

Sistemas de Bases de Datos

Sistemas de Base de Dados

- Definições Básicas de Bases de Dados
- Características e Funcionalidades
- O Modelo Relacional
- Álgebra Relacional e SQL

Definições Básicas

- **Dados:** factos conhecidos que têm algum significado e que podem ser guardados.
- **Base de dados (BD):** conjunto de dados que se relacionam entre si.
- **Universo:** parte do mundo real sobre o qual os dados guardados na base de dados dizem respeito.
 - Empresa: empregados, departamentos, projetos, ...
 - Escola: alunos, professores, cadeiras, inscrições, horários, ...
 - Contactos: nomes, endereços, telefones, ...

Um exemplo de base de dados

CADEIRA	NumCad	Nome	Docente
	12347	Bases de Dados	José Aguiar Mota
	34248	Álgebra	Maria das Dores
	32439	Introdução aos Computadores	Carlos Duarte

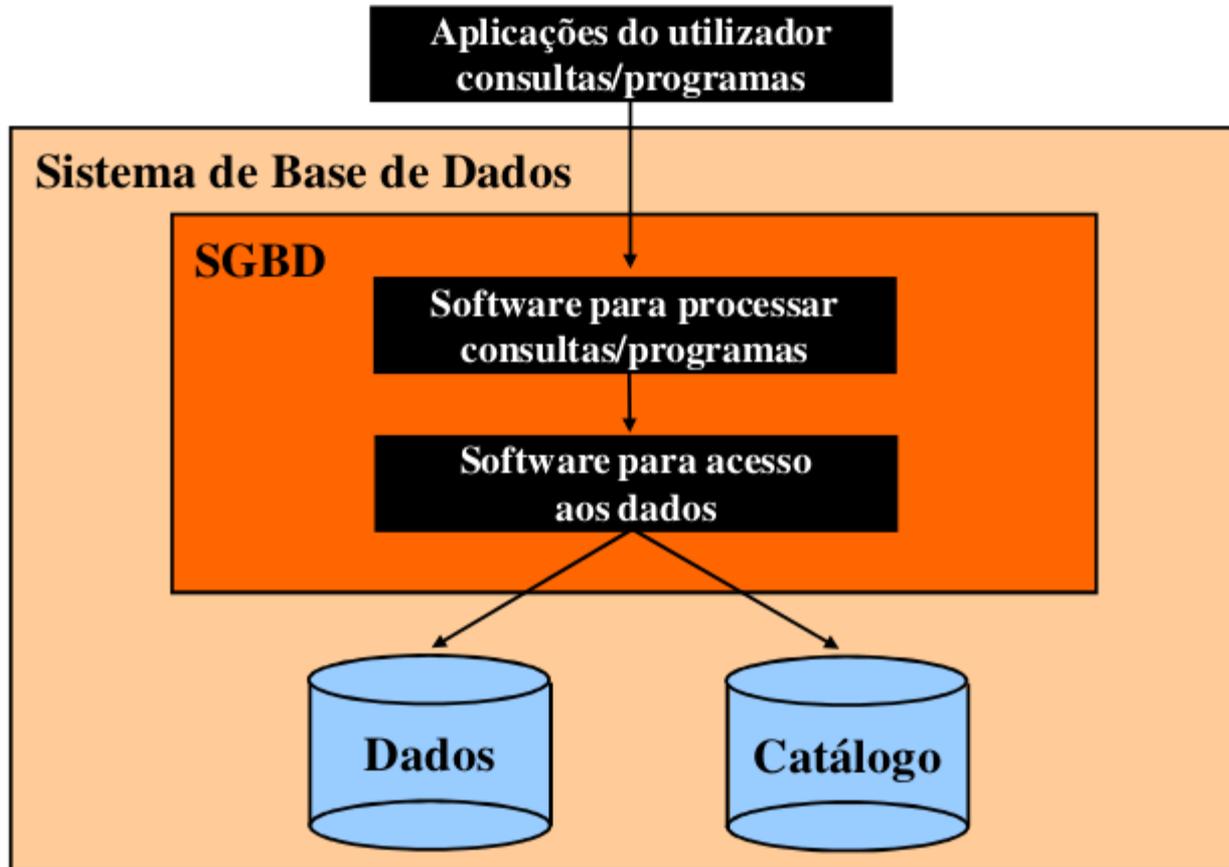
ALUNO	NumMec	Nome	Curso
	798764544	João Pinto	CC
	345673451	Carlos Semedo	ERSI
	487563546	Maria Silva	EG
	452212348	Pedro Costa	MAT

INSCRIÇÃO	NumMec	NumCad
	798764544	12347
	345673451	12347
	798764544	34248
	452212348	32439

