HUST

ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.

ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

Nhập môn
Khoa học dữ liệu
(IT4142)

PGS.TS Thân Quang Khoát & PGS.TS Phạm Văn Hải
Team lecturers

ONE LOVE. ONE FUTURE.

2

Contents

- Lecture 1: Tổng quan về Khoa học dữ liệu
- Lecture 2: Thu thập và tiền xử lý dữ liệu
- **Lecture 3: Làm sạch và tích hợp dữ liệu**
- Lecture 4: Phân tích và khám phá dữ liệu
- Lecture 5: Trực quan hoá dữ liệu
- Lecture 6: Trực quan hoá dữ liệu đa biến
- Lecture 7: Học máy
- Lecture 8: Phân tích dữ liệu lớn
- Lecture 9: Báo cáo tiến độ bài tập lớn và hướng dẫn
- Lecture 10+11: Phân tích một số kiểu dữ liệu
- Lecture 12: Đánh giá kết quả phân tích

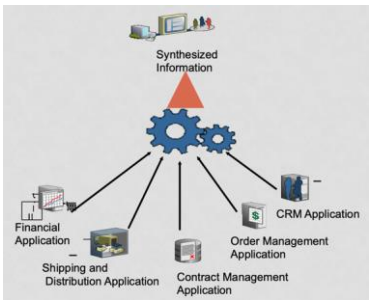
Nội dung

- Tích hợp dữ liệu - Data integration
  - Giới thiệu
  - Các hướng tiếp cận phổ biến
  - Giới thiệu Apache Nifi
  - Thực hành với Apache Nifi
- Tiền xử lý dữ liệu - Data preprocessing
  - Giới thiệu
  - Chất lượng dữ liệu - Data quality
  - Các bước tiền xử lý dữ liệu
  - Giới thiệu Openrefine
  - Tiền xử lý dữ liệu với python

Tích hợp dữ liệu

Đặt vấn đề

- Ngay cả đối với một tổ chức đơn, dữ liệu cũng thường được lưu trữ tại nhiều CSDL khác nhau và đến từ nhiều nguồn khác nhau
  - Định dạng dữ liệu khác nhau
  - Cùng mô hình dữ liệu nhưng khác cách đặt tên, quy chuẩn

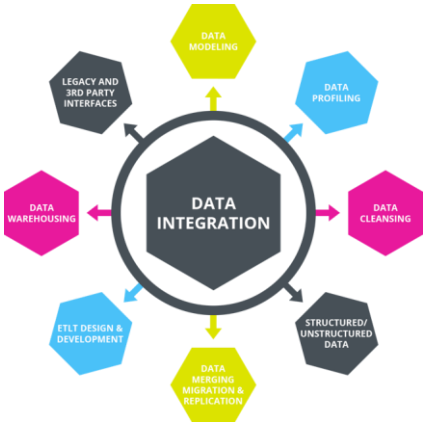


Đặt vấn đề (2)

Đặt vấn đề (3)

Tích hợp dữ liệu (THDL)

- Cung cấp truy cập đồng bộ tới một tập các nguồn dữ liệu tự trị và không đồng nhất
  - Truy vấn: Truy vấn trên các nguồn dữ liệu riêng biệt
  - Số lượng nguồn dữ liệu lớn
  - Tính không đồng nhất: các nguồn dữ liệu được phát triển độc lập, trên những hệ thống khác nhau: CSDL, hệ quản trị nội dung, file trong thư mục. Một số nguồn có cấu trúc, một số phi cấu trúc hoặc bán cấu trúc
  - Tự trị: các nguồn dữ liệu không nhất thiết thuộc về cùng một thực thể quản trị, mà có thể thuộc về các tổ chức con khác nhau.



Tại sao cần THDL

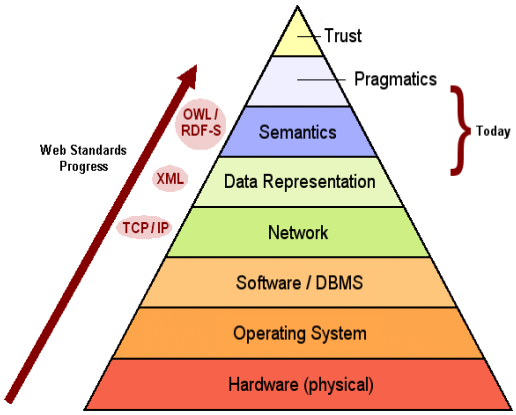
- Đơn giản hóa việc truy cập và tái sử dung thông tin thông qua một cổng, giao diện truy xuất thông tin duy nhất
- Dữ liệu từ các hệ thống khác nhau có thể được kết hợp để khai thác tối ưu hơn, hiệu quả hơn, đưa ra nhiều tri thức hơn
  - Tối ưu hóa quá trình ra quyết định
  - Tối ưu hóa trải nghiệm khách hang
  - Tăng khả năng cạch tranh, khai thông luồng công việc
  - Tăng năng xuất
  - Cho phép xây dựng mô hình dự báo, dự toán

Tại sao THDL là vấn đề khó?

