Компьютерные технологии в физике элементарных частиц.

Обработка данных

Введение

В ранних экспериментах в физике частиц (астрофизике,…) можно было зарегистрировать одно событие и сделать открытие. В современных экспериментах исследуются очень редкие или очень слабые явления, для их изучения необходимо проанализировать огромный массив данных.

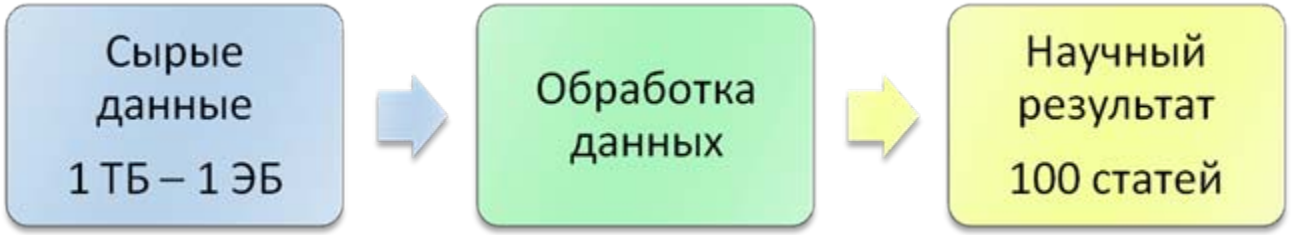
- Объем данных в небольшом эксперименте: 1-100 ТБ
- Объем данных в больших экспериментах: 100 ТБ – 10 ПБ
- Объем данных в экспериментах следующего поколения: 10 ПБ - 1 ЭБ

Анализ данных усложняется множеством факторов:

- Не все частицы регистрируются в детекторе – в детекторе есть “слепые” зоны.
- Системы детектора измеряют параметры частиц не с бесконечной точностью – необходимо учитывать разрешение и особенности работы измерительной системы.
- Не всегда удается уникально идентифицировать тип частицы, или определить, сколько частиц прошло через регистрирующую систему – необходимо учитывать эффективность, фон.
- Невозможно зарегистрировать все события – необходимо отобрать интересующие события и проанализировать, как критерии отбора влияют на результат.

От сырых данных к открытиям

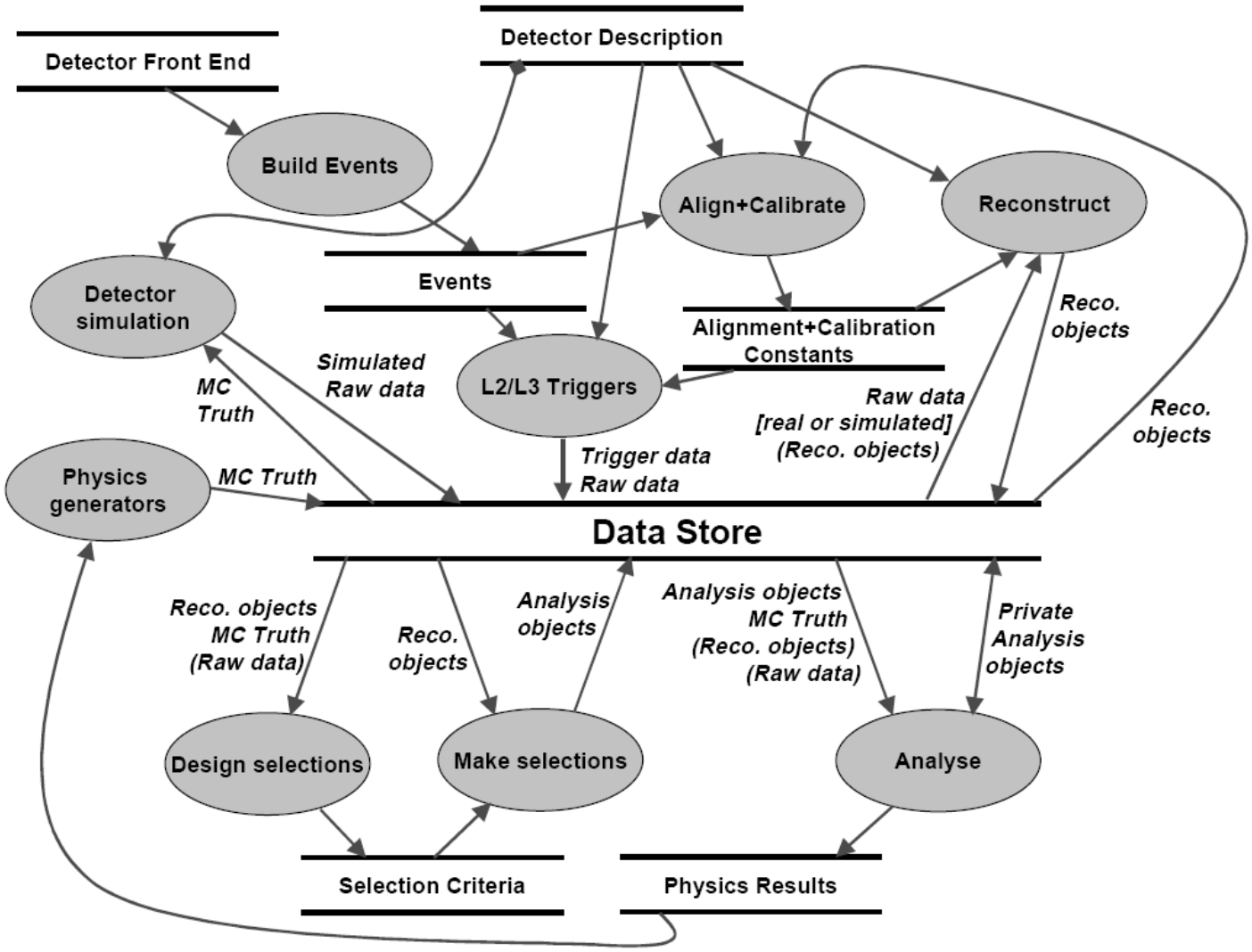
Если представлять обработку данных эксперимента в виде черного ящика, то на вход этого ящика поступает 1 ТБ (1 ПБ, 1 ЭБ) сырых данных, а на выходе публикуется ~100 статей в научных журналах (~1 ГБ).



Как достичь фактора сжатия 10^3-10^9? Составные элементы обработки данных:

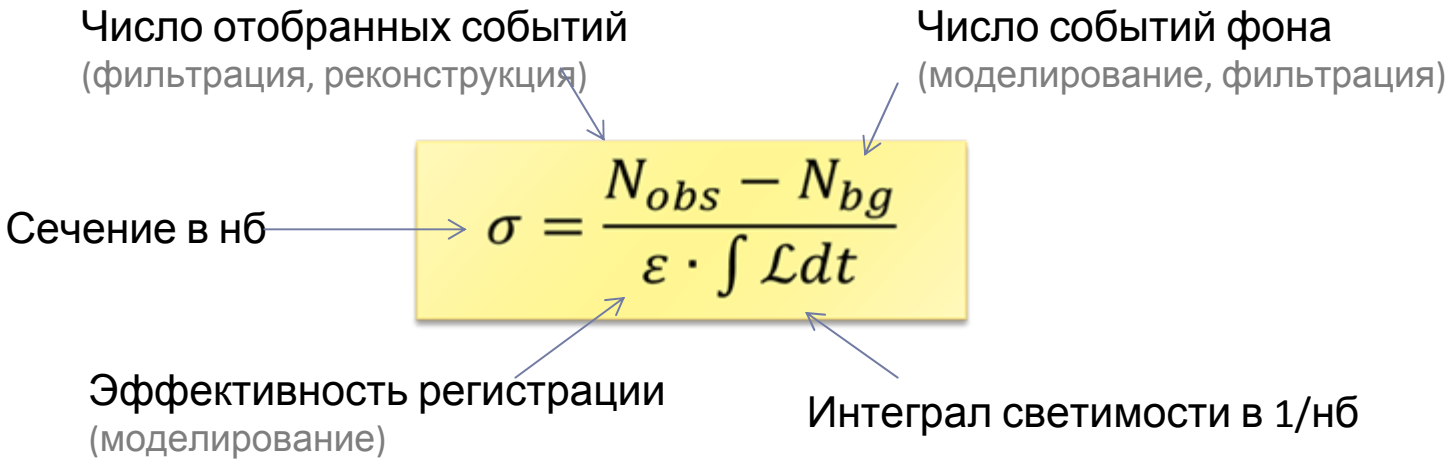
- Хранение, классификация и отбор данных
- Калибровка систем детектора
- Реконструкция данных
- Моделирование измерительной системы
- Физический (научный) анализ

Пример: составляющие обработки данных на LHCb



Пример: измерение сечения

Сечение – вероятность того, что произойдет событие определенного типа. Огромное количество знаний о том, как устроен мир, основано на изучении того, как сечение зависит от типа события, энергий, углов,…



Цель – измерить сечение с наилучшей относительной точностью. В простейшем случае минимизируем:

