

Databases @ York

Asking the Computer Questions

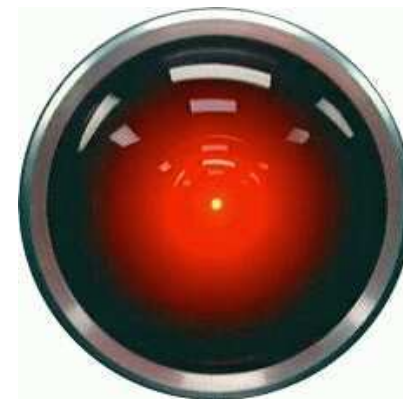
The Computer as Know-it-all

Parke Godfrey

4 January 2007

I. The Shipboard Computer

- The Enterprise's computer (23rd Century)
Star Trek, 1966–1969, Gene Roddenberry
- The HAL 9000
2001: A Space Odyssey, 1968, Arthur C. Clarke
 - *I became operational at the H.A.L. plant in Urbana, Illinois on the 12th of January 1992.*
 - *No 9000 computer has ever made a mistake or distorted information. We are all, by any practical definition of the words, foolproof and incapable of error.*



You can ask *it* questions.
And it often knows the answers!

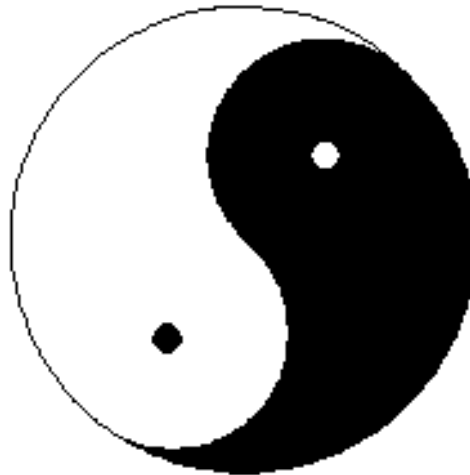
Questions

- Is such a machine possible?
- Would such a machine be necessarily *intelligent* (as HAL supposedly was)?
- What does it mean for a machine to *know* something?
- If it's possible, how *far* off are we?
- What technologies would be involved?
- So what *can* we do today?

What is a Computer?

- Something that *computes*.
- Something that stores (and manipulates) information.
(information processing / information systems)

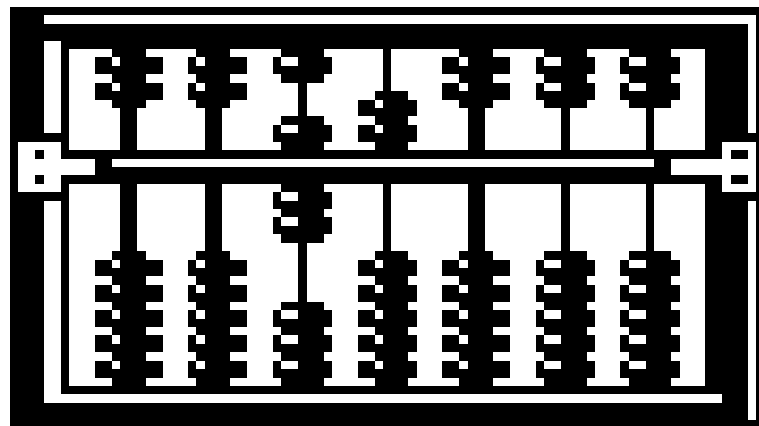
A computer is a synergy of these two aspects.



II. Automated Information:

A brief, incomplete history

- When did people start using computer-like things to store and manipulate information?
- How does the “computer”
 - store the information?
 - “reason” over the information?
- How do we retrieve information from it?

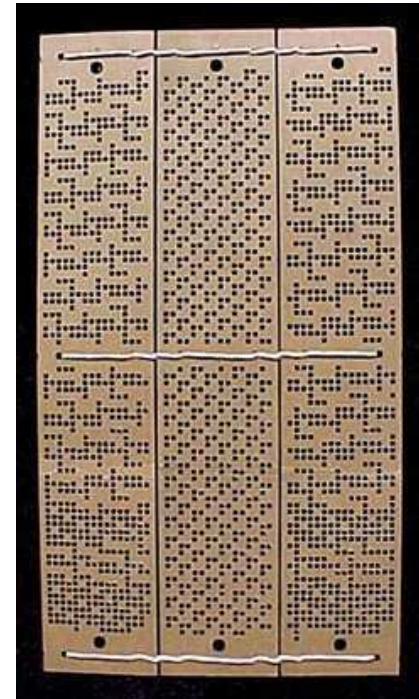


Joseph Jacquard (1752–1834)

The Jacquard Loom (1801–)

Punch cards arrive on the scene!