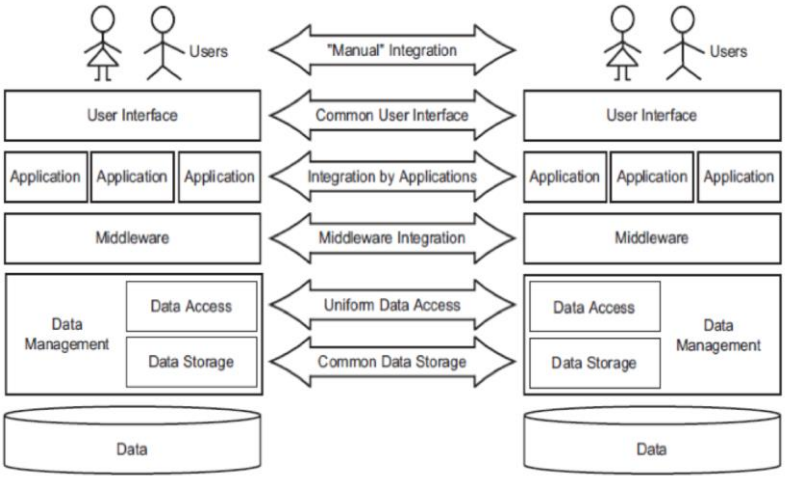
- Lý do hệ thống
  - khác nền, khác chuẩn
  - CSDL phân tán
  - khả năng xử lý truy vấn trên nhiều nguồn dữ liệu
- Lý do logic
  - dữ liệu được tổ chức logic trong các nguồn dữ liệu, thông qua lược đồ. Các lược đồ thường khác nhau
  - dữ liệu ở các nguồn khác nhau cũng được biểu diễn khác nhau
- Lý do xã hội và quản trị
  - có dễ dàng tiếp cận với các nguồn dữ liệu?
  - việc cho phép hệ thống tích hợp dữ liệu truy cập và sử dụng nguồn dữ liệu của tổ chức có thể thêm tải cho hệ thống của tổ chức.
  - các vấn đề an ninh, bảo mật

Các mức không đồng nhất

- Phần cứng và hệ điều hành
- Phần mềm quản trị dữ liệu
- Mô hình và lược đồ dữ liệu
- Middleware
- Giao diện người dung
- Các rằng buộc nghiệp vụ



Các mức trừu tượng hóa

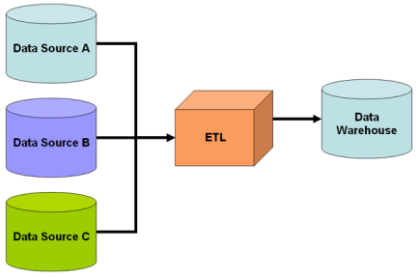


Các hướng tiếp cận phổ biến

- Ad-hoc programming
  - Tích hợp dữ liệu không theo chuẩn chung, giải pháp riêng biệt cụ thể cho từng nhu cầu và không có khả năng tổng quát hóa, tùy biến
- Kho dữ liệu - Data Warehouse
  - Dữ liệu từ các nguồn riêng biệt được nạp vào một CSDL vật lý (gọi là warehouse – kho dữ liệu), và trả lời truy vấn được thực hiện trên kho dữ liệu này
- Tích hợp ảo - Virtual integration
  - dữ liệu vẫn nằm ở các nguồn, và được truy cập khi cần thiết lúc xử lý truy vấn.

Kho dữ liệu (DW)

- Xây dựng một kho dữ liệu dùng chung
- Dữ liệu từ nhiều nguồn (OLTP) được trích rút, biến đổi, và đẩy về (ETL) kho dùng dung này
- Các phân tích OLAP có thể được thực thi trên kho dùng chung



Thu thập dữ liệu về kho dữ liệu

- Đưa dữ liệu về DW
  - Mã scipt linux shell, perl, python, ...
  - sqlldr + SQL
  - Viết mã chương trình Java, C#, C
  - Công cụ In-house built ETL tool
  - Công cụ Off-the shelf ETL tool
- Các vấn đề cần lưu tâm
  - Khả năng quản lý - Manageability
  - Khả năng bảo trì - Maintainability
  - Tính trong suốt - Transparency
  - Tính khả mở - Scalability
  - Khả năng linh hoạt - Flexibility
  - Tính phức tạp - Complexity
  - Khả năng kiểm toán - Auditing
  - Khả năng có thể thực thi lại - Job restartability
  - Khả năng kiểm thử - Testing

Tiến trình ETL

- 70-80% công việc của các dự án BI (DI hay DW) là thực thi tiến trình ETL
- ETL = Extract – Transform – Load
- Trích xuất - Extract
  - Thu thập dữ liệu từ các nguồn khác nhau hiệu quả nhất có thể
- Biến đổi - Transform
  - Thực thi các tính toán, biến đổi trên dữ liệu
- Đẩy về - Load
  - Đẩy dữ liệu sau khi biến đổi về kho dùng chung

Tầm quan trọng của ETL

- Mang lại giá trị cho dữ liệu
  - Loại bỏ các lỗi và chỉnh sửa lại dữ liệu đúng
  - Thống kê, đo độ tin cậy của dữ liệu
  - Nắm bắt các đặc trưng về luồng dữ liệu giao dịch đẩy vào kho dữ liệu
  - Chuyển đổi, chuẩn hóa dữ liệu từ nhiều nguồn khác nhau sao cho có thể phù hợp và sử dụng cùng nhau
  - Cấu trúc lại dữ liệu theo chuẩn yêu cầu của các công cụ BI
  - Cho phép phân tích và khai thác dữ liệu phục vụ BI

Tổng quan về thị trường ETL

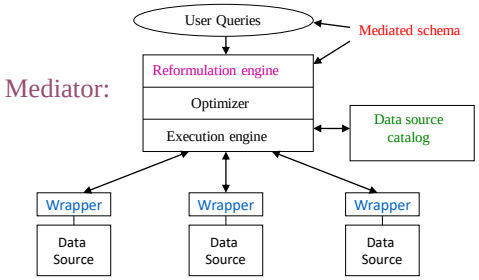
HUST
ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Các vấn đề với hướng tiếp cận DW

- Dữ liệu cần phải sạch, đưa về chuẩn chung
- Cần phải lưu trữ toàn bộ dữ liệu thêm một phiên bản về kho dùng chung, dẫn đến tốn kém chi phí
- Dữ liệu được cập nhật định kỳ
  - Các nguồn dữ liệu là độc lập - Khó khăn nếu nguồn dữ liệu đầu vào thay đổi cấu trúc và chuẩn kết nối
  - Chi phí ETL đắt đỏ để làm sạch và tổ chức lưu trữ

