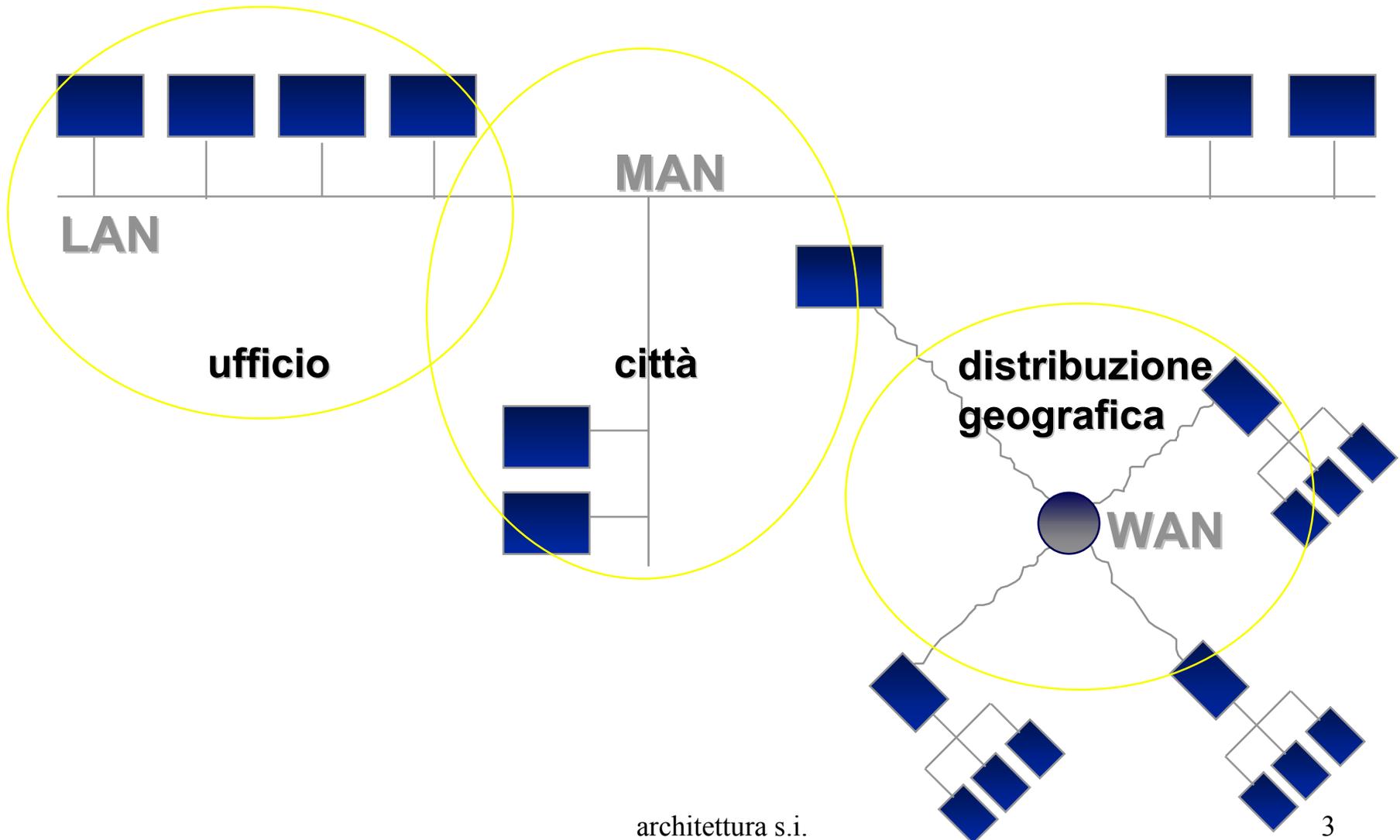


# Tecnologia dei Sistemi Informativi

# Sistema Informativo

- **comprende risorse umane**
- **è fortemente integrato con il sistema organizzativo**
- **è essenziale per il funzionamento dell'azienda**