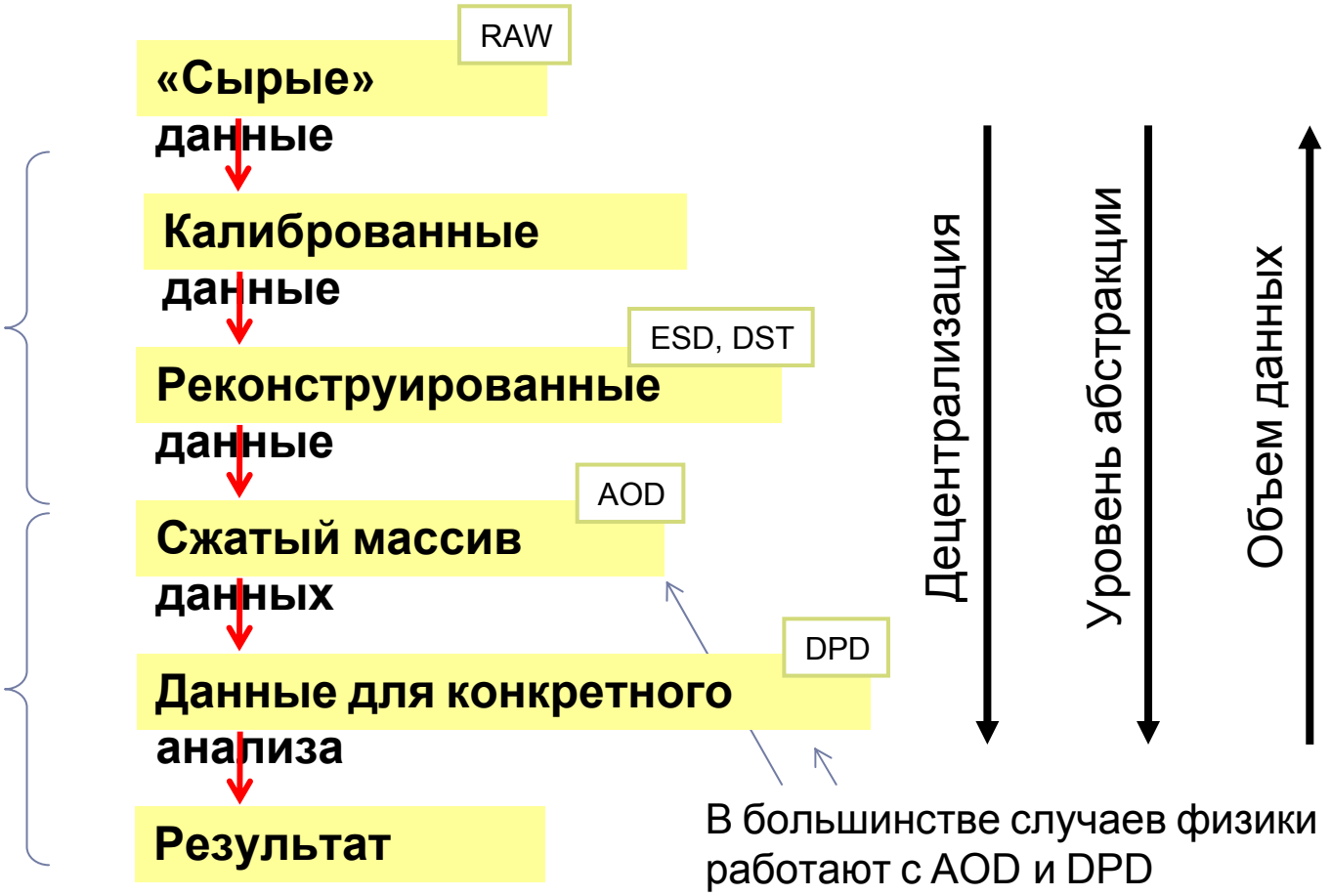
$$\frac{\delta\sigma}{\sigma} = \sqrt{\frac{\delta N_{obs}^2 + \delta N_{bg}^2}{\left(N_{obs} + N_{bg}\right)^2} + \left(\frac{\delta\mathcal{L}}{\mathcal{L}}\right)^2 + \left(\frac{\delta\varepsilon}{\varepsilon}\right)^2}$$

Этапы обработки данных

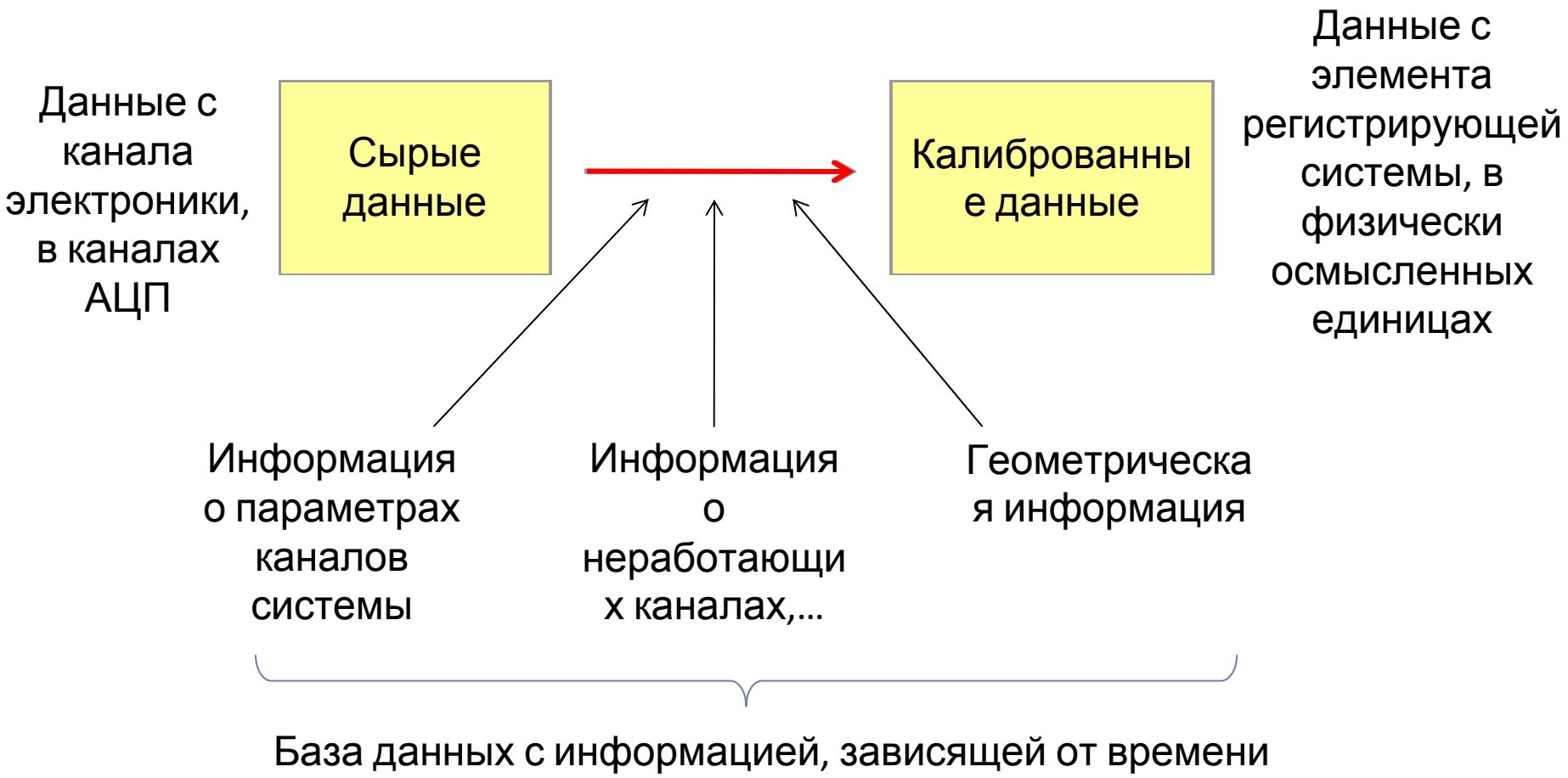
Обработка данных – всегда многоуровневый процесс. На каждом шаге обработки объем данных уменьшается, а уровень их абстракции растет.

Этапы, выполняемые на больших центральных кластерах, в идеале – один раз

Этапы, выполняемые многократно во многих местах



Калиброванные данные



Процедура получения калибровочной информации уникальна для каждой системы.

Калибровка

Процесс калибровки уникален для каждой измерительной подсистемы (или класса подсистем). Как правило, возможности проведения калибровки должны быть заложены еще на этапе проектирования детектора.

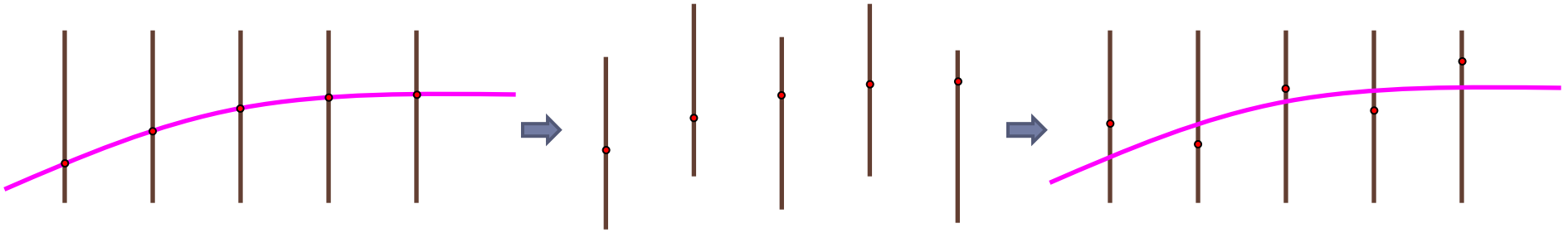
Распространенные подходы:

- Специальные системы: лазерная калибровка (луч лазера – идеально прямая линия), «внедренные» радиоактивные источники, светодиоды,…
- Использование самих измеряемых данных: космические частицы, каналы с монохроматическими частицами (например, $e^+e^- \to e^+e^-$), учет случайности входных данных (например, калибровка времени дрейфа), подбор калибровочных параметров так, чтобы достичь наилучшего разрешения, сравнение с моделированием,…

Как правило, калибровка производится периодически, и в период между проведением калибровок важно осуществлять мониторирование параметров измерительной системы.