Tiếp cận hệ tích hợp ảo

- Dữ liệu vẫn nằm lại tại nguồn dữ liệu gốc
- Với mọi truy vấn trên lược đồ trung gian (mediated schema)
  - Tìm nguồn dữ liệu nơi có dữ liệu cần quan tâm
  - Truy vấn tại nguồn dữ liệu
  - Kết hợp kết quả truy vấn từ nhiều nguồn khác nhau nếu cần thiết



Các thách thức

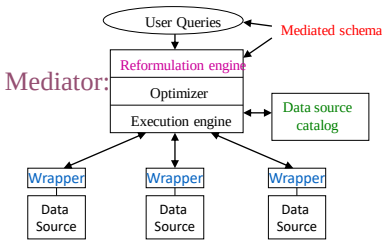
- Thiết kế ”được” một lược đồ dữ liệu trung gian chung
  - Nguồn dữ liệu gốc có thể tổ chức trên các lược đồ khác nhau, định dạng dữ liệu khác nhau
- Cần phải xây dựng cơ chế “biên dịch” truy vấn trên lược đồ trung gian về các truy vấn con trên các nguồn dữ liệu gốc
- Tối ưu hóa truy vấn
  - Khong có, hoặc ít, không cập nhật các thông tin về nguồn dữ liệu gốc
  - Chi phí xây dựng mô hình truy vấn cần lưu tâm tới chi phí truyền thông trên mạng
  - Lựa chọn nguồn dữ liệu gốc phù hợp với truy vấn
  - Phân rã truy vấn ban đầu thành các truy vấn con

Các thách thức (2)

- Thực thi truy vấn
  - Truyền thông trên mạng là thiếu tin cậy – dữ liệu gốc có thể không còn mới, không còn tồn tại hoặc bị trễ, bị mất mát
  - Truy vấn có thể được cached lại – làm như thế nào?
- Gửi truy vấn con tới đích
  - Cần cập nhật khả năng xử lý truy vấn và mô hình chi phí tại nguồn đích để tối ưu
- Nguồn dữ liệu không đầy đủ
  - Các nguồn dữ liệu có thể không đầy đủ, trùng lặp hoặc thậm chí mâu thuẫn lẫn nhau
  - Lựa chọn cách truy vấn từ nguồn nào? Theo thứ tự nào?

Các Wrappers

- Các nguồn cung cấp dữ liệu theo nhiều chuẩn khác nhau
- Wrappers là chương trình xây dựng riêng (custom-built programs) mà biến đổi dữ liệu gốc theo định dạng được chấp nhận bởi mediator



HTML

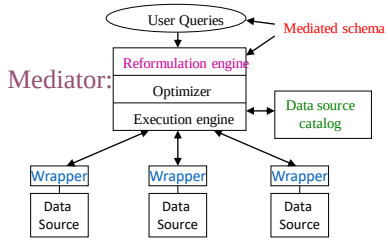
 Introduction to DB
 Phil Bernstein
 Eric Newcomer
Addison Wesley, 1999

XML

<book>
<title> Introduction to DB </title>
<author> Phil Bernstein </author>
<author> Eric Newcomer </author>
<publisher> Addison Wesley </publisher>
<year> 1999 </year>
</book>

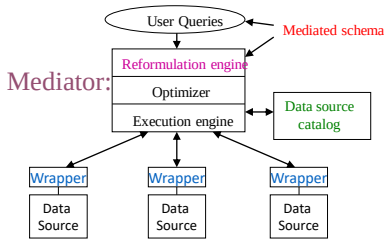
Wrappers (2)

- Có thể được cài đặt tại nguồn hoặc tại mediator
- Vấn đề bảo trì các wrappers
  - Wrapper cần phải thay đổi khi nguồn dữ liệu có thay đổi



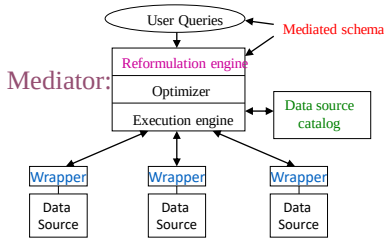
Danh sách nguồn dữ liệu - Data Source Catalog

- Lưu trữ thông tin metadata về các nguồn dữ liệu
  - Thông tin mô tả nội dung dữ liệu (books, new cars)
  - Khả năng xử lý của nguồn dữ liệu (can answer SQL queries)
  - Tính đầy đủ của nguồn dữ liệu (has all books)
  - Tính chất vật lý của nguồn và kết nối mạng tới nguồn
  - Thông tin thống kê về dữ liệu (like in an RDBMS)
  - Độ tin cậy của nguồn dữ liệu
  - Các nguồn sao lưu - Mirror sources
  - Tần xuất cập nhật của nguồn



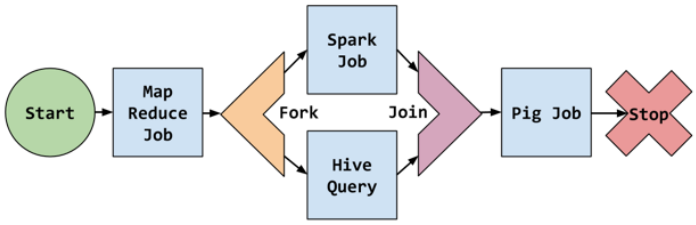
Lược đồ trung gian

- Người dùng truy vấn trên lược đồ trung gian - mediated schema
- Lược đồ dữ liệu ở các nguồn được gọi là lược đồ địa phương – local schema
- Phân rã truy vấn - Reformulation: Truy vấn trên lược đồ trung gian được viết lại thành các truy vấn con trên các lược đồ địa phương
- Vấn đề phân rã truy vấn?