# Architettura dei Sistemi Informatici



# Paradigma Client-Server

- tecnica per strutturare sistemi software
- viene resa “pubblica” una “interfaccia di servizi”
- due tipologie di sistemi:

**CLIENT**

richiedono i servizi

**SERVER**

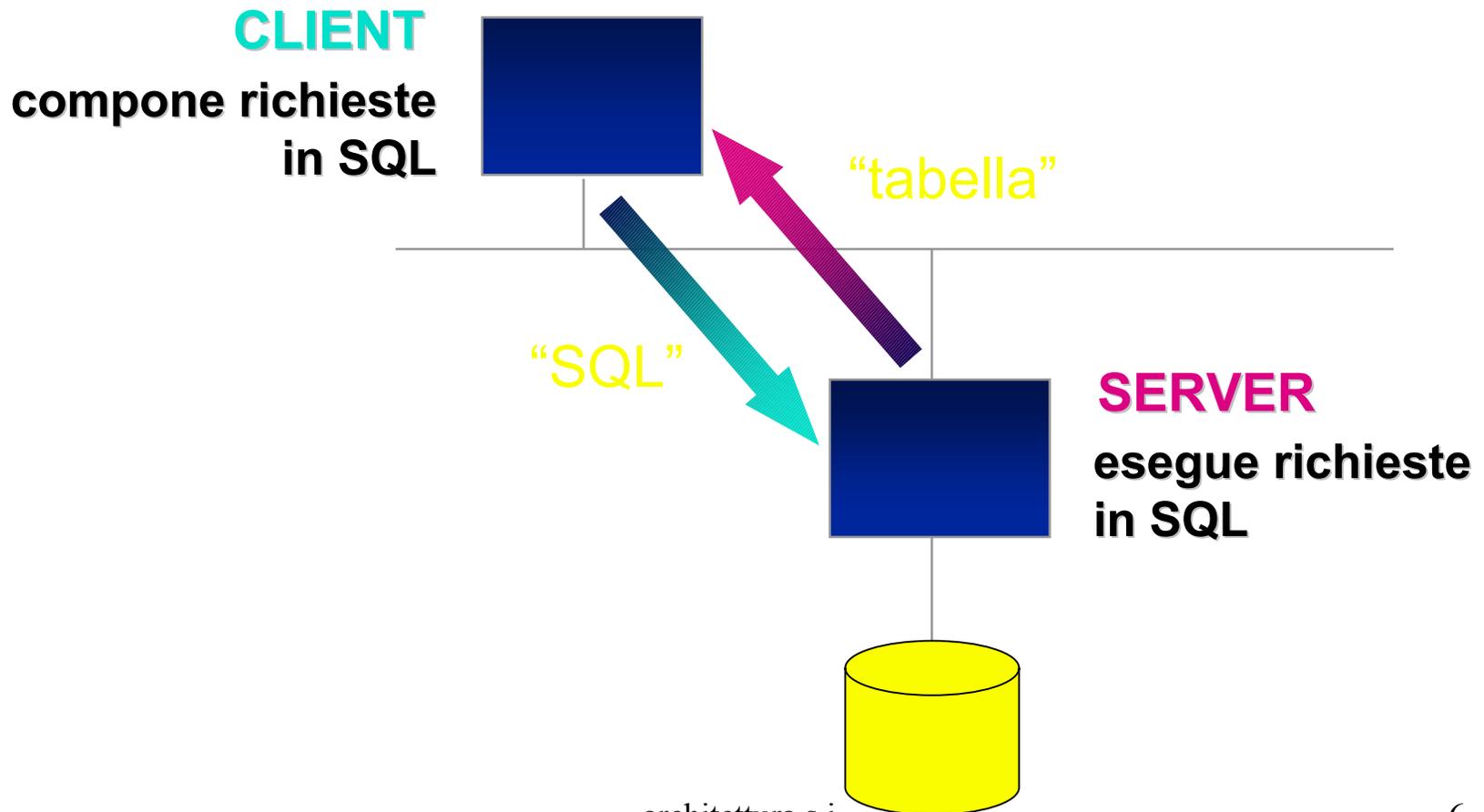
forniscono i servizi



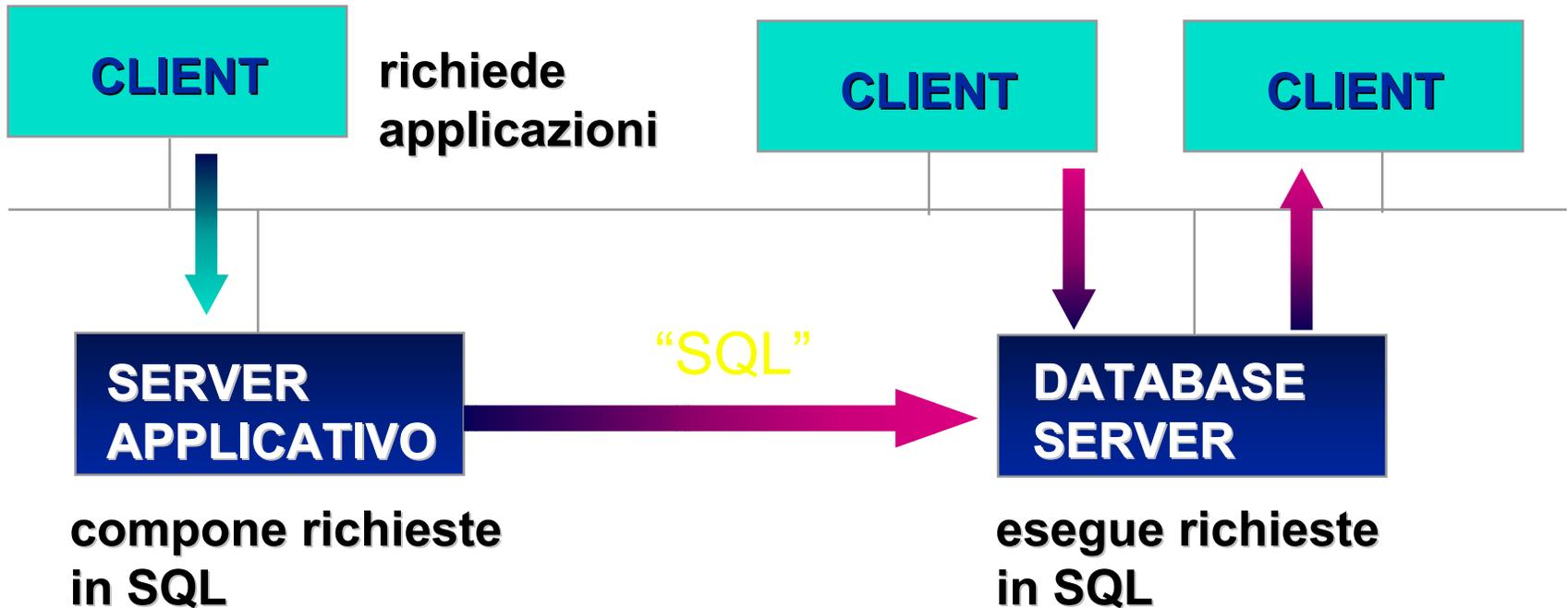
# Client-Server nei Sistemi Informativi

- **separazione funzionale ideale**
  - CLIENT** : presentazione dell'informazione
  - SERVER** : gestione dei dati
- **SQL** : il linguaggio ideale per separare gli ambienti
  - CLIENT** : formula query, elabora risultati
  - SERVER** : esegue query
  - RETE** : trasferisce i comandi di attivazione  
(es: di procedure SQL)

