

# BASI DI DATI E DBMS

VINCENZO CALABRÒ

# BASI DI DATI E SISTEMI INFORMATIVI

- UNA **BASE DI DATI** È UN INSIEME ORGANIZZATO DI DATI UTILIZZATI PER IL SUPPORTO ALLO SVOLGIMENTO DELLE ATTIVITÀ DI UN ENTE...
    - AZIENDA, UFFICIO, PERSONA...
  - UN **SISTEMA INFORMATIVO** È UNA COMBINAZIONE DI RISORSE UMANE, MATERIALI E DI PROCEDURE PER:
    - LA GESTIONE...
    - LA RACCOLTA...
    - L'ARCHIVIAZIONE...
    - L'ELABORAZIONE...
    - LO SCAMBIO...
- ...DELLE INFORMAZIONI NECESSARIE PER LE ATTIVITÀ DI UN ENTE.

# ESIGENZE MUTEVOLI NEL CORSO DEGLI ANNI

- INIZIO ANNI '60:
  - ELABORAZIONE RIPETITIVA DI GRANDI QUANTITÀ DI DATI:
    - FATTURE, PAGHE, STIPENDI..
- FINE ANNI '60:
  - SUPPORTO AI RESPONSABILI DELLE FUNZIONI AZIENDALI:
    - CONTABILITÀ GENERALE, CONTROLLO DI GESTIONE...
- INIZIO ANNI '70:
  - INTEGRAZIONE DATI COMUNI ALLE DIVERSE FUNZIONI:
    - **DBMS (DATA BASE MANAGEMENT SYSTEM)**

# ESIGENZE MUTEVOLI NEL CORSO DEGLI ANNI

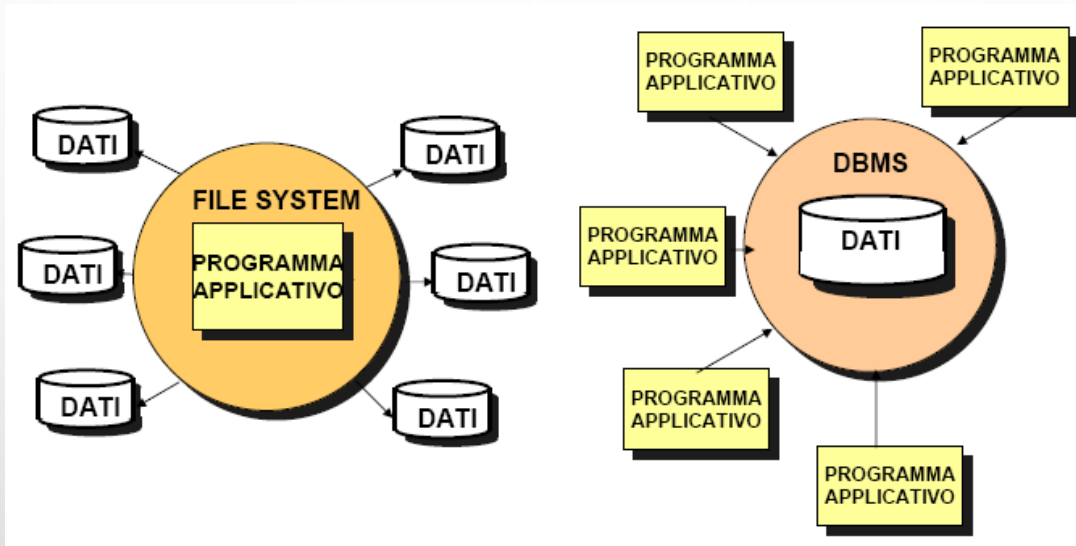
- INIZIO ANNI '80:
  - SINTESI DI INFORMAZIONI DAI DATI DELLA PRODUZIONE:
    - SUPPORTO ALLE DECISIONI, *DATAWAREHOUSE*...
- FINE ANNI '90:
  - INTERNET E WEB FAVORISCONO LO SVILUPPO DI PROTOCOLLI DI INTERAZIONE E COOPERAZIONE TRA SISTEMI DIVERSI:
    - COMMERCIO ELETTRONICO, *WEB SERVICES*...

# ARCHIVI TRADIZIONALI E BASI DI DATI

- Svantaggi:
  - Ridondanza
  - Inconsistenza
  - Privatezza
  - Condivisione
  - Accesso concorrente
- **Database Management System** è un sistema software progettato per la creazione e la manipolazione efficiente di basi di dati solitamente da parte di più utenti.



# BASE DI DATI E FILE SYSTEM



- I DBMS ESTENDONO LE FUNZIONALITÀ DEI FILE SYSTEM, FORNENDO PIÙ SERVIZI ED IN MANIERA INTEGRATA.
- NEI DBMS, C'È MAGGIORE **FLESSIBILITÀ**: SI PUÒ ACCEDERE CONTEMPORANEAMENTE A RECORD DIVERSI DI UNO STESSO FILE O ADDIRITTURA ALLO STESSO RECORD (IN LETTURA).
- I FILE SYSTEM PREVEDONO FORME DI CONDIVISIONE, PERMETTENDO ACCESSI CONTEMPORANEI IN LETTURA ED ESCLUSIVI IN SCRITTURA: SE È IN CORSO UN'OPERAZIONE DI SCRITTURA SU UN FILE, ALTRI NON POSSONO ACCEDERE AFFATTO AL FILE.

# DBMS

- UN SISTEMA PER LA GESTIONE DI DB GESTISCE BASI DI DATI:
  - **GRANDI** (OLTRE I GIGA, TERA BYTE),
  - **CONDIVISE** (APPLICAZIONI ED UTENTI DIVERSI),
  - **PERSISTENTI** (I DATI HANNO UN TEMPO DI VITA NON LIMITATO A QUELLO DELLE SINGOLE ESECUZIONI),
- ASSICURANDO LA LORO
  - **AFFIDABILITÀ** (CONSERVARE INTEGRO IL CONTENUTO NONOSTANTE I MALFUNZIONAMENTI),
  - **PRIVATEZZA** (AUTORIZZAZIONE).
- UN DBMS DEVE ESSERE **EFFICACE** ED **EFFICIENTE**.
  - UTILIZZO DI RISORSE (TEMPORALI E SPAZIALI) ACCETTABILI.