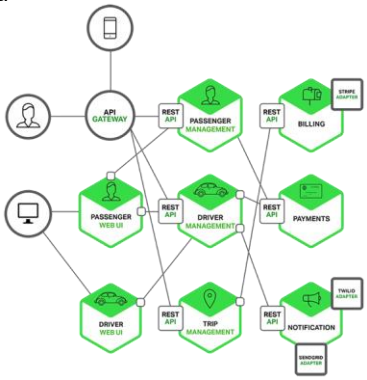
Hướng tiếp cận hệ luồng công việc

- Là hướng tiếp cận tích hợp mức ứng dụng - integration-by-application
- Luồng là chuỗi các bước, mỗi bước được thực hiện bởi 1 ứng dụng hay dịch vụ khác nhau hoặc bởi người dùng
- Hệ luồng công việc hỗ trợ mô hình hóa, thực thi, bảo trì luồng



Hướng tiếp cận Web service

- Thực thi theo chuẩn chung, hỗ trợ tương tác giữa các dịch vụ qua giao thức HTTP, định dạng dữ liệu dựa trên chuẩn XML
- Web service cho phép tương tác giữa nhiều chương trình mà có thể viết bằng nhiều ngôn ngữ khác nhau

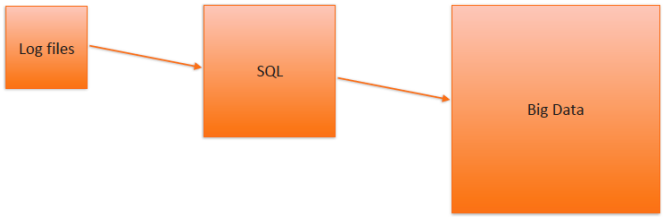


Apache Nifi

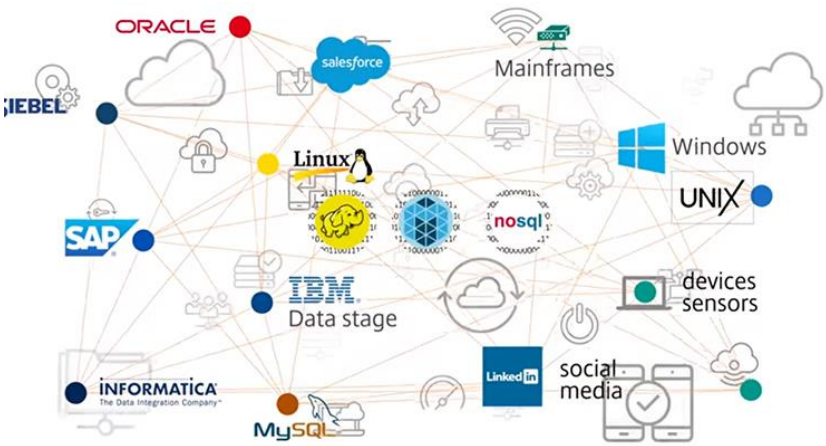
- Đặt vấn đề
- Các thuật ngữ trong Nifi
- Demo

Tích hợp dữ liệu giữa các thành phần khác nhau có phải là bài toán dễ dàng?

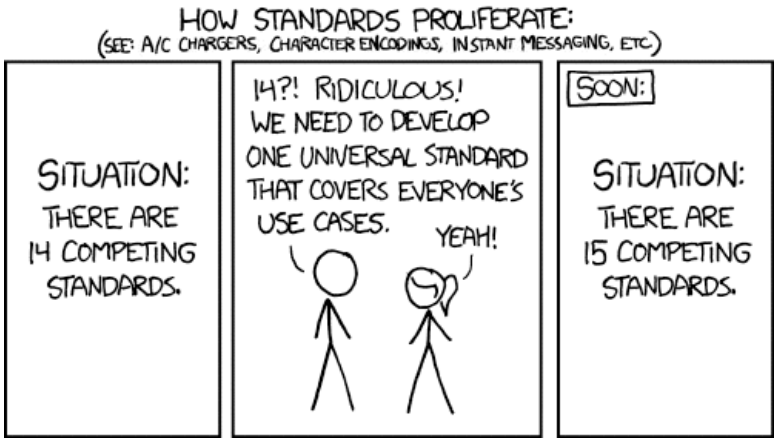
- Nếu thật sự đơn giản
  - Có thể dùng Bash/Ruby/Python
  - Hoặc SQL procedure
  - Vvv.



Tuy nhiên hướng tiếp cận này khó mở rộng và quản lý



Thực tế, tích hợp dữ liệu hiệu quả là một bài toán khó



Standards: http://xkcd.com/927/

Apache Nifi và Dataflow

- Dataflow là khái niệm được sử dụng trong Apache Nifi
  - Là luồng dữ liệu cần di chuyển từ A sang B
  - Dữ liệu có thể là bất cứ chuỗi bytebit nào
    - Logs
    - HTTP
    - XML
    - CSV
    - Images
    - Video

Dataflow hướng tới 3 nhóm thách thức

- Về dữ liệu
  - Standards
  - Formats
  - Protocols
  - Veracity
  - Validity
  - Schemas
  - Partitioning/Bundling
- Về hạ tầng - Infrastructure
  - “Exactly Once” Delivery
  - Ensuring Security
  - Overcoming
  - Security
  - Credential
  - Management
  - Network
- Về con người
  - Compliance
  - “That [person|team|group]”
  - Consumers Change
  - Requirements Change

NiFi dựa trên Flow Based Programming (FBP)