# Architettura Client-Server classica (1 tier)



# Architettura (2-tier) con Server Applicativo



# Esempi di Server Applicativi

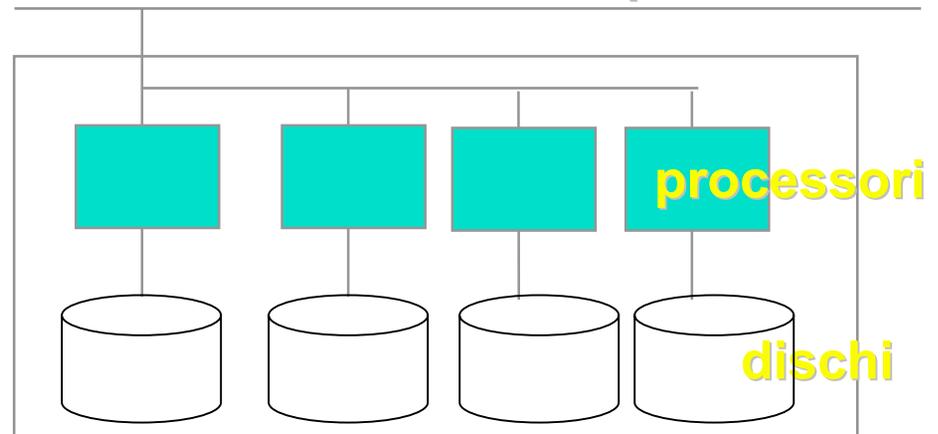
- **Sistemi di Presentazione**  
(es. per grafici, fogli di calcolo)
- **Decision Support System (DSS)**
- **Data Warehouse (DW)**
- **Workflow (WfMS)**
- **Enterprise Resource Planning (ERP)**
- ...

# Parallelismo nei DB Server

- uso di macchine multiprocessore
- elaborazione identica su ciascun processore
- obiettivo : prestazioni

→ base di dati parallela

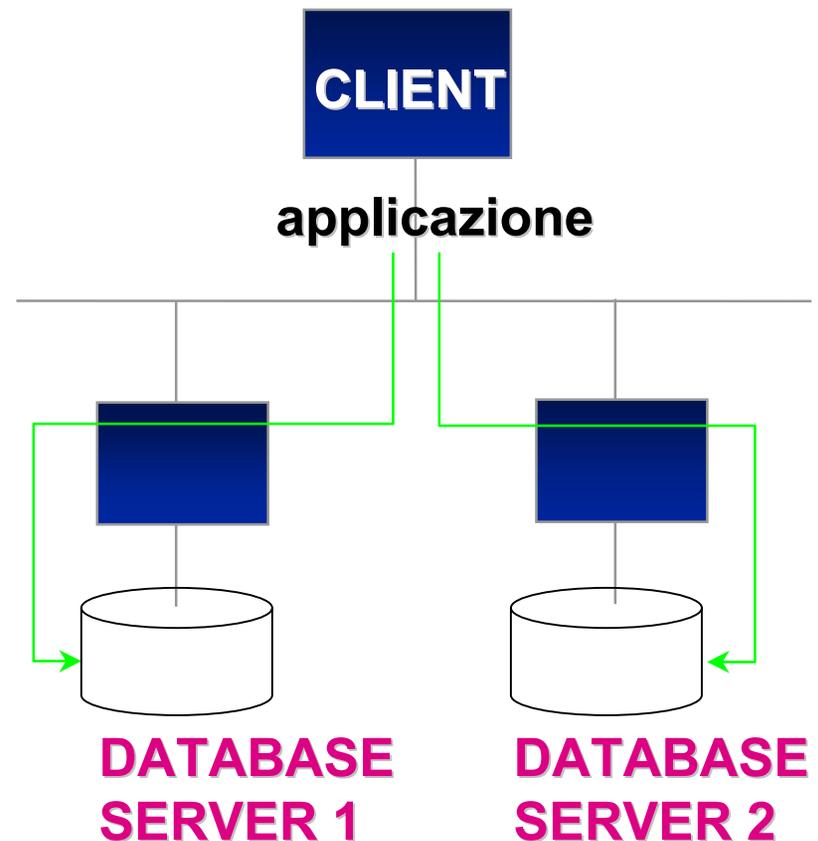
**DATABASE SERVER**  
con parallelismo



# Distribuzione dei Dati

- **NON SOLO** presenza di molteplici basi di dati
- **MA ANCHE** presenza di applicazioni che usano dati provenienti da più fonti