# IL MODELLO DEI DATI

- È UN INSIEME DI CONCETTI UTILIZZATI PER ORGANIZZARE I DATI DI INTERESSE E DESCRIVERNE LA STRUTTURA DI MODO CHE SIA COMPRENSIBILE AD UN ELABORATORE.
- OGNI MODELLO DATI FORNISCE MECCANISMI DI STRUTTURAZIONE (ANALOGHI AI COSTRUTTORI DI TIPO DEI LINGUAGGI DI PROGRAMMAZIONE) CHE PERMETTONO LA DEFINIZIONE DI NUOVI TIPI DI DATO.
- AD OGGI IL MODELLO PIÙ DIFFUSO NEI PRODOTTI DISPONIBILI SUL MERCATO È QUELLO *RELAZIONALE*.



# UN PO' DI STORIA...

- INIZIO ANNI '60: CHARLES BACHMAN (GENERAL ELECTRIC) PROGETTA IL PRIMO DBMS (INTEGRATED DATA STORE), BASATO SUL MODELLO RETICOLARE.
  - BACHMAN VINCERÀ IL PRIMO ACM TURING AWARD NEL 1973.
- FINE ANNI '60: L'IBM SVILUPPA L'INFORMATION MANAGEMENT SYSTEM (IMS), BASATO SUL MODELLO GERARCHICO E USATO ANCORA OGGI.
- 1970: EDGAR CODD (IBM) PROPONE IL MODELLO RELAZIONALE.
  - CODD VINCERÀ L'ACM TURING AWARD NEL 1981.
- ANNI '80: IL MODELLO RELAZIONALE PREVALE SUGLI ALTRI E I DBMS BASATI SU TALE MODELLO SI DIFFONDONO.
- IL LINGUAGGIO SQL VIENE STANDARDIZZATO COME LINGUAGGIO PER DBMS BASATI SUL MODELLO RELAZIONALE.
- ANNI '90: SULLA SPINTA DI INTENSE RICERCHE, I DBMS RELAZIONALI DIVENGONO SEMPRE PIÙ SOFISTICATI E DIFFUSI (DB2, ORACLE, INFORMIX).
- NEL 1999 JAMES GRAY VINCE L'ACM TURING AWARD PER IL SUO CONTRIBUTO ALLA GESTIONE DELLE TRANSAZIONI.
- RECENTEMENTE: I DBMS SI INTEGRANO CON IL CONTESTO GENERALE DELLO SVILUPPO DEL SOFTWARE E CON STRUMENTI WEB, E AMPIANO IL LORO SPETTRO DI UTILIZZAZIONE.

# BACHMAN ED IDS



Charles W. (Charlie) Bachman nasce l' 11 Dicembre 1924 a Manhattan, Kansas.

Nel 1973 riceve il premio Turing per... *“his outstanding contribution to database technology”*

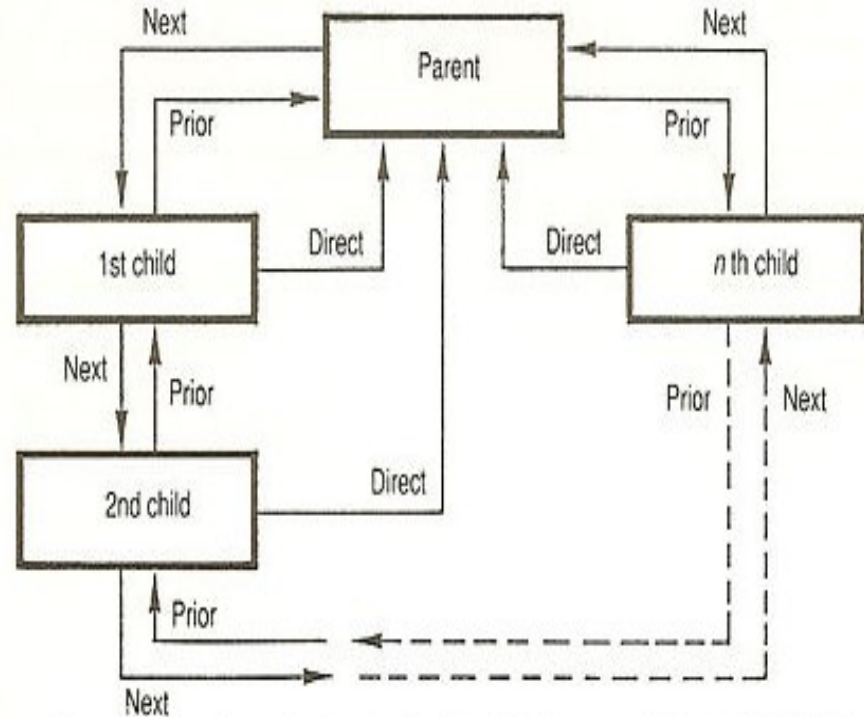
- LAVORA PER LA *GENERAL ELECTRIC* A PARTIRE DAL 1960. LÌ SVILUPPA **IDS (INTEGRATED DATA STORE)** UNO DEI PRIMI DBMS DELLA STORIA.
- IL SUO PRIMO INCARICO (1961-64) PER LA GE'S MANUFACTURING SERVICES (NEW YORK CITY) È IL PROGETTO E LA COSTRUZIONE DI UN *GENERIC MANUFACTURING INFORMATION AND CONTROL SYSTEM*: IL **MIACS**.