Калибровочные параметры (коэффициенты пересчета, списки неработающих каналов и т.п.) должны храниться в базе данных калибровок вместе со всей историей.

Пример: калибровка положения элементов трековой системы

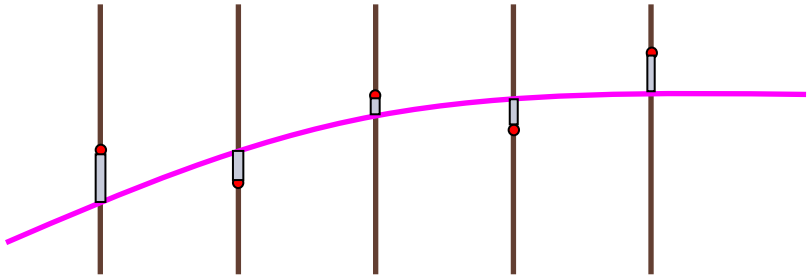


Идеальная система

В реальности положения камер отличаются от идеальных

Без учета реального положения камер ухудшается точность измерения координаты

Идея калибровки: подобрать положения элементов так, чтобы минимизировать расстояние между точками и траекторией для большого количества частиц – сотни тысяч параметров!



$$\chi^2(\mathbf{p},\mathbf{q}) = \sum_{j}^{tracks} \sum_{i}^{hits} \mathbf{r}_{ij}^{T}(\mathbf{p},\mathbf{q}_j)\mathbf{V}_{ij}^{-1}\mathbf{r}_{ij}(\mathbf{p},\mathbf{q}_j)$$

Реконструкция данных

Задача этапа реконструкции данных – найти, какие частицы вызвали срабатывание систем детектора, и определить их параметры. Это сложный многостадийный процесс, для реконструкции применяется множество различных алгоритмов и сложная статистическая обработка.

Конкретные алгоритмы, применяемые для реконструкции данных, выбираются исходя из особенностей конструкции детектора и особенностей регистрируемых данных (например, множественности частиц в конечном состоянии).

Типичная схема реконструкции данных в эксперименте в ФВЭ:

- Реконструкция заряженных частиц
- Реконструкция нейтральных частиц
- Идентификация частиц
- Реконструкция вершин взаимодействия
- Реконструкция кинематики взаимодействия

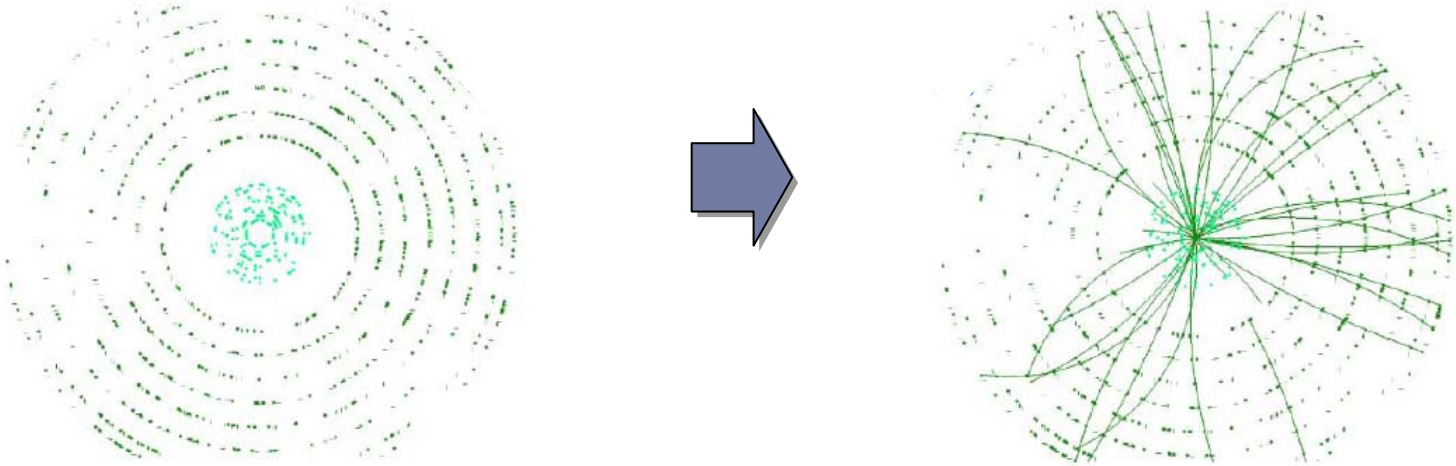
Результаты реконструкции сохраняются в виде коллекций треков, кластеров, вершин,... - Event Summary Data (ESD).

Реконструкция заряженных частиц (1)

Заряженные частицы регистрируются трековой системой, калориметром и мюонной системой.

Как правило, трековая система помещена в магнитное поле, что позволяет определить импульс заряженной частицы.

Задача реконструкции: по отдельным зарегистрированным точкам (хитам) восстановить траекторию движения частицы (трек) и ее параметры (точку вылета, угол, импульс,...)



Реконструкция заряженных частиц (2)

Как правило, задача реконструкции треков распадается на две подзадачи:

- поиск треков

  Выделение подмножеств точек, которые принадлежат одному треку: поиск “коридоров”, преобразование Хью, итерационный поиск в пространстве параметров, нейронные сети,…

- определение параметров трека («подгонка»)

  Статистическая обработка: метод наименьших квадратов, фильтр Калмана,…

Иногда эти шаги выполняют совместно.

Сложности:

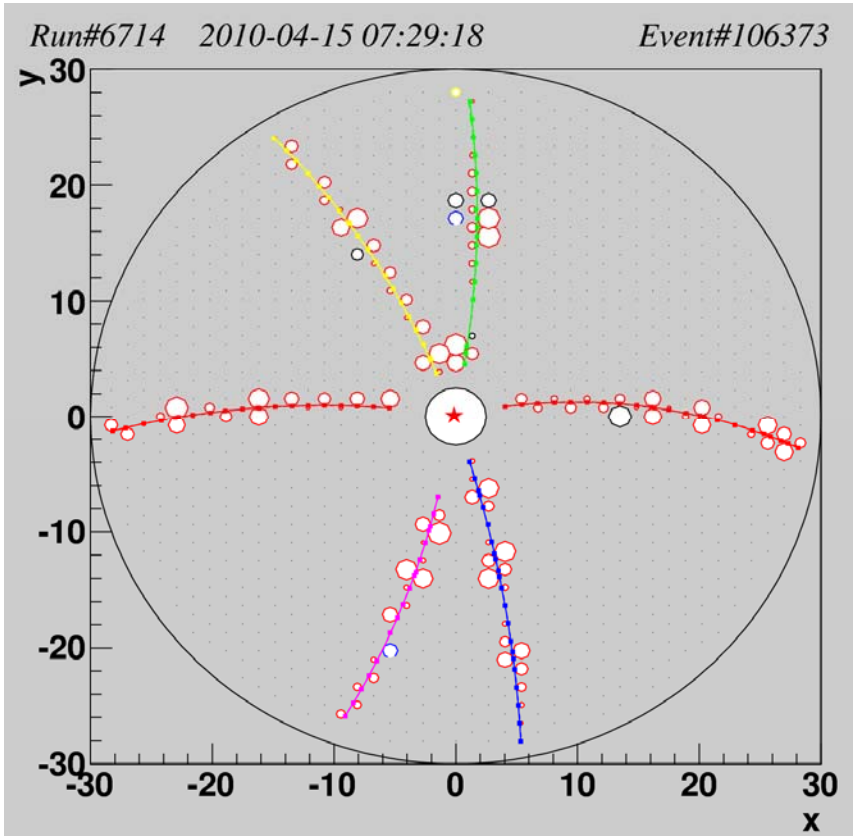
•Траектория частицы может сильно отличаться от прямой линии или спирали – она искажается случайным или систематическим образом из-за потери энергии за счет ионизации и многократного рассеяния.

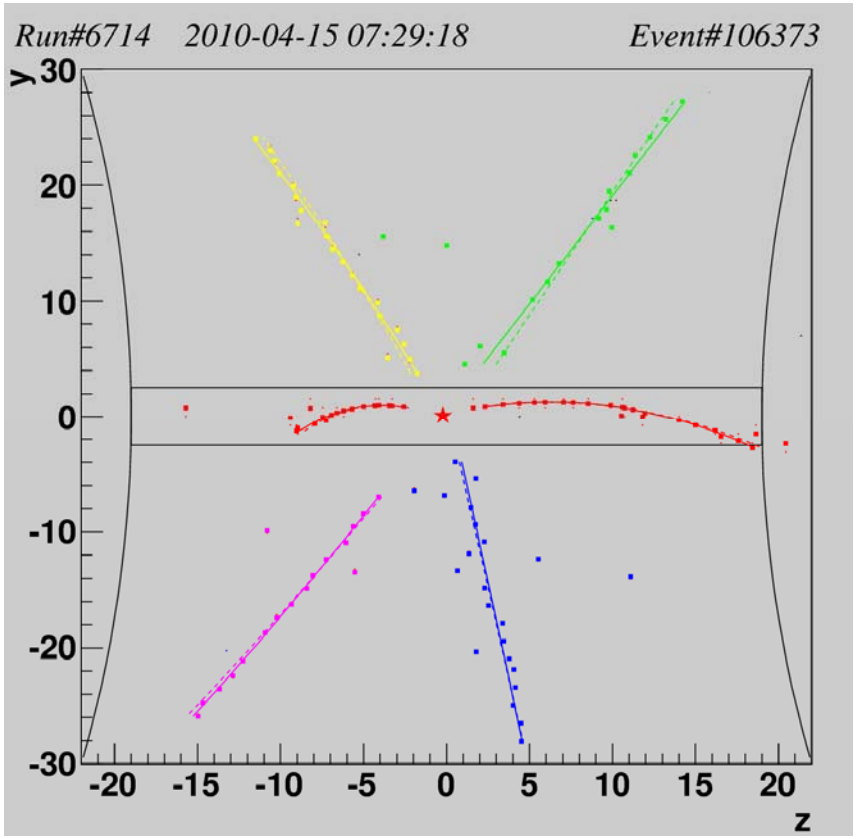
•Изначально неизвестно, к какому треку относится каждый хит.