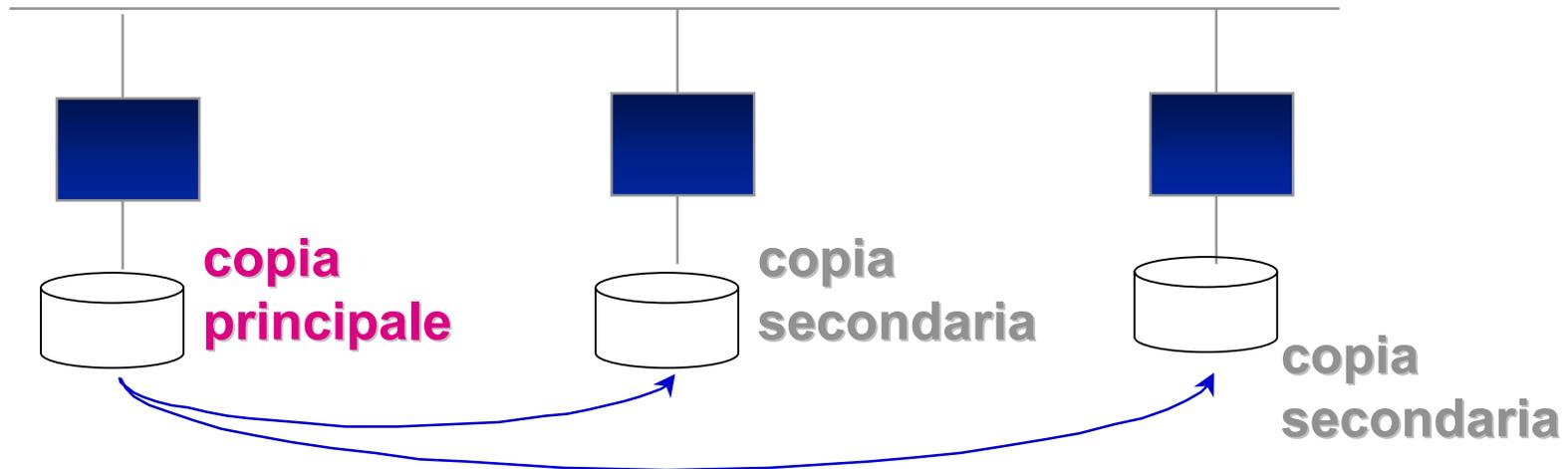
→ **base di dati distribuita**



# Replicazione dei Dati

- **motivazione: maggior disponibilità, efficienza, differente trattamento dei dati**