# MIACS E IDS

- IL SISTEMA MIACS CONTIENE LA PRIMA VERSIONE DELL'INTEGRATED DATA STORE (IDS) DATABASE MANAGEMENT SYSTEM, CHE RAPPRESENTA IL FULCRO PER:
  - GENERAL ELECTRICS IDS AND IDS II...
  - CULLINET'S IDMS...
  - ...ED ALTRI DBMS BASATI SUL MODELLO RETICOLARE DI BACHMAN'S.
- IDS È IL PRIMO DBMS BASATO SU DISCO UTILIZZATO IN PRODUZIONE. ESSO PRESENTA NUMEROSE INNOVAZIONI, RAGGRUPPATE IN UN UNICO PRODOTTO.
- VIENE COSTRUITO SU UN SISTEMA DI *MEMORIA VIRTUALE* APPLICATO SIA PER IL RECUPERO CHE LA MEMORIZZAZIONE DI DATI. SI AVVALE DI UN SISTEMA PER LA GESTIONE DI UN BUFFER DI *PAGE-TURNING* CHE CONSENTE UN ACCESSO PRESSOCHÉ Istantaneo AI DATI ACCEDUTI PIÙ DI RECENTE.
- IDS VIENE CLASSIFICATO COME DBMS CON MODELLO DEI DATI **RETICOLARE**.

# L'APPROCCIO CODASYL

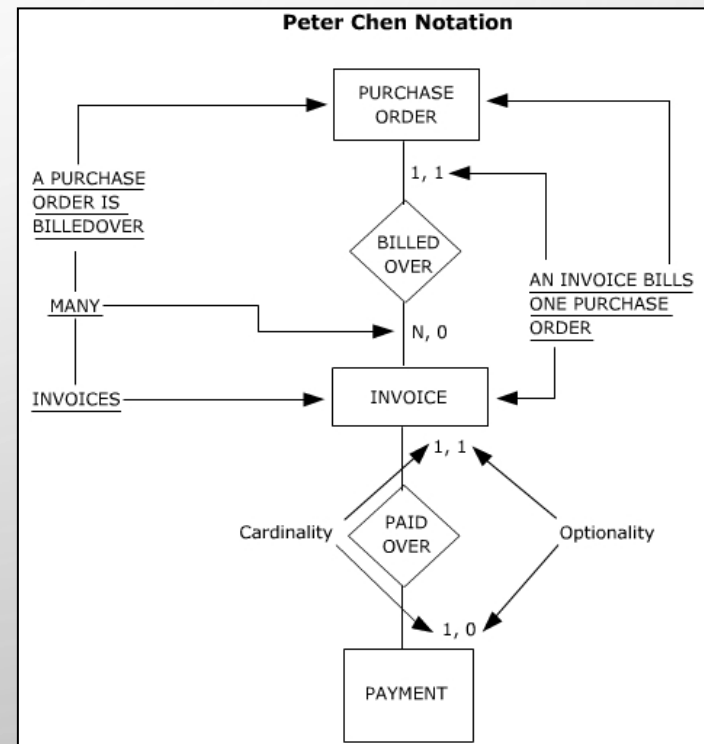
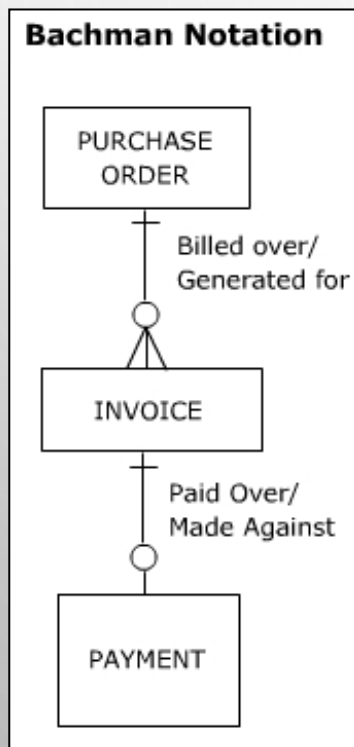
- IL GRUPPO DI LAVORO DI CUI BACHMAN FACEVA PARTE ERA INCARICATO DI SVILUPPARE ESTENSIONI IN LINGUAGGIO COBOL PER IL PROCESSAMENTO DI COLLEZIONI DI RECORD.
- TALE GRUPPO ERA DENOMINATO **CODASYL**. NEL 1967 ESSO CAMBIÒ LA DENOMINAZIONE IN **DATA BASE TASK GROUP** ED IL SUO PRIMO REPORT, DATATO GENNAIO 1968 ERA INTITOLATO "COBOL EXTENSIONS TO HANDLE DATA BASES".
- NELL'**APPROCCIO CODASYL** LE RELAZIONI TRA RECORDS VENGONO GESTITE UTILIZZANDO CATENE DI PUNTATORI.



A closed chain of records in a navigational database model (e.g. CODASYL), with *next pointers*, *prior pointers* and *direct pointers* provided by keys in the various records.

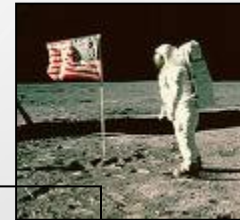
# ER DIAGRAMS

- BACHMAN SVILUPPÒ ANCHE DIAGRAMMI PER LA STRUTTURAZIONE DEI DATI (DIAGRAMMI ER, OVVERO ENTITA-RELAZIONE) COMUNEMENTE NOTI COME **DIAGRAMMI DI BACHMAN**.
- SI TRATTA DI RAPPRESENTAZIONI GRAFICHE FOCALIZZATE SUI DATI E SULLE LORO RELAZIONI RECIPROCHE.
- TALE MODELLO VIENE SPESSE UTILIZZATO NELLA PRIMA FASE DELLA PROGETTAZIONE DI UNA BASE DI DATI IN CUI È NECESSARIO TRADURRE LE INFORMAZIONI RISULTANTI DALL'ANALISI DI UN DETERMINATO DOMINIO IN UNO SCHEMA CONCETTUALE.