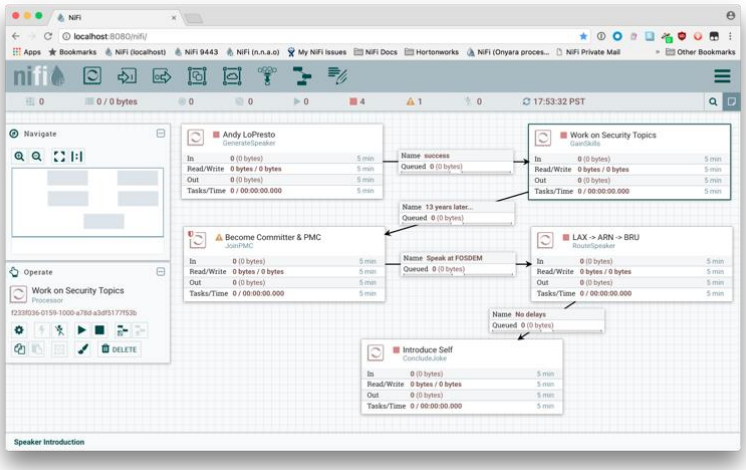
Thuật ngữ FBP	Thuật ngữ Nifi	Mô tả
Information Packet	FlowFile	Đối tượng dữ liệu vận chuyển trong hệ thống
Black Box	FlowFile Processor	Là các tiến trình thực thi các xử lý như định tuyến, biến đổi, hay làm trung gian giữa các hệ thống.
Bounded Buffer	Connection	Liên kết giữa các processors, có vai trò như các hàng đợi và cho phép nhiều tiến trình có thể tương tác với tốc độ xử lý khác nhau.
Scheduler	Flow Controller	Quản lý cách mà các tiến trình được kết nối và quản lý cấp phát các luồng (threads) cho các tiến trình sử dụng.
Subnet	Process Group	Một nhóm các tiến trình và kết nối giữa chúng, mà có thể nhận và gửi dữ liệu thông qua các cổng ports. Một process group cho phép tạo ra các thành phần mới trong hệ thống bằng việc lắp gép các thành phần hiện có.

HUST
ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

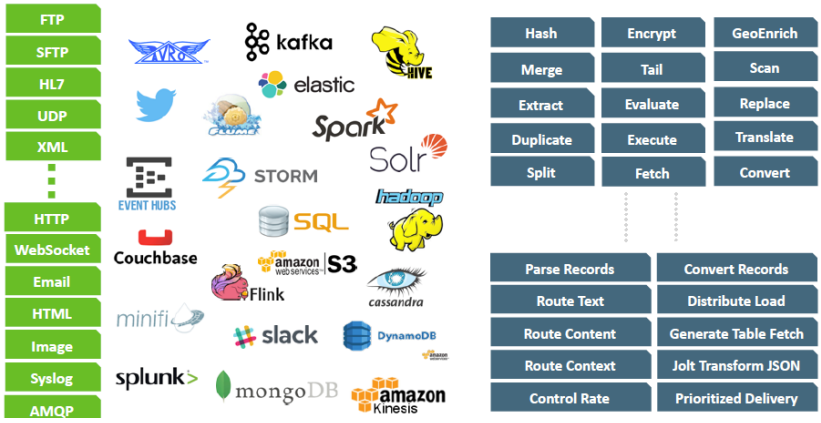
Các đặc trưng cốt lõi của Apach Nifi

- Đảm bảo sự vận chuyển dữ liệu
- Hàng đợi ưu tiên
- Quản lý QoS cho các luồng
  - Latency vs. throughput
  - Loss tolerance
- Data provenance
- Hỗ trợ mô hình push và pull
- Ghi và khôi phục từ các tệp tin nhật ký chi tiết các thay đổi
- Gao diện điều khiển trực quan
- Hỗ trợ các khuôn mẫu Flow templates
- Bảo mật đa nguyên, có thể mở rộng, tích hợp - Pluggable, multi-tenant security
- Thiết kế để mở rộng
- Hỗ trợ triển khai cụm Nifi dễ dàng

Giao diện đồ họa người dùng



Hệ sinh thái: 260+ Processors, 48 Controller Services

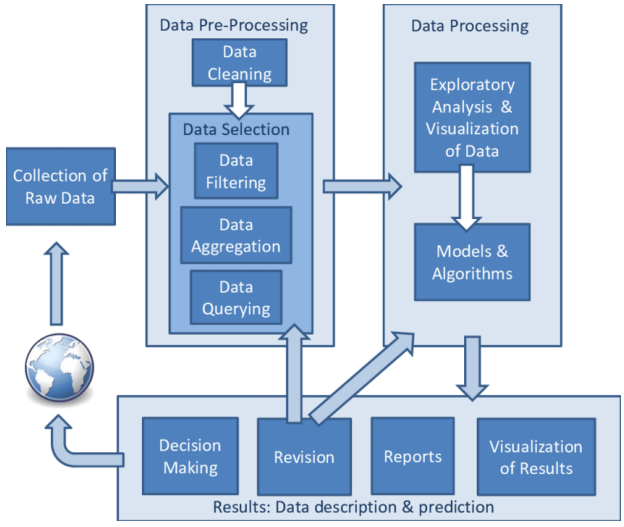


Nifi demo

- docker pull apache/nifi
- docker run --name nifi -p 8080:8080 -d apache/nifi:latest
- http://localhost:8080/nifi
- Hoặc trên windows 192.168.99.100:8080/nifi