→ **base di dati replicata**



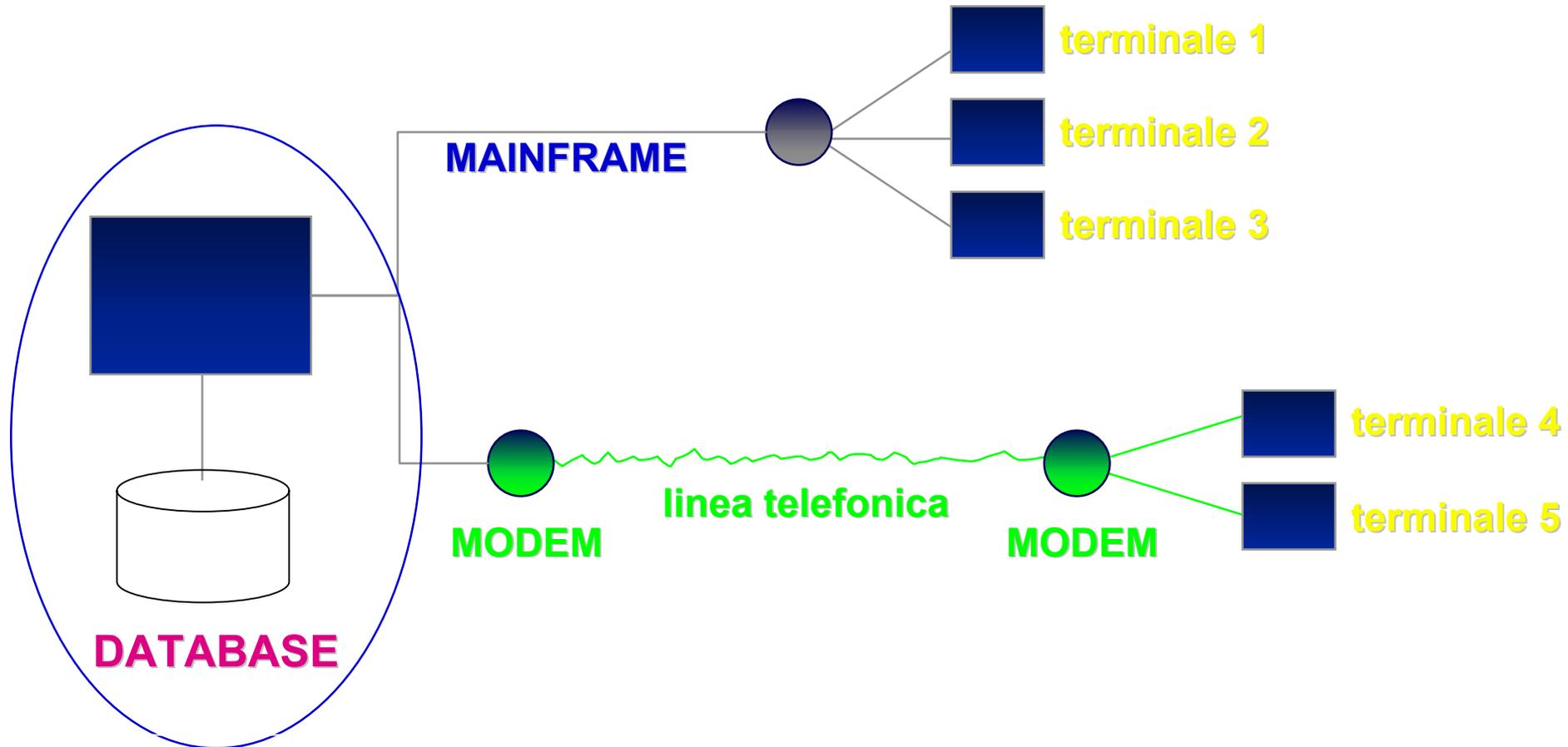
# Sistemi Legacy

- sistemi realizzati su architetture basate su **MAINFRAME** (grande calcolatore centrale) cui si collegano terminali poco sofisticati (con interfaccia a caratteri)

*...sistemi “ereditati”*

- **“FUNZIONANO!”**

# Sistemi Legacy



sistema legacy

# Molti sistemi Legacy sono tecnologicamente superati

- scelta dell'hardware
- scelta del software (COBOL, DL/1, RPG)
- uso archivi settoriali (senza DBMS)

*però sono sistemi affidabili in operatività  
il cui funzionamento continuo è  
indispensabile*

# Evoluzione dei sistemi Legacy

- downsizing:
  - trasformazione verso architetture client-server
- incapsulamento:
  - definizione di funzioni server di alto livello richiamabili tramite interfacce client standard che mascherano la reale struttura del software

# Interoperabilità

- **capacità di costruire applicazioni con sistemi fra loro eterogenei:**
  - **diverso sistema operativo**
  - **diverso protocollo di comunicazione**
  - **diverso software di gestione dei dati**

# Soluzioni per l'Interoperabilità

- 1 adozione di standard industriali**
- 2 impiego di GATEWAY:  
emulazione di un ambiente in un altro**