# IBM E IMS

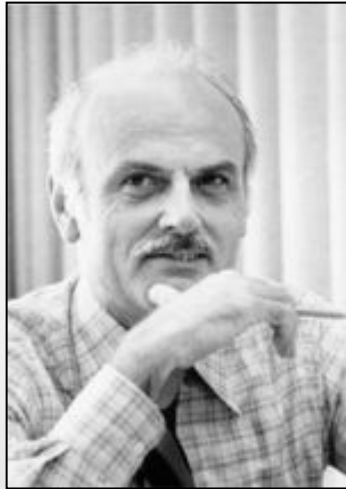
- NEL 1968, LA IBM SVILUPPÒ UN PROPRIO SISTEMA DBMS, CHIAMATO **IMS**. IMS ERA UNO SVILUPPO DI UN PROGRAMMA UTILIZZATO NELLE MISSIONI APOLLO SUI SISTEMI /360 E UTILIZZAVA UN SISTEMA SIMILE ALL'APPROCCIO DI BACHMAN, CON L'UNICA DIFFERENZA DI AVERE UN **SISTEMA GERARCHICO** ANZICHÉ A RETE.



## *Una curiosità ...*

*Nel 1969 il successo dell'atterraggio dell'Apollo 11 è supportato da numerosi System 360, Information Management System (IMS) 360 e da software IBM. Negli anni a venire, i computer IBM continueranno ad essere coinvolti nell'esplorazione spaziale.*

# EDGAR CODD



(Portland, 23 agosto 1923 - Williams Island, 18 aprile 2003) informatico britannico, fu tra i primi a contribuire alla teoria delle **basi di dati relazionali**.

- Nel 1970, mentre lavora per IBM, comincia a produrre documenti schematizzanti un nuovo approccio alla costruzione delle basi di dati, culminati nel *Modello Relazionale per Basi di Dati Condivise* (***Relational Model of Data for Large Shared Data Banks***).
- Invece di utilizzare delle righe collegate tra di loro attraverso un qualche tipo di struttura reticolare, propose di utilizzare una tabella di righe a lunghezza fissa.

# EDGAR CODD E IL MODELLO RELAZIONALE

- Questo sistema sarebbe stato molto inefficiente nell'archiviazione di dati *sparsi*, in cui la tabella avrebbe potuto avere diverse celle vuote.
- Tale errore di impostazione fu corretto dividendo i dati in **diverse tabelle**, in cui gli elementi opzionali venivano spostati, anziché sprecare spazio nella tabella principale.
- Nel modello relazionale, per ogni *record* viene definita una **chiave**, ovvero un *identificatore univoco* della tupla.



# EDGAR CODD E IL MODELLO RELAZIONALE

- CODD PROPOSE PER IL RECUPERO DEI DATI UN LINGUAGGIO DI INTERROGAZIONE APPOSITO.
- TALE LINGUAGGIO, PIÙ TARDI, SI È SVILUPPATO NELLA CODIFICA CHE OGGI È UNIVERSALMENTE ADOTTATA E CHE È IL MATTONI FONDAMENTALI DELLE BASI DI DATI: **SQL**.

## *Una curiosità ...*

*Quando IBM decise di implementare System R, sulla base delle idee promosse da Codd, lo sviluppo fu affidato ad un gruppo non particolarmente familiare con il modello relazionale.*

*Come risultato non fu utilizzato il linguaggio Alpha di Codd, ma ne fu creato uno non relazionale, denominato **SEQUEL**. Esso era tuttavia talmente superiore ai precedenti che fu copiato da Larry Ellison per il suo Oracle RDBMS.*

# CODD, ALTRI CONTRIBUTI

- UNA DELLE FORME NORMALI IMPIEGATE NELLA NORMALIZZAZIONE DI BASI DI DATI, LA **FORMA NORMALE DI BOYCE-CODD**, PRENDE PROPRIO IL NOME DA EDGAR CODD.
  - LA **NORMALIZZAZIONE** È UNA TECNICA UTILIZZATA PER IL DISEGNO DI DB RELAZIONALI ED HA LO SCOPO DI MINIMIZZARE LA DUPLICAZIONE DELLE INFORMAZIONI, PREVENENDO ALCUNI TIPI DI PROBLEMI STRUTTURALI E LOGICI (ANOMALIE DEI DATI).

## *Una curiosità ...*

*Quando il modello relazionale iniziò ad essere più di moda (inizio anni 80) Codd si prodigò in una strenua lotta per evitare che il termine “relazionale” fosse associato da db vendors a tecnologie in realtà obsolete. Egli pubblicò le **12 regole** per la definizione di un db relazionale. La sua campagna colpì anche SQL, che egli definì una non corretta implementazione delle sue teorie.*

# IBM System R

- LA IBM COMINCIÒ A IMPLEMENTARE LE IDEE DI CODD IN ALCUNI PROTOTIPI ALL'INIZIO DEGLI ANNI SETTANTA.
  - **SYSTEM R.**
- LA PRIMA VERSIONE FU REALIZZATA NEL 1974/75 CON UNO STRUMENTO MONOTABELLA.
- VERSIONI **MULTIUTENTE** FURONO REALIZZATE NEL 1978 E NEL 1979.
- NEGLI STESSI ANNI FU STANDARDIZZATO IL LINGUAGGIO **SQL.**
- LA SUPERIORITÀ DI QUESTO SISTEMA RISPETTO A CODASYL FU QUINDI EVIDENTE E LA IBM PASSÒ A SVILUPPARE UNA VERSIONE COMMERCIALE DI **SYSTEM R**, CHE PRESE IL NOME DI **SQL/DS** PRIMA E DI DATABASE 2 (**DB2**) POI.