•Заряженная частица взаимодействует с веществом на протяжении всей траектории, а регистрируется только та часть траектории, которая попала в чувствительный объем системы.

•Необходимость учета неоднородностей магнитного поля.

Пример: треки в КМД-3

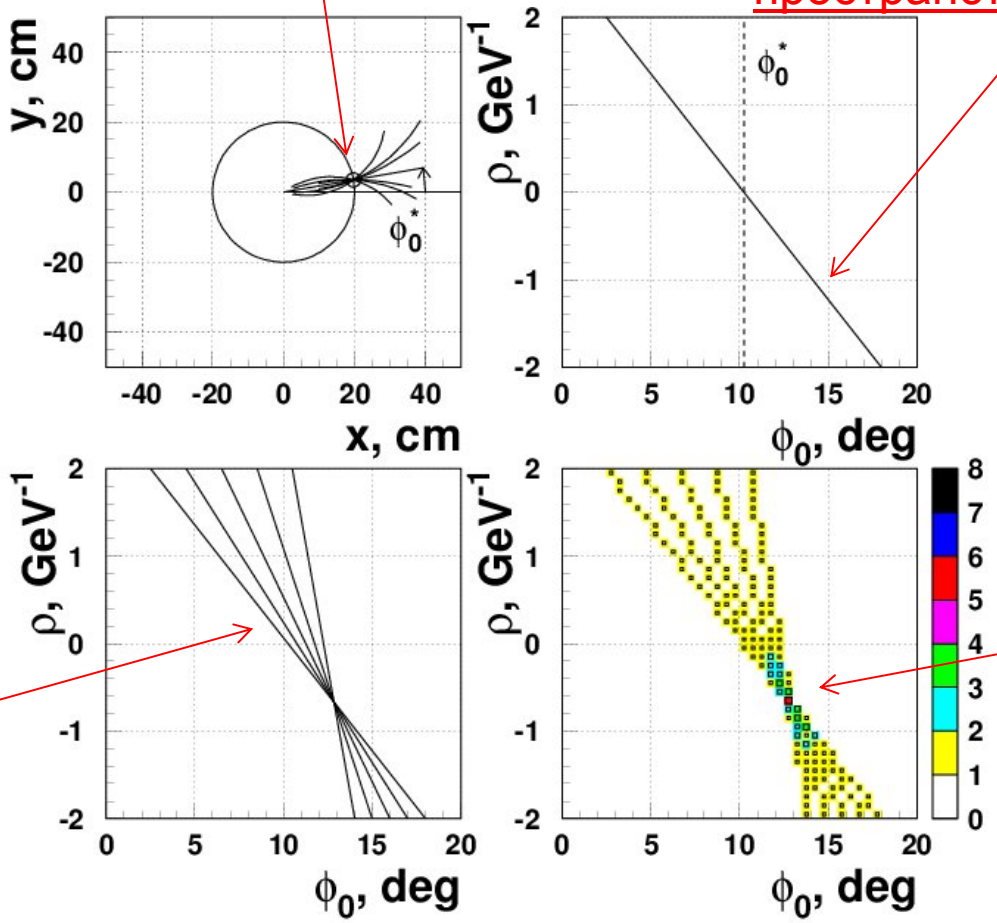




$$e^+e^- \to 6\pi$$

Пример: преобразование Хью

Каждой точке в реальном пространстве соответствует линия в пространстве параметров



Каждой точке соответствует своя линия

Пик в пространстве параметров соответствует треку

Реконструкция нейтральных частиц

Нейтральные частицы регистрируются в основном только калориметром.

Некоторые нейтральные частицы реконструируются только по продуктам своего распада (нейтральным или заряженным): например, π^0 по двум фотонам, или K_S по двум заряженным пионам.

Задача реконструкции: по отдельным сработавшим элементам калориметра сформировать кластеры и восстановить параметры (энергию, направление движения) частиц, которые вызвали срабатывание калориметра.

Сложности:

- Энерговыделение в калориметре зависит от типа частицы.
- Кластеры от разных частиц могут перекрываться.
- Энерговыделение струй в адронном калориметре сильно зависит от электромагнитной составляющей.

Существует множество алгоритмов формирования кластеров, разделения наложившихся кластеров и т.п.

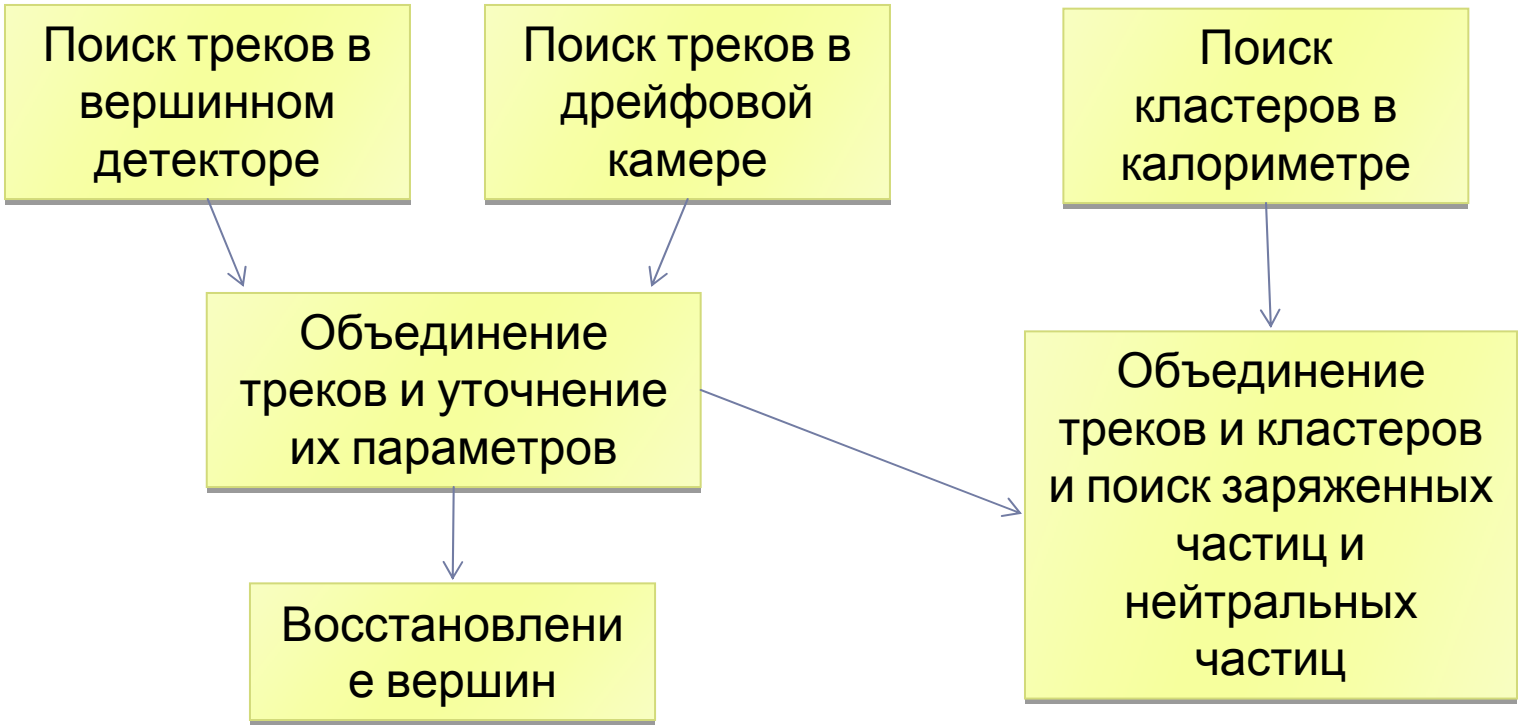
Другие шаги реконструкции

1. Идентификация частиц – определение типа заряженной или нейтральной частицы по отклику детектора. Для идентификации частиц используются:
   - специализированные детекторные системы: пороговые черенковские счетчики, RICH, измерение ионизационных потерь,...
   - объединение информации с разных подсистем: например, γ-квант, если есть кластер, но нет трека.
2. Восстановление вершин – поиск общей точки вылета для нескольких заряженных или нейтральных частиц. Вполне решаемая задача в случае первичной вершины, и очень сложная задача в случае поиска вторичных (распадных) вершин. Существует множество алгоритмов поиска вершины.
3. Восстановление кинематики события, «кинематический фит» - можно улучшить точность определения параметров частиц, если учесть законы сохранения энергии и импульса. Для этого строится функция правдоподобия, в которой законы сохранения учтены в виде множителей Лагранжа. Основная сложность состоит в правильном учете влияния разрешения измерительной системы с учетом всех корреляций.