Tiền xử lý dữ liệu

Quá trình khai thác dữ liệu

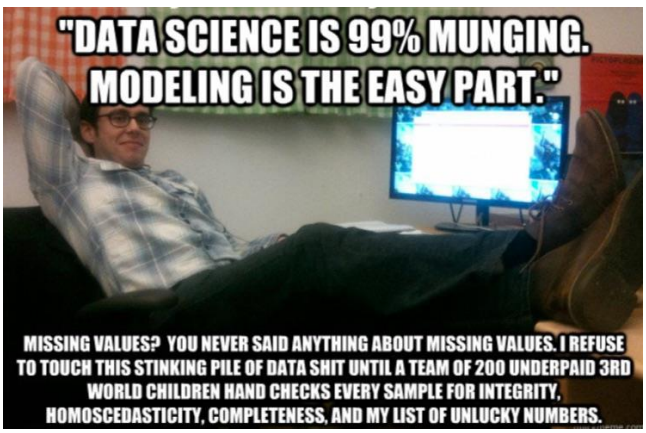


Tại sao cần tiền xử lý dữ liệu

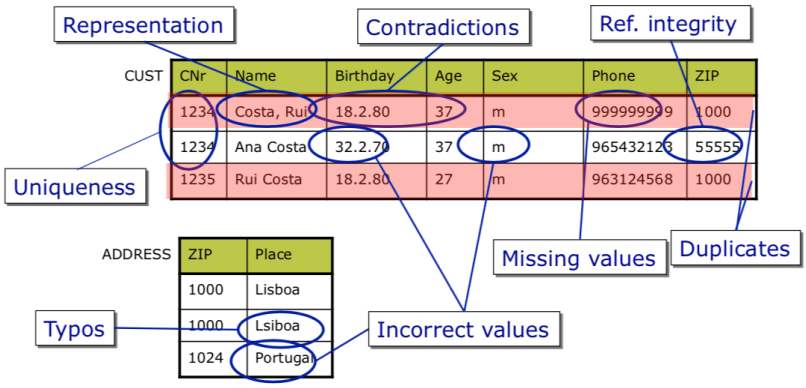
- Dữ liệu ngoài thực tế là không sạch "dirty"
- Không đầy đủ (e.g name = "")
  - Thuộc tính không có giá trị, thiếu một vài thuộc tính hoặc chỉ có giá trị thống kê không có giá trị chi tiết
  - Nguyên do có thể do sự không thống nhất khi thu thập dữ liệu và khi tiến hành phân tích dữ liệu
  - Lỗi con người, phần cứng, phần mềm
- Nhiễu (e.g. salary ='-10k')
  - Lỗi sai hoặc nhiều ngoại lệ
  - Lỗi thiết bị thu thập dữ liệu bị sai
  - Lỗi do người nhập liệu
  - Lỗi khi truyền dữ liệu
- Không nhất quán (e.g., Age="20" Birthday="02/02/1990")
  - Do dữ liệu thu thập từ nhiều nguồn khác nhau
  - Các rằng buộc phụ thuộc trong dữ liệu bị vi phạm
- Dữ liệu bị trùng lặp cũng cần được loại bỏ

Tiền xử lý dữ liệu thật sự rất tốn kém

- Trích xuất, làm sạch và biến đổi dữ liệu chiếm phần lớn khối lượng công việc khi xây dựng kho dữ liệu



Ví dụ về vấn đề chất lượng dữ liệu

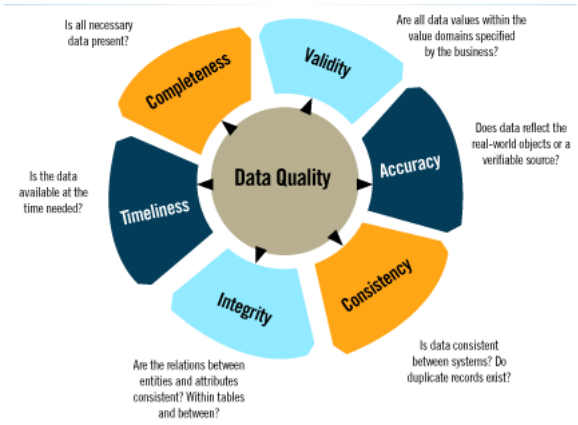


Dữ liệu không có chất lượng, rất khó để phân tích ra kết quả có ý nghĩa

- Các quyết định tốt cần được đúc rút từ dữ liệu có chất lượng
  - Vd., dữ liệu thiếu, mâu thuẫn lẫn nhau có thể đưa ra kết quả phân tích không đúng

Các thước đo về chất lượng dữ liệu

- "Even though quality cannot be defined, you know what it is." Robert Pirsig



Phân loại các vấn đề về chất lượng dữ liệu

- Mức giá trị đơn lẻ - Value-level
- Mức tập giá trị - Value-set (attribute/column) level
- Mức bản ghi - Record level
- Mức quan hệ - Relation level
- Mức giữa các quan hệ với nhau - Multiple relations level

Mức giá trị đơn - Value level

- Giá trị thiếu: Một trường thuộc tính cần phải có nhưng lại không có giá trị
  - Vd.:birthdate=‘’
- Vi phạm cú pháp: giá trị gi nhận không thỏa mãn luật về cú pháp định nghĩa cho giá trị trường thuộc tính
  - Vd.:zipcode=27655-175;syntacticalrule:xxxx-xxx
- Lỗi chính tả
  - Vd.:city=‘Lsboa’, thay vì giá trị đúng ‘Lisbon’
- Vi phạm miền xác định: giá trị không thuộc về tập các giá trị có thể có
  - Vd.:age=240; trong khi age:{0,120}

Mức tập giá trị và mức bản ghi

- Mức tập giá trị
  - Tồn tại các giá trị đồng nghĩa: thuộc tính có giá trị khác nhau nhưng có cùng ngữ nghĩa
    - Vd.: emprego = ‘futebolista’; emprego = ‘jogador futebol’
  - Tồn tại các từ đồng âm khác nghĩa
    - Vd: Cùng một tên nhưng thuộc về nhiều tác giả khác nhau
  - Tồn tại vi phạm tính đơn nhất:
    - Vd.: hai khách hàng có cùng ID
  - Tồn tại các vi phạm về rằng buộc toàn vẹn (Integrity contraint):
    - Vd.: Tổng các % thành phần vượt quá 100
- Mức bản ghi
  - Vi phạm rằng buộc toàn vẹn
    - Vd.: giá bán cuối cùng không bằng tổng giá + thuế VAT

Mức quan hệ

- Nhiều dạng biểu diễn khác nhau của dữ liệu: đây là vấn đề rất bình thường trong thực tế
  - Vd.: name = 'John Smith'; name = 'Smith, John'
- Vi phạm ràng buộc phụ thuộc hàm
  - Vd.: (2765-175, 'Estoril') và (2765-175, 'Oeiras')
- Sự tồn tại của các bản ghi **gần như** bị trùng lặp
  - Vd.: (1, André Fialho, 12634268) và (2, André Pereira Fialho, 12634268)!
- Vi phạm ràng buộc toàn vẹn
  - Vd.: Tổng lương của nhân viên lớn hơn tổng ngân quỹ lương