# STRUCTURED QUERY LANGUAGE (SQL): EVOLUZIONE

- SQL È UN LINGUAGGIO STANDARD PER LA DEFINIZIONE E LA MANIPOLAZIONE DI DATABASE RELAZIONALI.
- STORIA DI SQL:
  - LA PRIMA VERSIONE VIENE DEFINITA NELL'AMBITO DEL PROGETTO SYSTEM R (IBM S. JOSE) NEL 1976.
  - IL NOME ORIGINARIO È **SEQUEL**.
  - VIENE PROGRESSIVAMENTE ADOTTATO DA TUTTI I SISTEMI COMMERCIALI.
  - VIENE STANDARDIZZATO DA ANSI (1986) E ISO.
- STANDARD SQL: PROGREDISCONO PER ESTENSIONI QUASI COMPATIBILI.
  - SQL-1: PRIMA VERSIONE ANSI DEL 1986.
  - SQL-89: ESTENSIONE DI SQL-1 CON **INTEGRITÀ REFERENZIALE**.
  - SQL-2 E SQL-92: SECONDA VERSIONE DEL 1992.
  - SQL –3: ULTIMA VERSIONE STANDARD RILASCIATA NEL 1999.

# EUEGENE WONG E MICHAEL STONEBRAKER

## INGRES

- IL LAVORO DI CODD VIENE PROSEGUITO PRESSO L'UNIVERSITÀ DI BERKELEY DA **EUEGENE WONG** E **MICHAEL STONEBRAKER**.
- IL LORO PROGETTO, CHIAMATO **INGRES** E FINANZIATO CON FONDI DESTINATI ALLA CREAZIONE DI UN DATABASE GEOGRAFICO, VEDE LA LUCE NEL 1973 E PRODUCE I PRIMI RISULTATI NEL 1974.
- INGRES È ASSAI SIMILE A "SYSTEM R" E PREVEDE UN LINGUAGGIO ALTERNATIVO A SQL, CHIAMATO **QUEL**.



### *Una curiosità ...*

*INGRES venne completato anche grazie all'opera di numerosi studenti che si prestarono quali programmatori (quasi 30 persone lavorarono al progetto).*

# DERIVAZIONI DI INGRES

- MOLTE DELLE PERSONE COINVOLTE NEL PROGETTO SI CONVINSERO DELLA FATTIBILITÀ COMMERCIALE DELLO STESSO E E FONDARONO IMPRESE PER ENTRARE NEL MERCATO CON QUESTO PRODOTTO.
- **SYBASE, INFORMIX, NONSTOP SQL** E ALLA FINE **INGRES** STESSA NACQUERO QUALI SPIN-OFF PER LA DIFFUSIONE DI INGRES ALL'INIZIO DEGLI ANNI OTTANTA.
- PERFINO **MICROSOFT SQL SERVER** È, PER CERTI VERSI, UNA DERIVAZIONE DI SYBASE E, QUINDI, DI INGRES.
- SOLAMENTE LA **ORACLE** DI **LARRY ELLISON** PARTÌ UTILIZZANDO UN APPROCCIO DIVERSO, BASATO SUL SYSTEM R DELLA IBM, E ALLA FINE PREVALSE SULLE ALTRE COMPAGNIE CON IL SUO PRODOTTO, LANCIATO NEL 1978.

# LAWRENCE J. ELLISON ED ORACLE



**Lawrence Joseph Ellison** (17 Agosto, 1944) è il co-fondatore e *Chief Executive Officer* della *Oracle Corporation*, una della più grandi aziende di fornitura di software database del mondo.

- Basandosi sulle pubblicazioni di Codd antecedenti al lancio di SEQUEL, Ellison fa suo l'approccio del dipendente IBM, raggiungendo il mercato prima di SQL/DS (Structured Query Language/Data System) con il suo prodotto, denominato *Oracle*.
- Nel 1979 viene messo in commercio *Oracle V.2*, di fatto il primo db relazionale al mondo. La Relational Software Incorporated (RSI), titolare del prodotto, diviene nel 1983 *Oracle Corporation*.

# LA STORIA DI ORACLE

- NEL 1985 SONO OLTRE 1000 I SITI CHE UTILIZZANO ORACLE. ESSO VIENE PORTATO SU 30 PIATTAFORME (HW+SW) DIVERSE,
  - OGGI SONO CIRCA 70...
- LA SUA DIFFUSIONE HA CONTRIBUITO ALLA DIFFUSIONE DEI MINICOMPUTER E DEI SERVER UNIX NEGLI ANNI '80.
- NEL 1986 VIENE INTRODOTTO L'APPROCCIO CLIENT/SERVER NEL DBMS ORACLE.
- NEL 1989 VIENE RILASCIATO ORACLE6.
- NEL 1991 VIENE INTRODOTTO PARALLEL SERVER, PER ARCHITETTURE A PARALLELISMO MASSICCIO.



# LA STORIA DI ORACLE

- NEL 1993 VIENE RILASCIATO ORACLE7 CON L'OTTIMIZZAZIONE BASATA SUL COSTO.
- NEL 1997 VIENE RILASCIATO ORACLE8 CHE INTRODUCE IL MODELLO RELAZIONALE AD OGGETTI ED IL SUPPORTO PER DATABASE DI GROSSE DIMENSIONI (VLDB).
- NEL 1999 VIENE RILASCIATO ORACLE8I CON L'INCLUSIONE DELLA JAVA VIRTUAL MACHINE.
- ATTUALMENTE SI È ALLA VERSIONE ORACLE10G, IN CUI LA G STA PER *GRID* E INDICA UN SUPPORTO AVANZATO AL *GRID COMPUTING*.