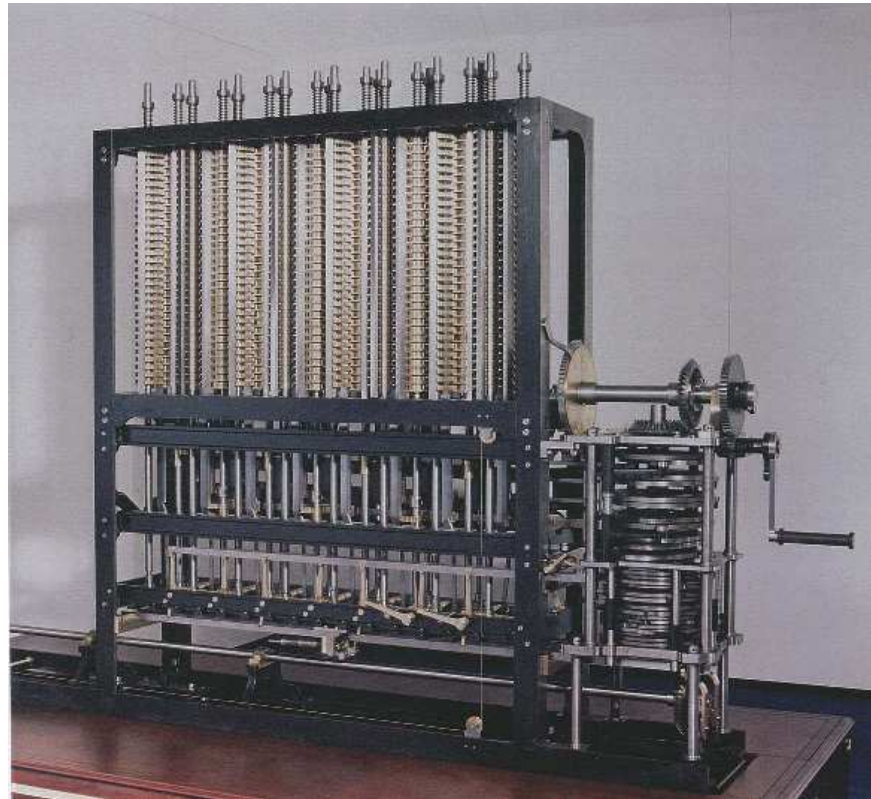
- Information: The pattern design for the fabric.
- Process (“computation”): Production of the fabric.



Charles Babbage (1791–1871)

The Analytical Engine (1834–)

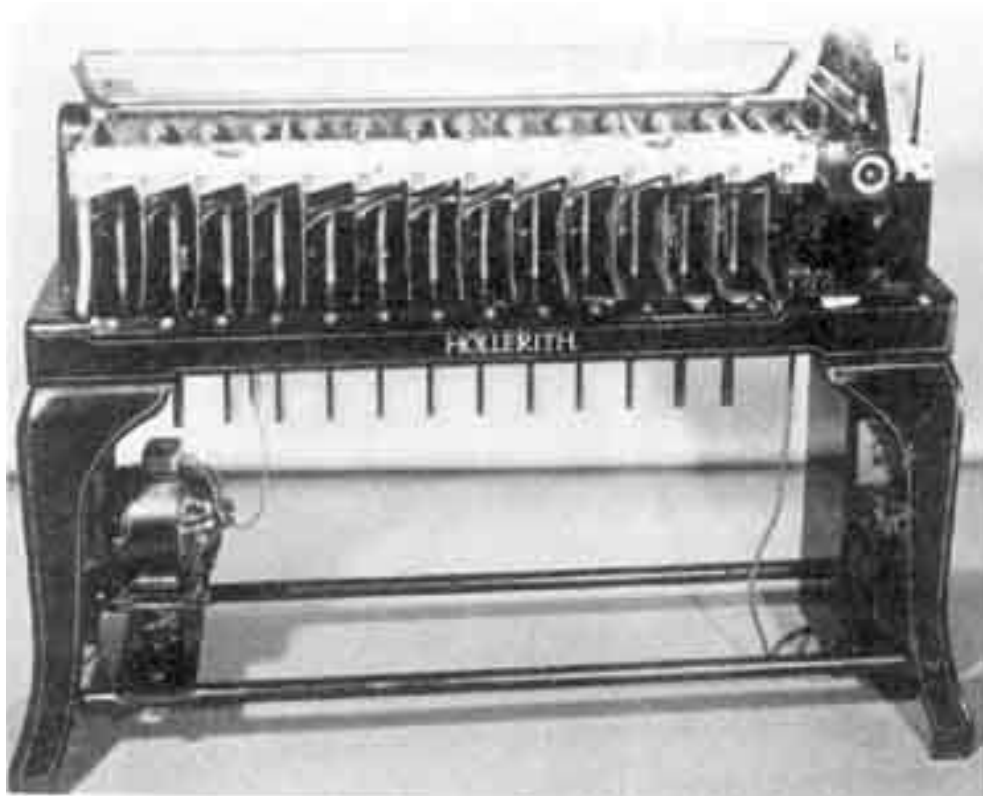
- Information: a *program* on punch-cards
- Process: a computation