Организация программы реконструкции

Реконструкция данных – это сложный многостадийный процесс, для реконструкции применяется множество различных алгоритмов и сложная статистическая обработка. Поэтому программы реконструкции, как правило, организованы как системы модулей, связанных единым каркасом.

Пример стадий реконструкции:



Framework

Типичный размер программы реконструкции данных с детектора – от сотен тысяч до миллионов строк кода. Над программой реконструкции работает от 10 (на небольшом эксперименте) до 1000 (на LHC) человек, причем их уровень как программистов различается на порядок. Единственный способ организовать работу такого масштаба – создать единую «культурную среду». Роль такой среды выполняет программный каркас (framework).

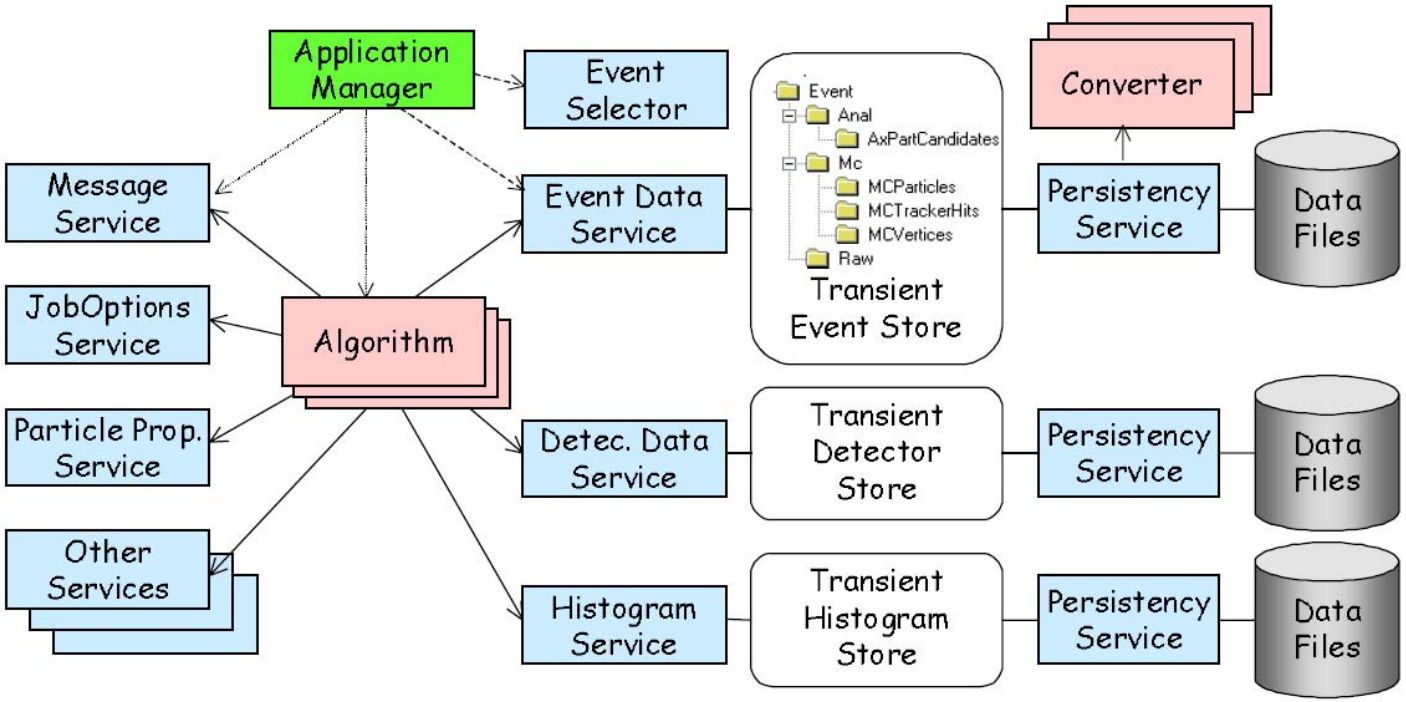
Составляющие элементы программного каркаса:

Существуют универсальные каркасы, которые используются в разных экспериментах, например GAUDI.

Framework GAUDI

Примером широко используемого программного каркаса является GAUDI. Изначально он разрабатывался для эксперимента LHCb, в настоящее время используется многими экспериментами (LHCb, ATLAS, BES-3, CMD-3,…)

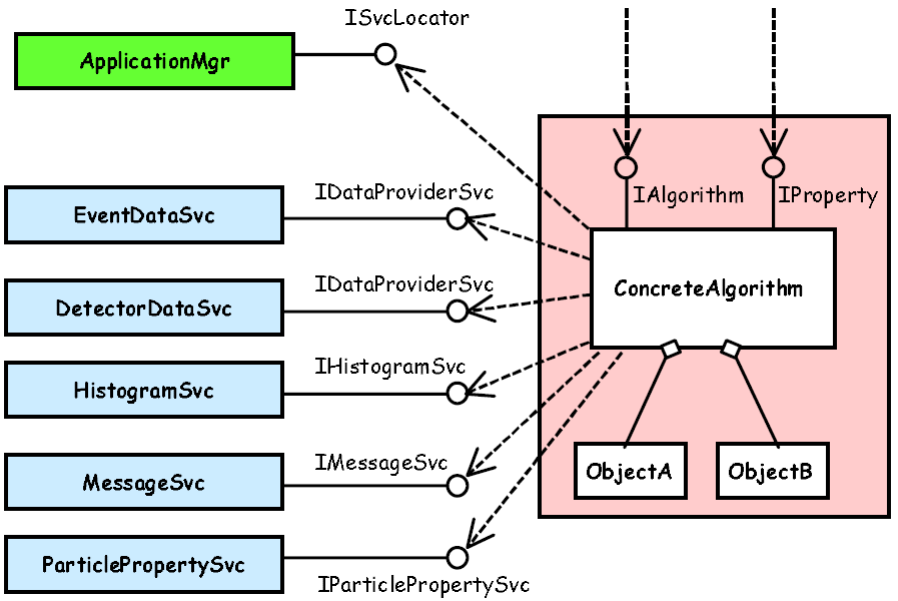
Архитектура GAUDI:



Модуль реконструкции в GAUDI

Ключевым элементом каркаса является модуль (Algorithm). Модули разрабатывают конечные пользователи, GAUDI определяет интерфейс модуля и предоставляет ряд стандартных сервисов:

- включение модуля и его параметризация
- доступ к входным данным и хранение выходных данных
- доступ к калибровочной информации
- механизмы общения с другими модулями
- механизмы хранения мониторирующих гистограмм



Результат реконструкции – коллекции объектов, описывающих треки, кластеры, частицы,..., и связей между ними

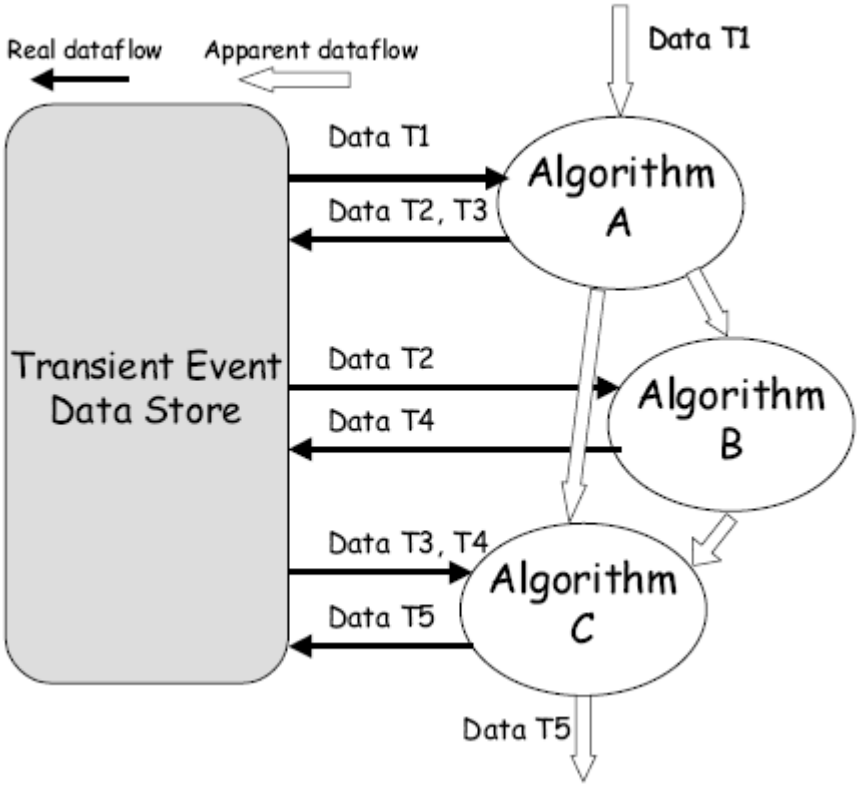
Обмен данными между модулями в GAUDI

В GAUDI используется непрямой обмен данными между модулями:

• все данные хранятся в памяти в специальном хранилище (data store) и доступны по ключам (имя данных, тип данных,…);

• каждый модуль работает только с data store – читает входные данные и сохраняет выходные данные;

• для долгосрочного сохранения данных специальный модуль фильтрует данные из data store и преобразует их во внешний формат.