# CURIOSITÀ...

## *Una curiosità ...*

*Le vecchie versioni di Oracle, precedenti alla 10g, avevano uno schema di default chiamato **scott**. Una volta completato il processo di installazione, l'utente può collegarsi al database con l'username **scott** e password **tiger**. L'idea di chiamare uno schema scott è dovuta a Bruce Scott, uno dei primi impiegati alla Oracle. Sua l'idea anche della password, Tiger era infatti il nome del suo gatto.*

## *Qualche curiosità ...su Larry Ellison*

*Ellison è nato a New York City da Florence Spellman, una ragazza-madre di 19 anni che non poteva mantenerlo; dopo soli nove mesi ella chiese a sua zia Lillian Spellman Ellison e a suo marito Louis Ellison di adottarlo. I due accettarono e il piccolo Larry venne trasferito a Chicago.*

# EVOLUZIONE DEL MODELLO DEI DATI

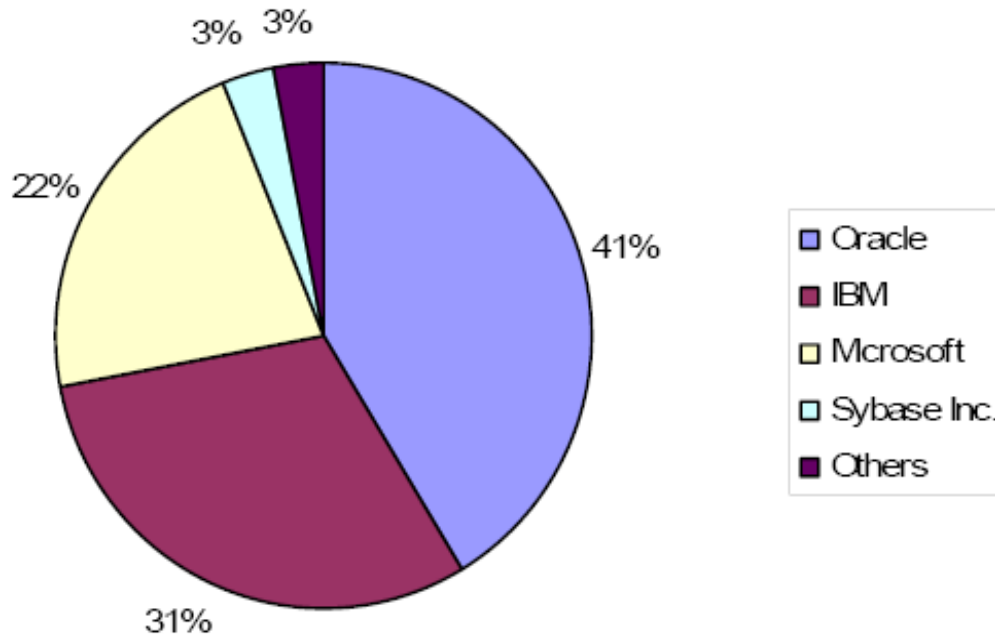
- PRIMA METÀ ANNI '90, **SISTEMI RELAZIONALI DISTRIBUITI**:
  - ARCHITETTURE CLIENT/SERVER.
  - SISTEMI A OGGETTI E RELAZIONALI A OGGETTI (GEMSTONE, ONTOS, OBJECTSTORE, O2, UNISQL).
    - RIPRENDE I CONCETTI DELL'OMONIMO PARADIGMA DI PROGRAMMAZIONE.
- SECONDA METÀ ANNI '90, **INTEGRAZIONE CON INTERNET**:
  - ARCHITETTURE SOA.
  - INTERAZIONE SU RETE.
  - WEB SERVICES.

# ODBMS

- I DBMS ORIENTATI AGLI OGGETTI ESTENDONO I LINGUAGGI DI PROGRAMMAZIONE CON PROPRIETÀ QUALI:
  - LA PERSISTENZA DEI DATI
  - IL CONTROLLO DELLA CONCORRENZA
  - IL DATA RECOVERY
  - LE QUERIES ASSOCIATIVE.
- I PRIMI ODBMS ERANO INTEGRATI CON LINGUAGGI SPECIFICI:
  - *SMALLTALK* (PER GEMSTONE)
  - *LISP* (PER GBASE)
  - *COP* (PER VBASE)
- PER GRAN PARTE DEGLI ANNI '90, **C++** HA DOMINATO LA SCENA DEI PRODOTTI COMMERCIALI.
- LO SVILUPPO DELL'APPROCCIO AD OGGETTI, SEBBENE PIÙ PERFORMANTE RISPETTO A QUELLO RELAZIONALE PER TASK SPECIFICI, È STATO OSTACOLATO DA:
  - **MANCANZA DI INTEROPERABILITÀ** CON NUMEROSI OLAP (ONLINE ANALYTICAL PROCESSING) TOOLS, NONCHÉ CON STANDARD DI BACKUP E RECOVERY.
  - L'APPROCCIO AD OGGETTI **MANCA** INOLTRE DI UNA **FORMALIZZAZIONE MATEMATICA** (PRESENTE INVECE NELL'APPROCCIO RELAZIONALE).

# QUOTE DI MERCATO *LICENSED*

Top Five Database Vendors

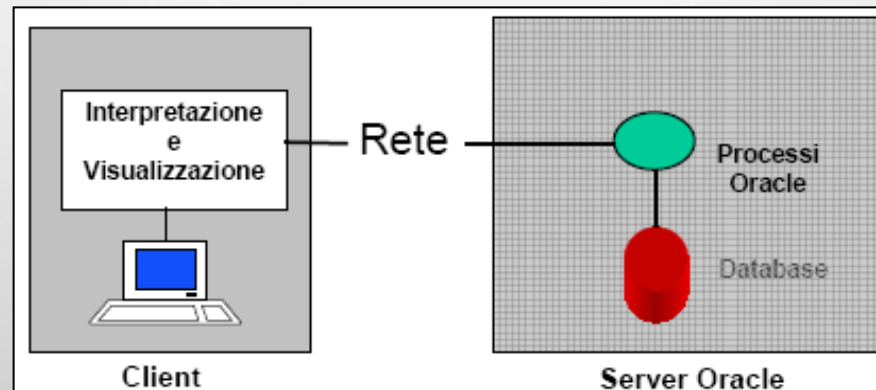


Il fatturato del mercato dei DBMS si aggira intorno agli **8 miliardi di US \$**, di cui l'**80%** imputabile ai **RDBMS**

...e cresce del 10% ogni anno...