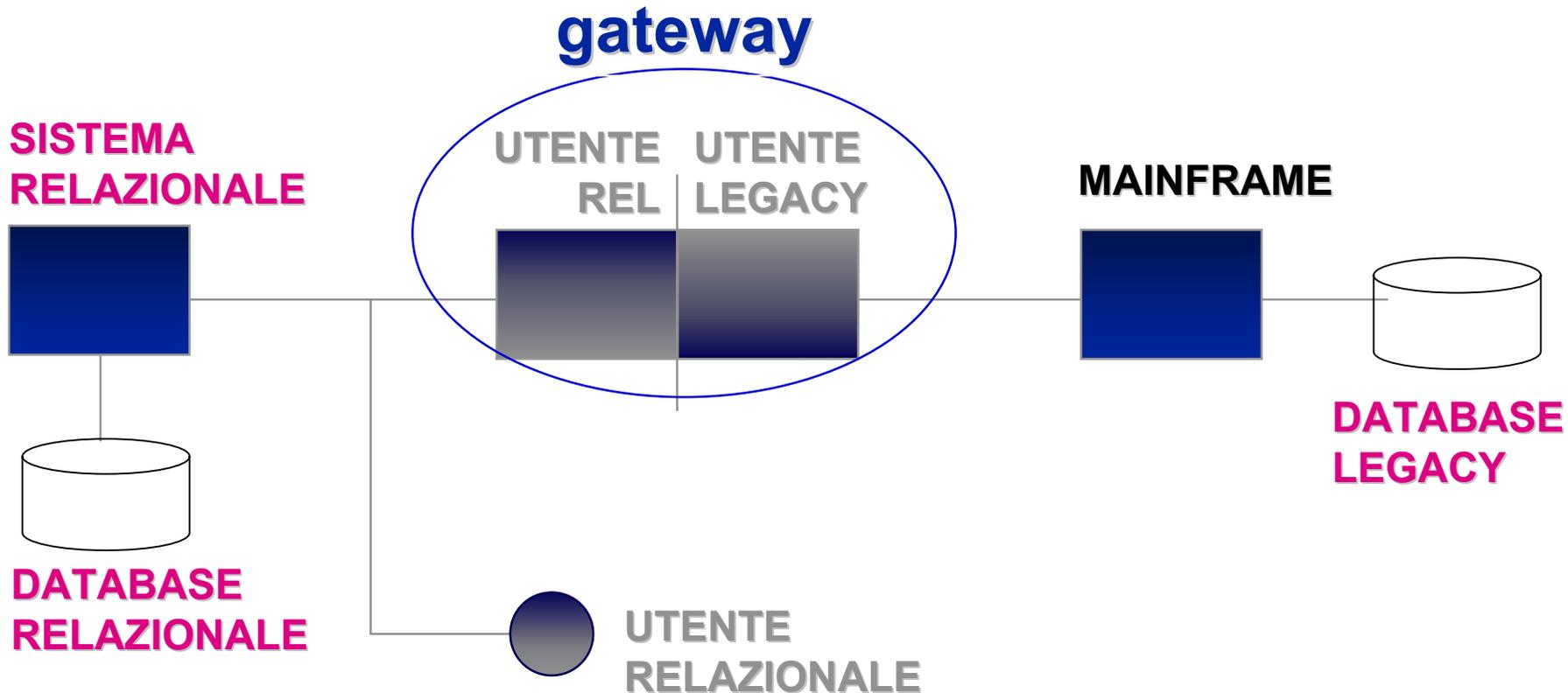
# Uso di Standard

(un mondo complicatissimo e in perenne evoluzione)

- a software di costruzione delle applicazioni  
(ODBC)**
- b software di gestione dei dati  
(SQL)**
- c software di gestione delle reti  
(TCP-IP)**
- d software di base (sistema operativo)  
(UNIX, Windows-NT)**

# Uso di GATEWAY

- tra sistemi relazionali
- verso **LEGACY SYSTEM**



# Evoluzione delle Interfacce Utente

- amichevoli, orientate ad utenti inesperti
- grafiche (GUI: graphic user interface)
- spesso multimediali
- guidate dallo sviluppo corrispondente dei sistemi operativi (ambienti Mac, Windows, Linux)
- influenzate da Internet (Web)