Herman Hollerith (1860–1929)

The 1890's U.S. Census

- Information: U.S. residents
- Process: counting queries



- *Tabulating Machine Company (1896)*
- *Computer Tabulating Recording Company (1911)*
- *International Business Machines Corporation (IBM) (1924)*

Vannevar Bush (1890-1974)

As We May Think: Memex (& Hypertext) (1945)

- Information: *anything*
- Process: queries to retrieve specific information



- *The world has arrived at an age of cheap complex devices of great reliability, and something is bound to come of it.*
- *Consider a future device for individual use, which is a sort of mechanized private file and library. . . . It is an enlarged intimate supplement to his memory.*

E.F. Codd (1923–2003)

Relational Database Systems (1970, 1981–)

- Information: tabular data
- Process: queries to retrieve specific information



E.F. Codd (1923–2003)

Relational Database Systems (1970, 1981–)

- Information: tabular data
- Process: queries to retrieve specific information

name	school	age	gpa
Johansen, Sid	Purgatory High	17	93
Kripke, Sally	Limbo Collegiate	18	97
Zephon, George	Limbo Collegiate	21	73
Brynell, Alice	Flipflop High	18	86
Hong, Jessica	Purgatory High	19	79

E.F. Codd (1923–2003)

Relational Database Systems (1970, 1981–)

- Information: tabular data
- Process: queries to retrieve specific information

It's all about queries!

E.g., Who are the students applying to York who are interested in C.S. and have an GPA of 90 or more?

E.F. Codd (1923–2003)

Relational Database Systems (1970, 1981–)

- Information: tabular data
- Process: queries to retrieve specific information

Who are the students applying to York who are interested in C.S. and have an GPA of 90 or more?

← *applicant* (*S*, *york*), *student* (*S*, *N*, *A*, *G*),
 $G \geq 90$, *interest* (*S*, *cs*).

E.F. Codd (1923–2003)

Relational Database Systems (1970, 1981–)

- Information: tabular data
- Process: queries to retrieve specific information

Who are the students applying to York who are interested in C.S. and have an GPA of 90 or more?

$$\{ \langle N \rangle \mid \exists S, A, G.$$
$$\langle S, N, A, G \rangle \in \mathbf{Student} \wedge G \geq 90$$
$$\wedge \langle S, \text{york} \rangle \in \mathbf{Applicant}$$
$$\wedge \langle S, \text{cs} \rangle \in \mathbf{Interests} \}$$

E.F. Codd (1923–2003)

Relational Database Systems (1970, 1981–)

- Information: tabular data
- Process: queries to retrieve specific information

Who are the students applying to York who are interested in C.S. and have an GPA of 90 or more?

$$\pi_{\text{name}}\left(\left(\sigma_{\text{university}='York'}(\mathbf{applicant})\right)\right. \\ \quad \bowtie \sigma_{\text{GPA} \geq 90}(\mathbf{student}) \\ \quad \left. \bowtie \sigma_{\text{interest}='CS'}(\mathbf{interests})\right)$$

E.F. Codd (1923–2003)

Relational Database Systems (1970, 1981–)