Такой подход используется в большинстве современных программных каркасов.

Фильтрация событий

После того, как события реконструированы, необходимо в общем массиве зарегистрированных событий найти те, которые нас интересуют.

$$\sigma = \frac{N_{obs} - N_{bg}}{\varepsilon \cdot \int \mathcal{L} dt}$$

Как правило, в больших экспериментах это делается в два этапа.

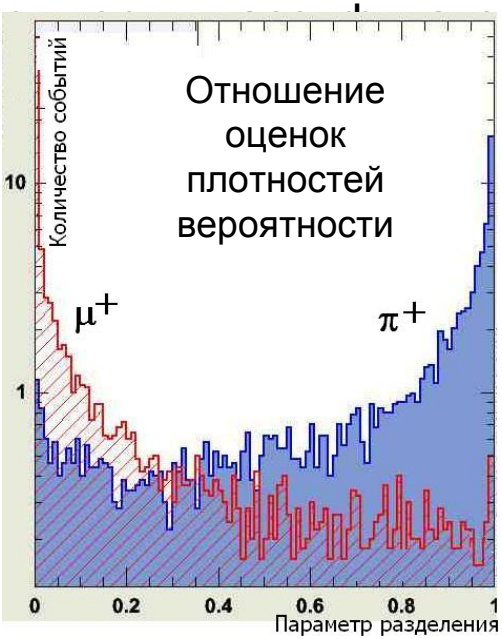
1. Централизованная фильтрация событий. На этом этапе весь объем данных разбивается на достаточно широкие подмножества, которые интересны при изучении целого класса конечных состояний. Например: события с 4 и более заряженными частицами. На этом этапе сжимается и содержимое событий – оставляется только наиболее необходимая информация (Analysis Object Data, AOD). Вместо фильтрации может производиться формирование тегов событий (event tags).

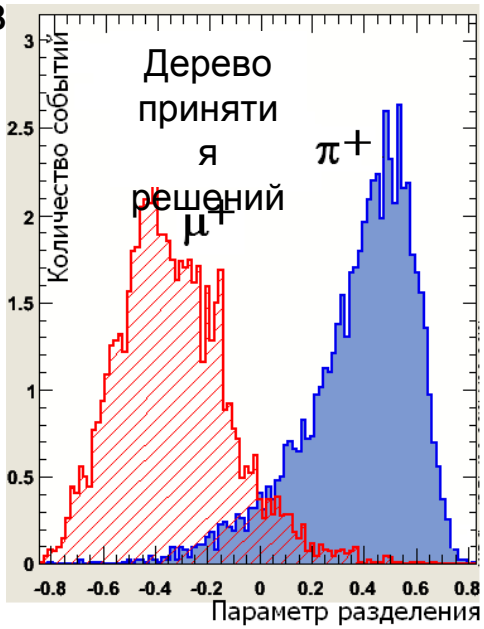
2. Выделение событий для конкретного научного анализа. Каждый пользователь (группа) самостоятельно более детально фильтрует AOD, отбирает события для конкретного анализа и сохраняет их в виде Derived Physics Data (DPD).

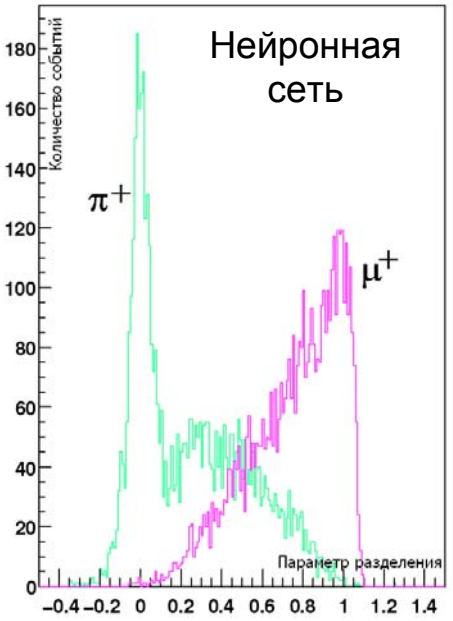
Выделение событий

Существует множество подходов к выделению интересного класса событий. Наиболее распространён метод наложения критериев отбора – он наиболее предсказуем и прост с точки зрения анализа систематических ошибок. Все чаще применяются методы многомерного статистического анализа (MVA), в которых события отбираются с помощью классификатора (параметра разделения), объединяющего информацию о многих признаках события.

П ров

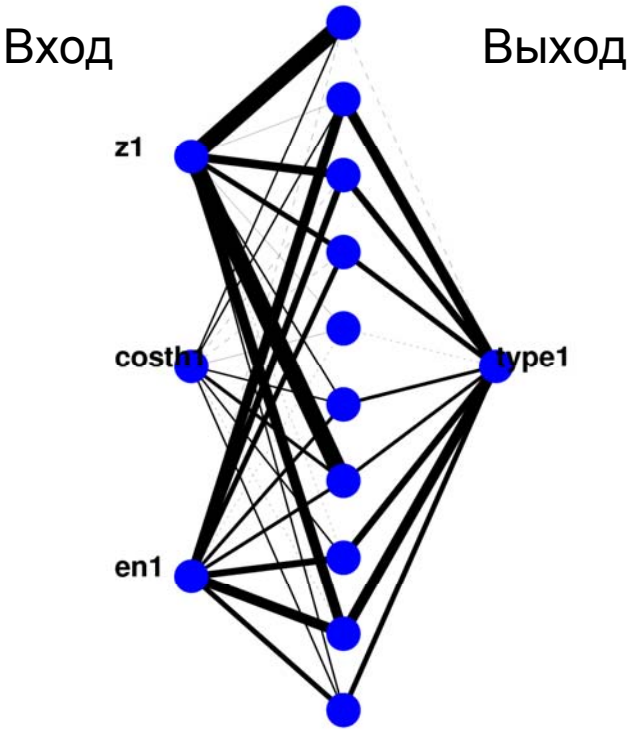


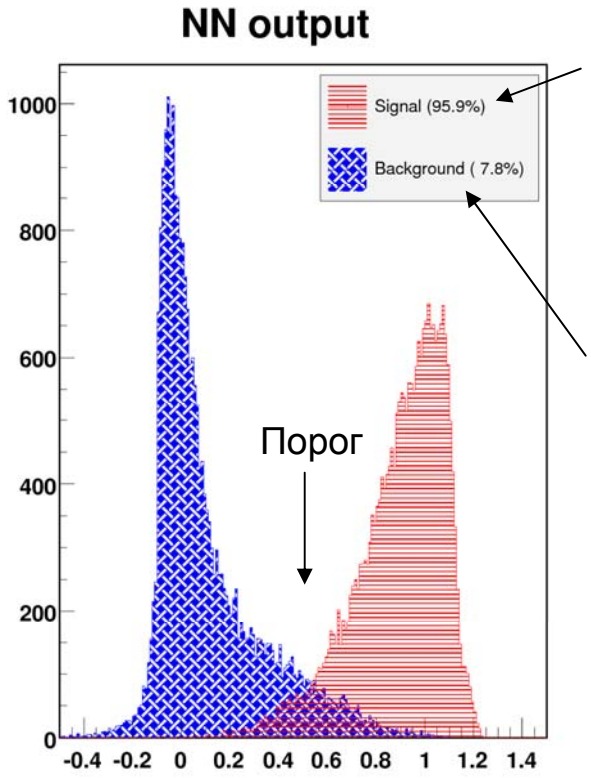




Пример: классификация событий с помощью нейронной сети

Подаем на вход сети измеренные параметры частиц, выход сети – тип частицы (или события). Обучаем сеть на множестве событий, для которого известно, какие события являются сигналом, а какие – фоном.





Результат:

- **эффективность** (доля событий сигнала со значением сети выше порога)
- **ошибка** (доля событий фона со значением сети выше порога, т.е. ошибочно принятых за сигнал)

Программный пакет ROOT

Для проведения статистического анализа событий необходимо использовать специализированное программное обеспечение. Для настольных экспериментов может хватить MATLAB или Origin. В физике частиц de-facto стандарт – программный пакет ROOT.

