡ Verwendung von zusätzlichen Ressourcen
    - kommerzielle Cloud-Anbieter (z.B. Google, Amazon)
    - freie CPU in High-Performance Computing Zentren
  - ➡ große HPC Zentren haben mehr CPU als WLCG
    - X86, BlueGene, NVIDIA GPUs, ARM, ...
  - ➡ GRID (ARC Middleware) oder Cloud (OpenStack) Interface



- geeignete Anwendungen
  - ➡ hoher Rechenbedarf bei geringen Datenmengen
  - ➡ Physik Generatoren, später Detektor Simulation

- X86 basierte Systeme
  - ➡ geringerer Aufwand zur Anpassung der Software

- GPU basierte Systeme
  - ➡ komplette Anpassung erforderlich (bisher)
  - ➡ z.B. Physik Generatoren VEGA und Baes/Spring



Zusammenfassung

- Computing für die LHC Experimente
  - ➡ Erfahrungen der ersten 3 Jahre sind sehr positiv

- der Ausbau des LHC in den kommenden Jahren
  - ➡ Anforderungen steigen dramatisch
  - ➡ neue Strategien für das Computing werden notwendig

- neue Technologien und Entwicklungen
  - ➡ sowohl im Online-, als auch im Offline-Computing
  - ➡ Co-Prozessoren und Entwicklung neuer Detektor-Konzepte
  - ➡ neue Ansätze zur Optimierung der Algorithmen
  - ➡ Vektorisierung und Multi-Threading auf modernen Prozessoren
  - ➡ Nutzung von freien Ressourcen in High-Performance Computing Zentren für spezielle Anwendungen