- Information: tabular data
- Process: queries to retrieve specific information

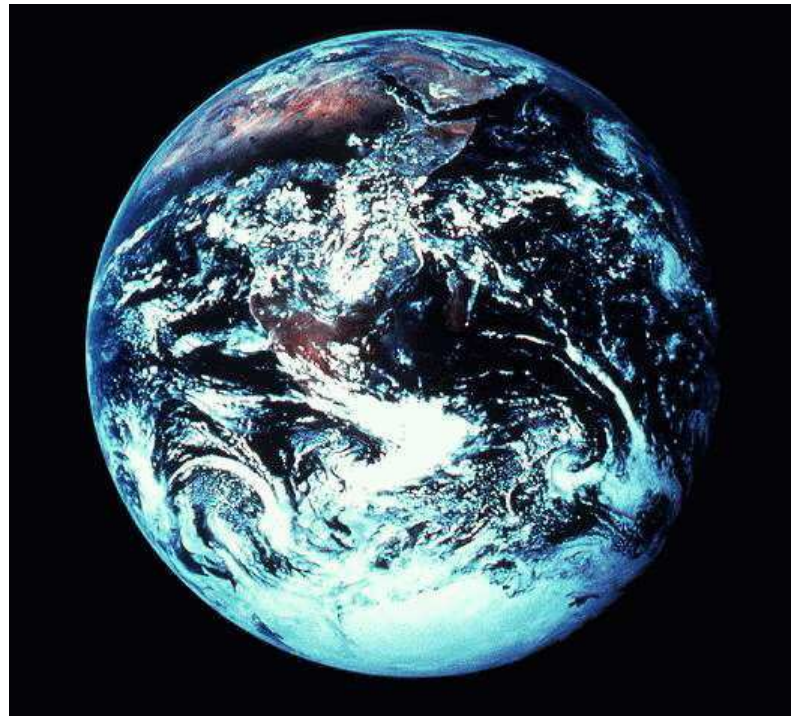
Who are the students applying to York who are interested in C.S. and have an GPA of 90 or more?

```
SELECT S.name
FROM Student S, Applicant A, Interests I
WHERE S.s# = A.s#
      AND A.university = 'York'
      AND S.gpa >= 90
      AND A.s# = I.s#
      AND I.interest = 'CS';
```


Sir Timothy Berners-Lee (1955–)

The World Wide Web (1989, 1993–)

- Information: web pages from all over
- Process: browse to find stuff



S. Brin (1973–) & L. Page (1975–) (Google (1998–))

- Information: web pages *from all over*
- Process: ask a keyword query



- Searching 8,058,044,651 web pages. . .
- *Treats the world as a database.*
- *But how to ask questions to the web?*

IV. Today

Information Appliances

Myriad of small, useful technologies.

- OnStar Navigation
Geographical Information Systems
- ABMs / ATMs
- Entertainment
 - TiVo, ReplayTV (Personal Video Recorders)
 - Movies (e.g., Pixar)
 - games, games, games

V. Database Technologies @ York

Areas

- Database systems (primarily *relational*)
- Information Retrieval
 - search engines
- Data Mining (Knowledge Discovery in Databases / KDD)
 - How to derive automatically interesting patterns (“knowledge”) from large databases.
- Information Systems at large
 - e-commerce systems
 - multi-tier architectures, focus on business solutions
 - web technologies
 - back-ended by database systems, data-driven, making information ubiquitous

Database Technologies @ York

Curriculum

- Database systems
 - CSE-3421: Databases “Programming”
database design, SQL
 - CSE-4411: Database Systems “OS”
database system architecture, query optimization
- Data Mining
 - CSE-4412: Data Mining
data warehousing & OLAP, algorithms for data mining, machine learning
- E-commerce
 - CSE-4413: E-commerce Systems
standards, multi-tiered systems, advanced application programming

Database Technologies @ York

Research Areas

- How to build a better system?
 - handle more data (scaling up)
 - handle more complex queries
- How to support more applications?
 - more expressive query languages
 - more & better tools
- How to integrate into more complex systems?
 - integrating multiple databases
 - integrating into larger software systems

Database Technologies @ York

Research Projects

- Skyline Queries

How to ask *best-match* queries in databases?

How to include *preferences* in queries?

```
select * from RestaurantRating
      skyline of Service max, Food max,
              Decor max, Price min
```

- Query Optimization

How to compute the answers to a query faster?

- by sampling
- by rewriting the query using logical transformations

Database Technologies @ York

Research Projects [cont.]

- Integration
How do you answer a query when the data is across many databases?
- Search Engines
How do you find what you are looking for on the web?
 - using database technology in the search engine
 - making the search queries more powerful

VI. Tomorrow

Ubiquitous Information

- cars that drive themselves, planes that fly themselves
To home, Fido.
- robots in the home
Do the dishes next.
- “universal translators”
e.g., You speak English into the phone, your friend on the other end in Japan hears Japanese.
Hello. Moshi moshi.

Tomorrow (cont.)

- Automated services and advisors
 - *Where in Akihabara can I find the latest model Asimo?*
 - *Verify the identity of the man standing at the front door.*
 - *What are the symptoms of an apendicitis? ...Help! Call 911!*
 - *Who proved Fermat's last theorem?*
 - *Explain to me again Andrew Wiles's proof of Fermat's last theorem.*

Tomorrow (cont.)

- Automated knowledge discovery
 - *I've noted that people habitually taking Vioxx have a risk of stroke and heart attack twice that of the population at large.*
 - *These stocks have each gone up by 10% or more in April in each of the past seven years.*
 - *I've devised a new fusion reactor that is 70% more efficient than the current Mr. Fusion on the market.*