# INTRO ALLE CARATTERISTICHE DI ORACLE

- **AMBIENTE CLIENT/SERVER** (ELABORAZIONE DISTRIBUITA)
  - ORACLE CONSENTE DI DISTRIBUIRE L'ELABORAZIONE FRA IL SERVER ED I PROGRAMMI APPLICATIVI. IL COMPUTER SU CUI È IN ESECUZIONE IL DBMS (SERVER) GESTISCE TUTTE LE ATTIVITÀ RELATIVE AI DATI, MENTRE LE WORKSTATION SU CUI È IN ESECUZIONE L'APPLICATIVO SI CONCENTRANO SULL'INTERPRETAZIONE E LA VISUALIZZAZIONE DEI DATI.
  - IL CARICO DI ELABORAZIONE VIENE DISTRIBUITO FRA CLIENT E SERVER (CHE POSSONO RISIEDERE SU HOST DIVERSI).



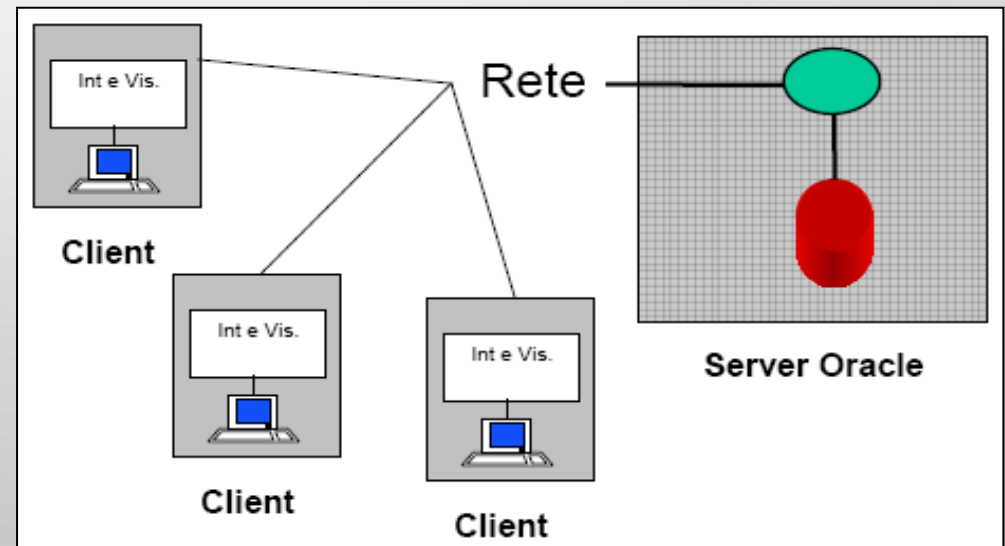
- **Gestione di database di grosse dimensioni**
  - Oracle fornisce supporto per database di grosse dimensioni (terabyte). Per ottenere un uso efficiente delle risorse hardware (costose!), Oracle consente il controllo dello spazio utilizzato.

# INTRO ALLE CARATTERISTICHE DI ORACLE

- **GESTIONE DI PIÙ UTENTI CHE ACCEDONO CONCORRENTEMENTE AL DB**
  - ORACLE FORNISCE IL SUPPORTO PER LA MULTIUTENZA.
  - PIÙ UTENTI, IN CONCORRENZA, ACCEDONO AL DATABASE E POSSONO ESEGUIRE APPLICATIVI DIVERSI CHE POSSONO ACCEDERE AGLI STESSI DATI.
  - L'ARCHITETTURA INTERNA DI ORACLE
    - MINIMIZZA LA COMPETIZIONE NELL'ACCESSO AI DATI
    - GARANTISCE LA PROPRIETÀ ACID.
- **ELEVATA DISPONIBILITÀ (AVAILABILITY)**
  - NORMALI ATTIVITÀ SISTEMISTICHE QUALI IL BACK-UP DEI DATI O CRASH PARZIALI DI SISTEMA NON INTERROMPONO L'UTILIZZO DELLA BASE DI DATI.

- **Connettività**

- Il software Oracle consente a differenti tipi di computer e di sistemi operativi di condividere informazioni per mezzo della rete.



# INTRO ALLE CARATTERISTICHE DI ORACLE

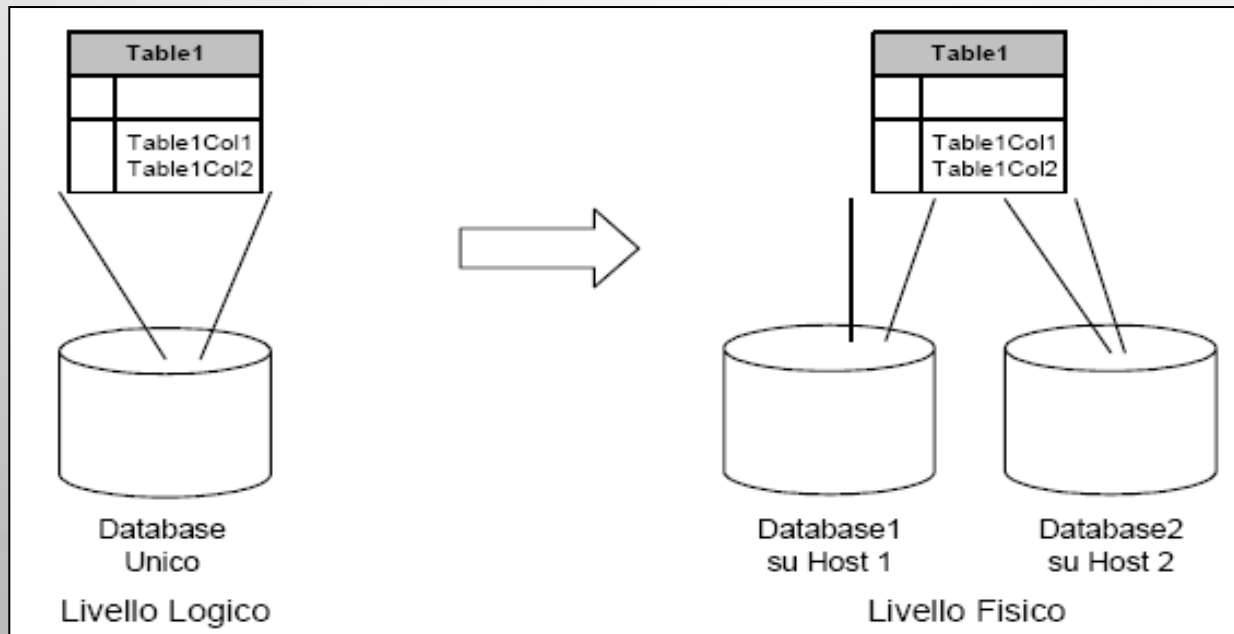
- **DISPONIBILITÀ CONTROLLATA**
  - PUÒ ESSERE CONTROLLATA SELETTIVAMENTE LA DISPONIBILITÀ DEI DATI, A LIVELLO DELL'INTERO DATABASE O AL SUO INTERNO. AD ESEMPIO L'AMMINISTRATORE PUÒ DISABILITARE L'UTILIZZO DI UNA SPECIFICA APPLICAZIONE IN MODO CHE I DATI DELL'APPLICAZIONE SIANO AGGIORNATI, SENZA INFLUENZARE ALTRE APPLICAZIONI.
- **PORTABILITÀ**
  - IL SOFTWARE ORACLE "GIRA" SOTTO DIFFERENTI SISTEMI OPERATIVI.
  - GLI APPLICATIVI SVILUPPATI SOTTO ORACLE POSSONO ESSERE PORTATI SOTTO QUALSIASI SISTEMA OPERATIVO CON POCO O NESSUN INTERVENTO. SISTEMI DISPONIBILI PER ORACLE: UNIX, LINUX, SOLARIS, WINXP/2000, WIN98/ME.