- Бесплатный
- Открытый код
- Кросс-платформенный
- Может использоваться и как приложение, и как библиотека
- Огромное количество инструментов:
  - гистограммы, функции, статистическая обработка
  - хранение и обработка очень больших объемов данных
  - научная графика
  - математическая библиотека
  - многомерный анализ данных (например, нейронные сети)
  - интеграция с Python, Ruby, Mathematica

Определение эффективности и фона

После того, как отобрали интересующие нас события, необходимо определить, сколько из них являются фоном и какова эффективность отбора сигнала. В эффективность также входит эффективность триггера, эффективность реконструкции и т.п.

$$\sigma = \frac{N_{obs} - N_{bg}}{\varepsilon \cdot \int \mathcal{L} dt}$$

Два основных подхода (как правило, применяются оба):

- Оценка с помощью самих данных

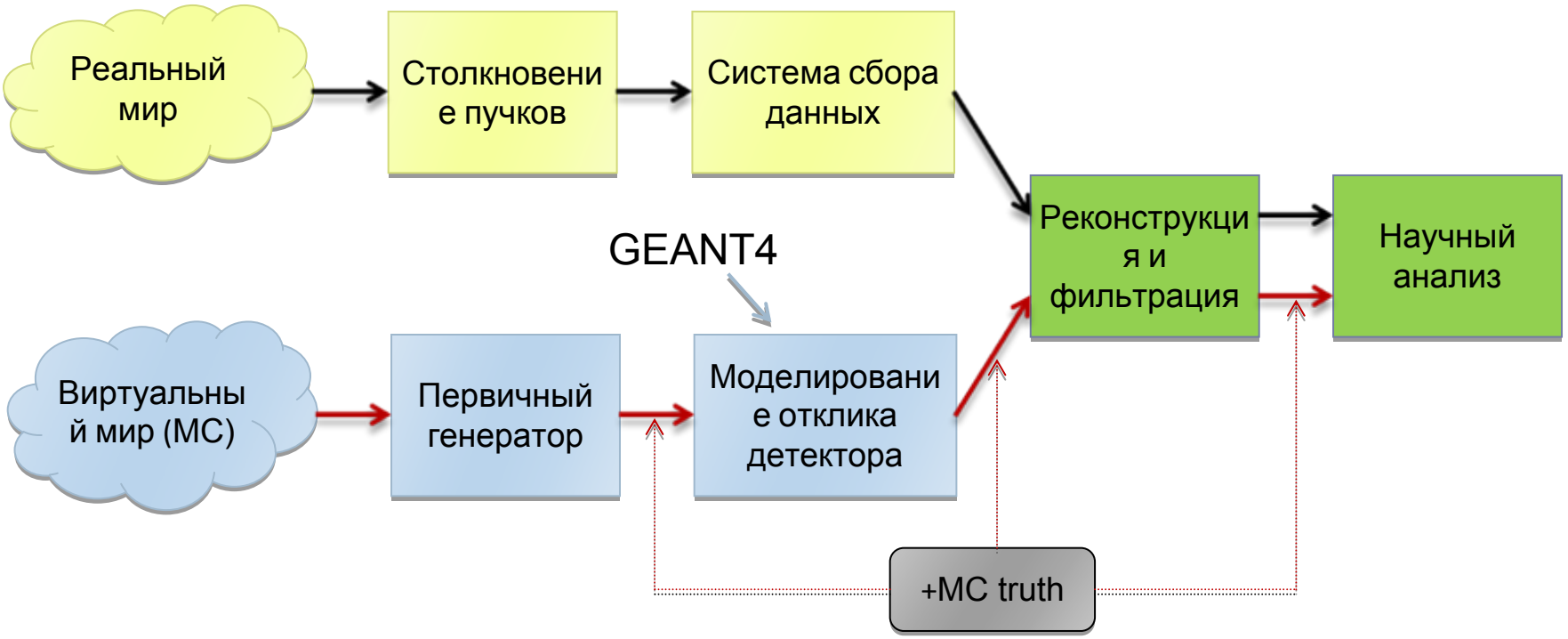
  Изучение распределения данных в различных областях значений параметров, при различных условиях набора статистики. Иногда один и тот же класс событий можно выделить по двум различным признакам, что позволяет изучать кросс-корреляции и получить оценку эффективностей.

- Моделирование

  Часто, единственно возможный способ

Моделирование

Повторяем весь эксперимент на компьютере – начинаем со сталкивающихся частиц, затем “проводим” все родившиеся частицы через детектор и моделируем отклик всех систем детектора. Затем «сырые» данные, полученные в моделировании, реконструируем так же, как и реальные данные.



Уровни моделирования

Уровни моделирования:

- Полное

  Исходные частицы → Энерговыделение → Отклик детектора → Сигналы в электронике → «Сырые» данные

  Занимает большое время, наилучшая точность. Детали в отдельной лекции.

- «Быстрое» или параметризованное

  Используются упрощенные модели, описывающие отклик детектора, эффективности и т.п.

  Значительно быстрее, можно смоделировать большие объемы данных, но хуже точность

- Toy Monte Carlo

  Генерируются данные согласно заложенным распределениям вероятности

  Используется в основном для оценки статистических и систематических ошибок

На каждом этапе необходимо проверять «разумность» моделирования, сравнивая его предсказания с измеряемыми параметрами

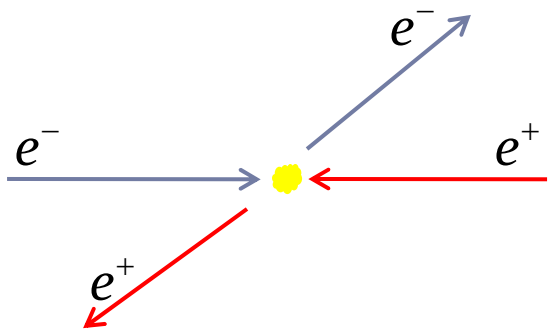
Нормировка результата: светимость

Для нормировки результата необходимо измерить светимость. Основная идея: отберем события, которые хорошо идентифицируются в детекторе и для которых сечение известно с хорошей точностью. Тогда светимость можно измерить, «перевернув» формулу для измерения сечения:

$$\int \mathcal{L} dt = \frac{N_{obs} - N_{bg}}{\varepsilon \cdot \sigma_{known}}$$

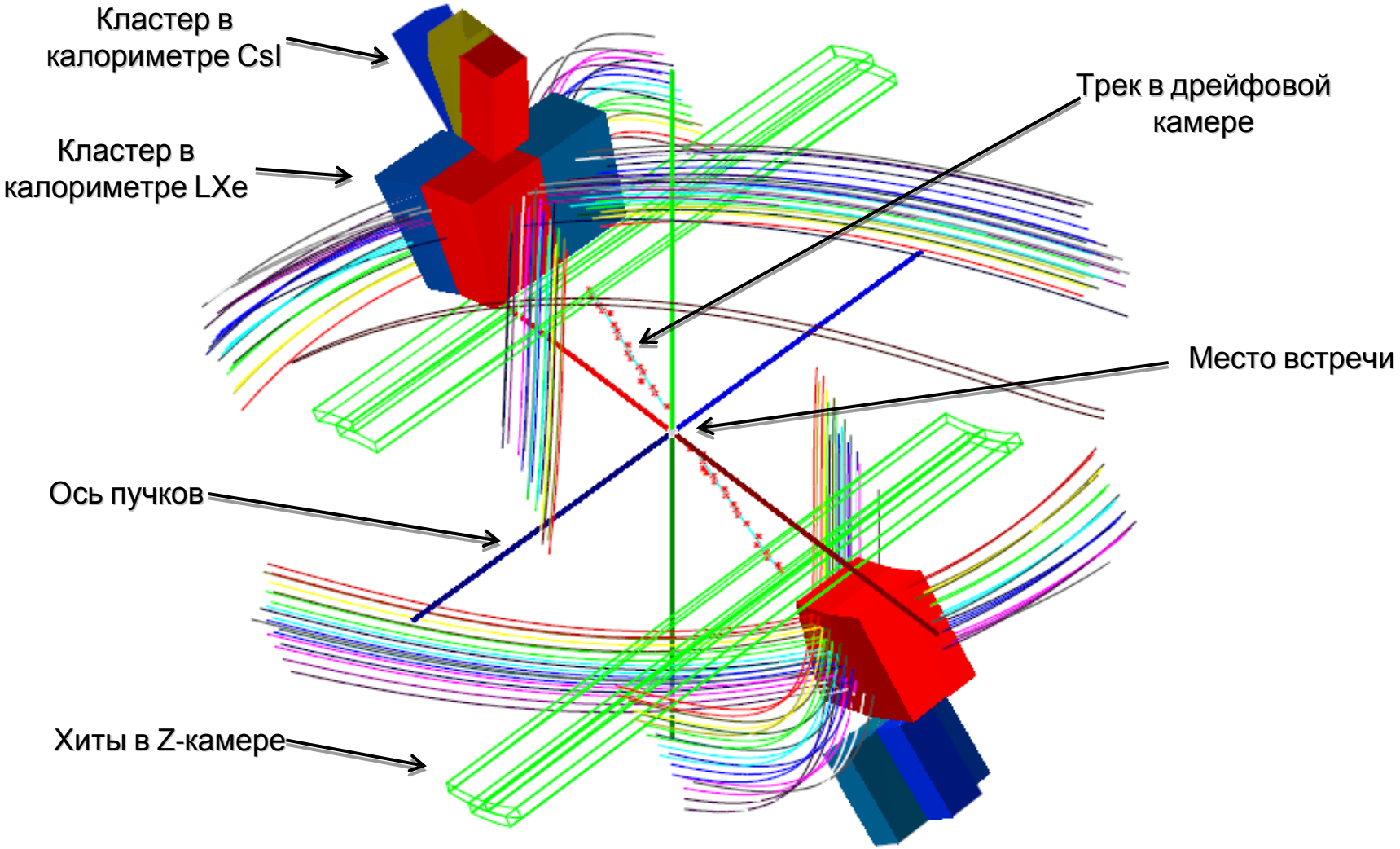
Очень удобным процессом для измерения светимости является Баба-рассеяние:

$$e^+e^- \to e^+e^-$$



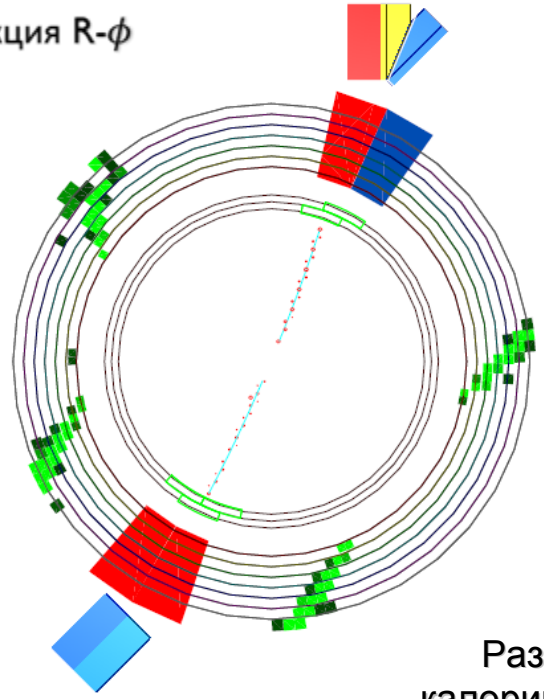
В качестве других мониторирующих процессов используются $e^+e^- \to \gamma\gamma$, тормозное излучение $e^+e^- \to e^+e^-\gamma$, $e^+e^- \to e^+e^-\gamma\gamma$,...