Mức đa quan hệ, đa bảng

- Nhiều dạng biểu diễn khác nhau của dữ liệu
  - Vd.: một bảng lưu trữ số đo theo đơn vị metter, một bảng theo đơn vị inch
- Tồn tại các đồng nghĩa
- Tồn tại các đồng âm khác nghĩa
- Sự khác nhau về đơn vị phân mức (granularity level):
  - Vd.: age:{0-30,31-60,>60};age:{0-25,26-40, 40-65, >65}
- Vi phạm ràng buộc tham chiếu
- Tồn tại các bản ghi gần như trùng lặp
- Vi phạm ràng buộc toàn vẹn

Các tác vụ chính của tiền xử lý dữ liệu

- Làm sạch dữ liệu - Data cleaning
  - Điền đầy các giá trị thiếu, làm mịn nhiễu, xác định và loại bỏ các ngoại lệ, phân giải sự không nhất quán trong dữ liệu
- Tích hợp dữ liệu - Data integration
  - Tích hợp nhiều cơ sở dữ liệu, nhiều nguồn dữ liệu khác nhau
- Chuyển đổi dữ liệu - Data transformation
  - Chuẩn hóa dữ liệu (quy chiếu dữ liệu theo phạm vi xác định)
  - Kết tập dữ liệu – aggregation
- Giảm nhẹ dữ liệu - Data reduction
  - Tìm một biểu diễn của dữ liệu nhỏ hơn về khối lượng nhưng vẫn đảm bảo kết quả phân tích
  - Rời rạc hóa dữ liệu - Data discretization: đặc biệt quan trọng với dữ liệu số
  - Kết tập dữ liệu, giảm chiều dữ liệu, nén dữ liệu, khái quát hóa dữ liệu

Làm sạch dữ liệu - Data cleaning

- Các tác vụ của làm sạch dữ liệu
  - Điền đầy các giá trị còn thiếu
  - Xác định ngoại lệ và làm mượt dữ liệu nhiễu
  - Sửa đúng cho dữ liệu không nhất quán

Xử lý dữ liệu thiếu

- Bỏ qua các bản ghi này: nếu số lượng bản ghi có dữ liệu thiếu không quá nhiều, có thể xem xét bỏ đi
- Xem xét từng giá trị bị thiếu và thêm vào: tốn kém và không khả thi?
- Sử dụng giá trị hằng toàn cục cho mọi giá trị thiếu: Vd., “unknown”, “NULL”?!
- Sử dụng giá trị trung vị median để điền các giá trị thiếu
- Sử dụng giá trị bình quân (mean) của từng phân lớp cho giá trị thiếu của bản ghi ứng với phân lớp đó
- Sử dụng giá trị có xác xuất cao nhất cho giá trị thiếu: dựa vào học suy diễn như hồi quy, công thức Bayesian, cây quyết định

Xử lý dữ liệu nhiễu

- Phương pháp tạo cột - Binning:
  - Sắp xếp dữ liệu và phân rã thành các cột có độ dầy bằng nhau
  - Làm mịn dữ liệu bằng cách sử dụng trung vị (median), số bình quân (mean) của các cột này, giá trị biên của các cột này
- Gom cụm - Clustering:
  - Nhận định và loại bỏ các ngoại lệ
- Hồi quy - Regression
  - Sử dụng các hàm hồi quy
- Bán tự động
  - Kết hợp mô hình và con người để xử lý dữ liệu nhiễu

Xử lý dữ liệu không nhất quán

- Xử lý bằng sức người, sử dụng các tài liệu tham chiếu bên ngoài (external references)
- Bán tự động sử dụng công cụ
  - Phát hiện các vi phạm ràng buộc phụ thuộc hàm và các ràng buộc khác của dữ liệu
  - Chỉnh sửa lại dữ liệu dư thừa

Phương pháp luận cho làm sạch dữ liệu

- Trích chọn các trường thuộc tính đơn mà có liên quan lẫn nhau
- Chuẩn hóa các trường bản ghi
- Chỉnh sửa lại dữ liệu ở mức giá trị đơn
- Chỉnh sửa lại dữ liệu ở mức tập giá trị và mức bản ghi
- Chỉnh sửa lại dữ liệu ở mức quan hệ
- Chỉnh sửa lại dữ liệu ở mức đa quan hệ
- Xem xét đến feedback của người sử dụng
  - Để giải quyết các vấn đề của dữ liệu mà không thể làm bằng các phương pháp chuẩn và tự động
- Tính hiệu quả của làm sạch dữ liệu và biến đổi dữ liệu cần được đánh giá cho cùng 1 tập dữ liệu

Tích hợp dữ liệu - Data integration

- Là bài toán kết hợp dữ liệu từ nhiều nguồn thành một hệ dữ liệu nhất quán, chặt chẽ
- Xây dựng lược đồ trung gian :
  - Vd. Chuẩn hóa các trường thuộc tính vd., A.cust-id = B.cust-#
- Định danh các thực thể bản ghi dữ liệu từ nhiều nguồn
  - Vd., Bill Clinton = William Clinton
- Phát hiện và giải quyết các xung đột dữ liệu khi tích hợp từ nhiều nguồn
  - Biểu diễn dữ liệu khác nhau, phạm vi dữ liệu khác nhau (scales)

Vấn đề dư thừa khi tích hợp dữ liệu

- Vấn đề dư thừa khi tích hợp dữ liệu từ nhiều nguồn
  - Định danh các đối tượng dữ liệu: Cùng thuộc tính, cùng đối tượng dữ liệu nhưng ghi nhận khác nhau trên các nguồn dữ liệu trước tích hợp
  - Dữ liệu phái sinh: Một thuộc tính có thể được phái sinh từ các thuộc khác trên các quan hệ khác. Vd.: doanh thu hàng năm được phái sinh từ doanh thu hàng tháng
- Các thuộc tính dư thừa có thể được phát hiện thông qua phân tích tương quan (correlation analysis)
- Sự cẩn trọng trong tích hợp dữ liệu từ nhiều nguồn có thể làm giảm hoặc tránh dư thừa, sự không nhất quán, từ đó tăng chất lượng và tốc độ khai thác dữ liệu