# INTRO ALLE CARATTERISTICHE DI ORACLE

- **DATABASE DISTRIBUITO**

- PER AMBIENTI DISTRIBUITI, ORACLE COMBINA I DATI LOCATI FISICAMENTE IN COMPUTER DIFFERENTI IN UN UNICO DATABASE LOGICO CHE PUÒ ESSERE ACCEDUTO DA TUTTI GLI UTENTI IN RETE.
- IL DATABASE DISTRIBUITO ORACLE HA LO STESSO GRADO DI TRASPARENZA AGLI UTENTI E CONSISTENZA DEI DATI DI UN DATABASE NON DISTRIBUITO.



A livello logico (livello dell'utente) la tabella è vista come appartenente ad un unico DB, anche se a livello fisico essa è memorizzata su due host distinti.

# STRUMENTI DI SVILUPPO IN ORACLE

- STRUMENTI PER L'ACCESSO E LA MANIPOLAZIONE DEI DATI CONTENUTI NELLA BASE DI DATI
  - **SQL** (STRUCTURED QUERY LANGUAGE)
    - È IL LINGUAGGIO DI INTERROGAZIONE STANDARD DEI DB RELAZIONALI
    - ADERENTE AGLI STANDARD SQL86-SQL92
    - TUTTA L'ATTIVITÀ DI GESTIONE/ACCESSO AL DB È EFFETTUATA IN SQL
  - **SQL+**
    - SQL+ È L'INTERFACCIA BASATA SULLA LINEA DI COMANDO AL DATABASE ORACLE.
    - È ESSENZIALMENTE UN *QUERY TOOL* INTERATTIVO CON LIMITATE CAPACITÀ DI SCRIPTING.
    - CON SQL+ SI PUÒ:
      - INVOCARE UNA QUERY E VISUALIZZARNE I RISULTATI
      - UTILIZZARE GLI STATEMENT DI DDL
      - MANDARE IN ESECUZIONE BLOCCHI PL/SQL
      - MANDARE IN ESECUZIONE CODICE SQL SU FILE
      - FORMATTARE L'OUTPUT PER OTTENERE DEI REPORT

# STRUMENTI DI SVILUPPO IN ORACLE

- **PL/SQL**
  - È L'ESTENSIONE PROCEDURALE DI SQL
  - È UTILIZZATO PER COSTRUIRE
    - STORED PROCEDURES
    - TRIGGERS
    - STRUTTURE DI CONTROLLO
    - GESTIONE DEGLI ERRORI
  - LE PROCEDURE PL/SQL POSSONO ESSERE COMPILATE E MEMORIZZATE NEL DATABASE.
  - POSSONO ESSERE ESEGUITE TRAMITE SQL+.
- **SUPPORTO A JAVA**
  - LA JAVA VIRTUAL MACHINE È INCLUSA A PARTIRE DA ORACLE8I
  - JAVA PUÒ ESSERE UTILIZZATO COME LINGUAGGIO PROCEDURALE PER LA SCRITTURA DI
    - STORED PROCEDURES
    - TRIGGERS
    - METODI

# DBMS: *VENDOR* E PRODOTTI

- ESISTONO SUL MERCATO TANTISSIMI PRODOTTI DIVERSI:
  - PER USO PERSONALE.
  - PER SERVER IN UNA RETE.
  - PER MAINFRAME.
- ALCUNI DEI PRODOTTI COMMERCIALI PIÙ DIFFUSI COMPREDONO:
  - MICROSOFT: SQL SERVER.
  - ORACLE: ORACLE9I.
  - IBM: IBM DATABASE 2 (DB2) UNIVERSAL DATABASE SERVER FOR OS/390.
  - MICROSOFT: ACCESS.
- IN AMBITO OPEN SOURCE:
  - MY SQL ([HTTP://WWW.MYSQL.COM/](http://www.mysql.com/))
  - POSTGRESS ([HTTP://WWW.POSTGRESQL.ORG/](http://www.postgresql.org/))
  - INFORMIX ([HTTP://WWW-306.IBM.COM/SOFTWARE/DATA/INFORMIX/](http://www-306.ibm.com/software/data/informix/))

# APPROFONDIMENTI

- *DOMANDE???*