Пример: событие Баба-рассеяния в детекторе КМД-3

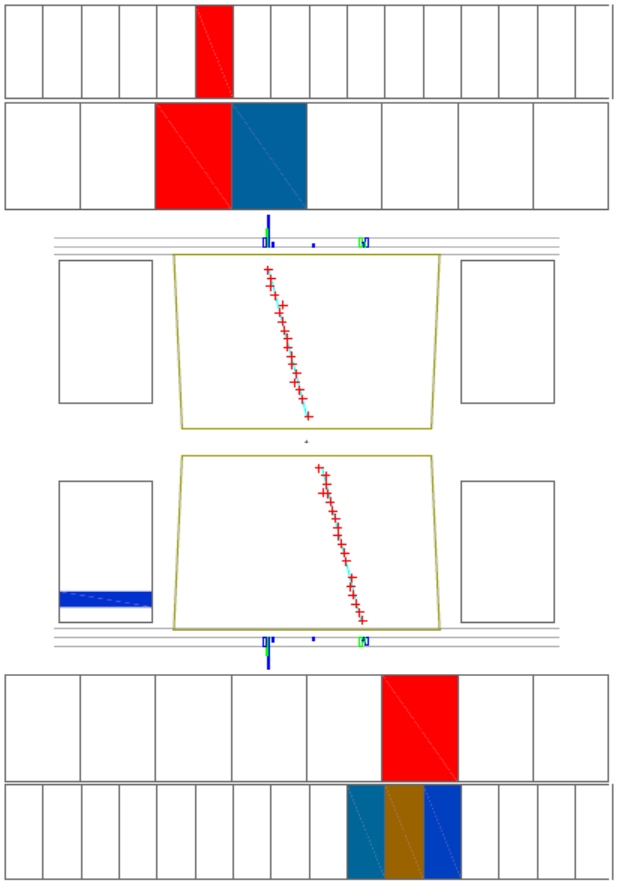


Пример: событие Баба-рассеяния в детекторе КМД-3

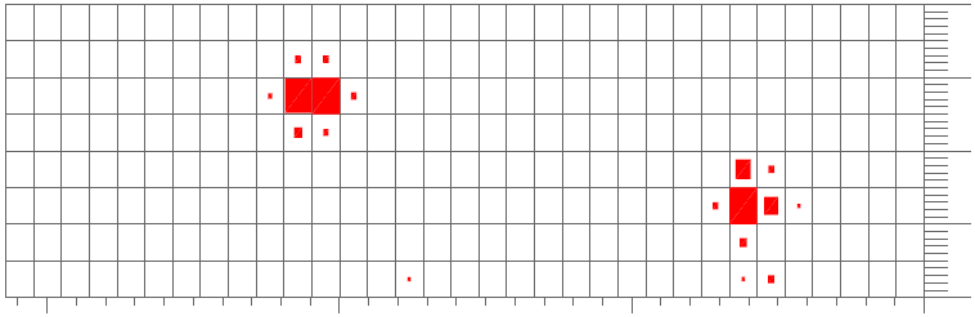
Проекция R-ϕ



Проекция R-Z



Развертка калориметра LXe



Получение результата

Итак, знаем все ингредиенты в правой части формулы – можем посчитать сечение.

$$\sigma = \frac{N_{obs} - N_{bg}}{\varepsilon \cdot \int \mathcal{L} dt}$$

Игра окончена? Нет!

- Предвзятость экспериментатора

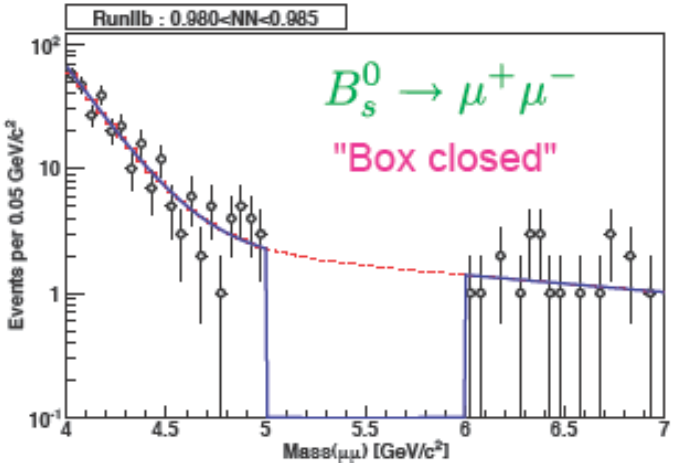
  Как правило, ученый старается проводить эксперименты непредвзято. Но очень часто на подсознательном уровне ученый выбирает алгоритм анализа под влиянием результатов эксперимента. Например: только если результат расходится с ожидаемым, мы начинаем исследовать влияние дополнительных факторов. Стратегия анализа должна быть построена так, чтобы минимизировать подобные ошибки. Общая идея: не позволять себе знать промежуточные результаты до тех пор, пока не будет выработан алгоритм анализа.

- Систематические ошибки

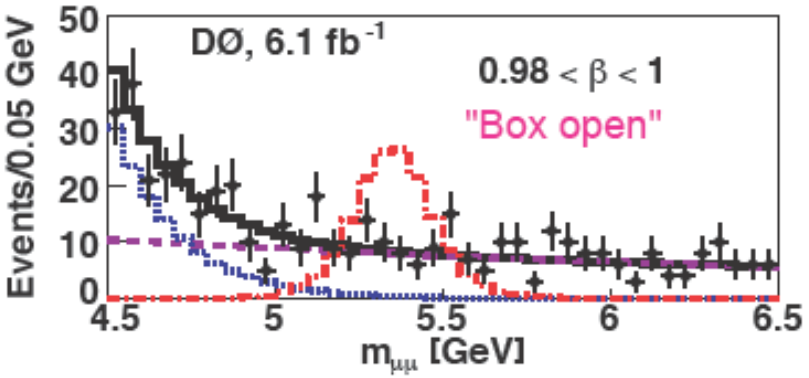
  Всевозможные внешние факторы могут приводить к искажению результатов. Стратегия анализа должна включать подробное изучение влияния на результат всевозможных факторов.

Пример: непредвзятый анализ

Пример: поиск редкого распада $B_S^0 \to \mu^+\mu^-$ на детекторе D0 (arXiv:1006.3469v2)
«Слепой» анализ: число событий в сигнальной области (“box”) скрывается до тех пор, пока не будут окончательно выбраны критерии отбора и проведен анализ фона и систематических ошибок. Только в самом конце подсчитывается число событий в сигнальной области (“box opening”), после чего никакие параметры анализа не должны меняться.



Количество фона оценивается с помощью экстраполяции снаружи от сигнальной области.



До открытия сигнальной области нужно договориться,
•когда ее следует открывать,
•и какова будет процедура проверки результата.

Систематические ошибки

- Оценка влияния известных факторов:
  - калибровочных параметров
  - различных поправок
  - эффективностей
  - теоретической модели
  - …

  Общий подход: изменяем соответствующий параметр и проверяем изменение результатов обработки
- Оценка влияния неизвестных факторов:

  Включаем фантазию и пытаемся их найти! Проверяем зависимость результатов от критериев отбора, доли фона, версии программного обеспечения, времени суток, фазы луны,… Если что то обнаружилось, идентифицируем источник, после чего оцениваем его влияние на результат

“As we know, there are known knowns; there are things we know we know. We also know there are known unknowns; that is to say we know there are some things we do not know. But there are also unknown unknowns -- the ones we don't know we don't know."

Donald Rumsfeld

Пример: систематические ошибки при измерении массы t-кварка

e.g., top quark mass

Source of systematic uncertainty	Magnitude (GeV/c^2)
Residual JES	0.42
b-JES	0.60
Generator	0.19
ISR	0.72
FSR	0.76
b-tag E_T dependence	0.31
Background composition	0.21
PDF	0.12
Monte Carlo statistics	0.04
Lepton p_T scale factor	0.22
Multiple Interactions	0.05
Total	1.36

Possible Variation with E_T or (change by $\pm 1\sigma$)

How different from light quarks?

PYTHIA vs. HERWIG

Vary parameters in generator

Change by $\pm 1\sigma$ in estimated efficiency

Change backgrounds by estimated undertainties and vary model of W+jets

Divide sample

Shift lepton p_T by ±1%

Room for MC not to model properly

Зачастую, анализ систематических ошибок – самая сложная часть анализа данных.

Для того, чтобы оценить число в каждой строчке (влияние определенного фактора), необходимо повторить весь анализ много раз! Для этого очень важно уметь правильно автоматизировать всю цепочку анализа данных.