Biến đổi dữ liệu - Data transformation

- Làm mịn - Smoothing: khử nhiễu từ dữ liệu
- Kết tập - Aggregation: tổng kết, thống kê về dữ liệu
- Khái quát hóa - Generalization
- Chuẩn hóa - Normalization: co kéo dữ liệu theo phạm vi phù hợp
  - Chuẩn hóa min-max
  - Chuẩn hóa z-score
  - Chuẩn hóa theo thang thập phân - decimal scaling
- Kỹ nghệ xây dựng các đặc trưng - Attribute/feature engineering

Ví dụ về chuẩn hóa

- Chuẩn hóa Min-max: biến đổi, ánh xạ dữ liệu theo khoảng [NewMin, NewMax]

$$x_i' = \frac{x_i - OriginalMin}{OriginalMax - OriginalMin} \times (NewMax - NewMin) + NewMin$$

- Chuẩn hóa Z-score : (μ: mean, σ: standard deviation):

$$v' = \frac{v - \mu_A}{\sigma_A}$$

- Chuẩn hóa theo thang thập phân: Đảm bảo dữ liệu được kéo về khoảng 1 và −1
  - Với điều kiện n là số chữ số của giá trị lớn nhất

$$x_i' = \frac{x_i}{10^n}$$

Giảm nhẹ dữ liệu - Data reduction

- Tìm một biểu diễn của dữ liệu nhỏ hơn về khối lượng nhưng vẫn đảm. bảo kết quả phân tích
  - Giảm chiều dữ liệu - Dimensionality reduction
    - Lựa chọn đặc trưng
    - Trích rút đặc trưng (vd. Phân tích PCA)
  - Nén dữ liệu - Data Compression
    - Chuyển đổi dữ liệu văn bản thành dữ liệu số
    - Phân cụm dữ liệu
  - Rời rạc hóa dữ liệu - Discretization
    - Biến đổi dữ liệu liên tục về dữ liệu phân lớp

Hands-on openrefine

- https://guides.library.illinois.edu/openrefine

Tổng kết

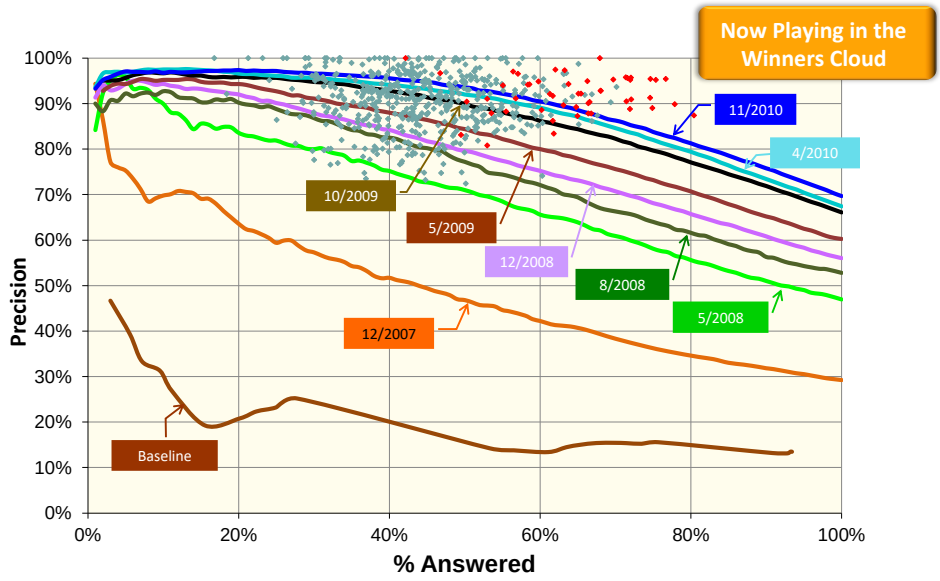
- Trong bài giảng này chúng ta đã tìm hiểu về
  - Tích hợp dữ liệu
  - Tiền xử lý dữ liệu
- Các công cụ có thể sử dụng
  - Apache Nifi
  - Openrefine
- Ghi nhớ
  - Đầu vào là rác, đầu ra cũng là rác

HUST

hust.edu.vn fb.com/dhbkhn

THANK YOU !

DeepQA: Incremental Progress in Precision and Confidence 6/2007-11/2010



Appendix

Correlation analysis (numerical data)

- Correlation coefficient (also called Pearson's product moment coefficient)

$$r_{A,B} = \frac{\sum (A-\bar{A})(B-\bar{B})}{(n-1)\sigma_A\sigma_B} = \frac{\sum (AB) - n\bar{A}\bar{B}}{(n-1)\sigma_A\sigma_B}$$

- where n is the number of tuples, and are the respective means of A and B, σA and σB are the respective standard deviation of A and B, and Σ(AB) is the sum of the AB cross-product.
- If rA,B > 0, A and B are positively correlated (A's values increase as B's). The higher, the stronger correlation.
- rA,B = 0: independent;
- rA,B < 0: negatively correlated

Correlation analysis (categorial data)

- X2 (chi-square) test

$$\chi^2 = \sum_i \sum_j \frac{(Observed - Expected)^2}{Expected}$$

- The larger the X2 value, the more likely the variables A, B are related (Observed is actual count of event (Ai,Bj))
- The cells that contribute the most to the X2 value are those whose actual count is very different from the expected count (based on totals)
- Correlation does not imply causality
  - # of hospitals and # of car-theft in a city are correlated
  - Both are causally linked to the third variable: population

Chi-square calculation: An example

- X2 (chi-square) calculation (numbers in parenthesis are expected counts calculated based on the data distribution in the two categories)
- It shows that like_science_fiction and play_chess are correlated in the group

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840} = 507.93$$

	Play Chess	Don't play chess	Sum (Row)
Like science fiction	250(90)	200(360)	450
Don't Like science fiction	50(210)	1000(840)	1050
Sum (Column)	300	1200	1500