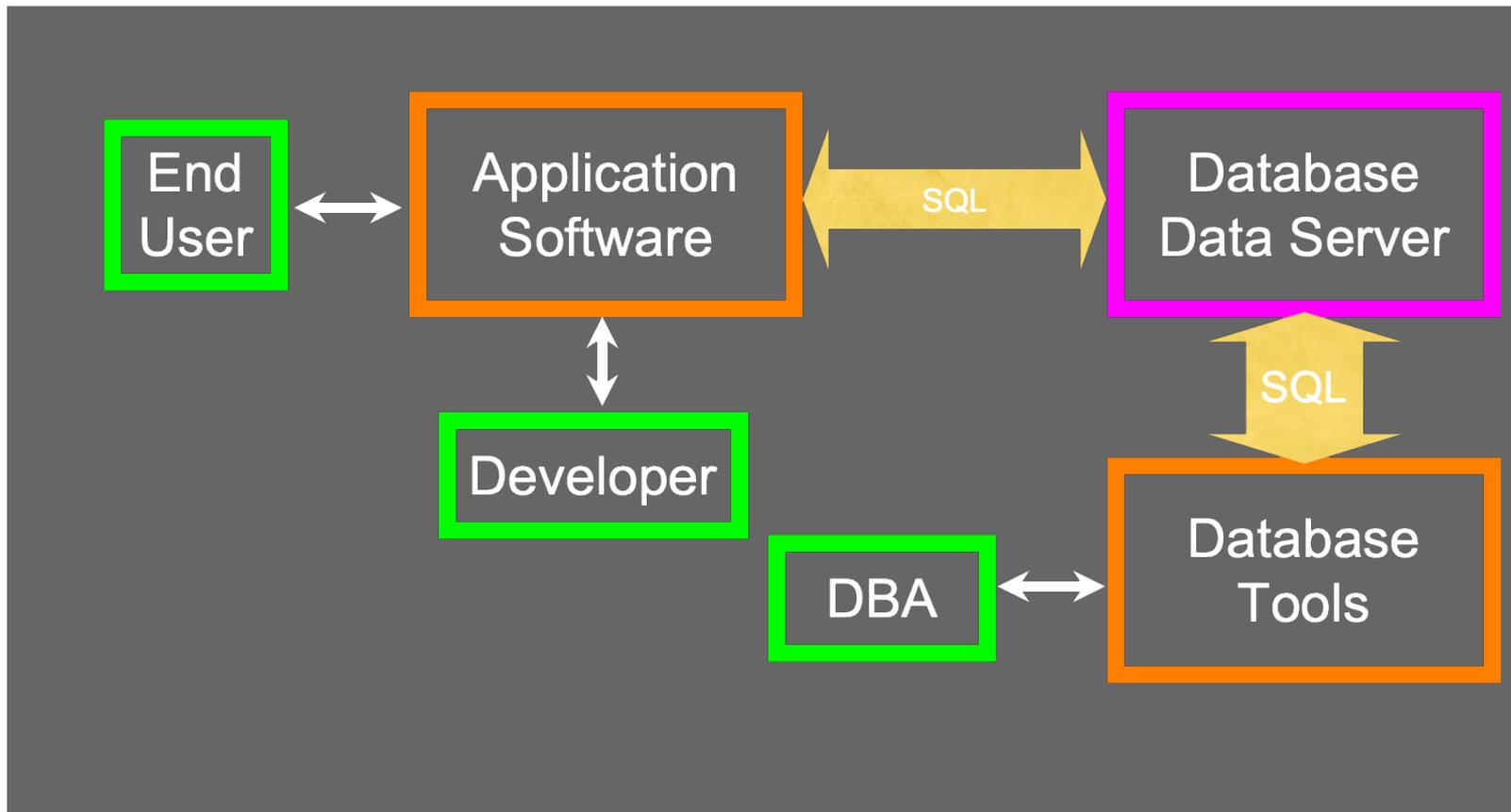
Definições Básicas

- **Sistema de gestão de base de dados (SGBD):** pacote de software que permite criar e manipular uma base de dados.
 - MySQL, MariaDB, PostgreSQL, ...
- **Sistema de base de dados (SBD):** o sistema de gestão de base de dados juntamente com os dados.

Sistema de base de dados



Sistema de base de datos



Características de um SBD

- **Abstração dos dados**

- Num sistema tradicional de ficheiros, a estrutura dos ficheiros com os dados está inserida nos programas que manipulam esses ficheiros.
- Dificuldade em alterar a organização dos dados. A alteração de um ficheiro de dados, obriga à alteração de todos os programas que manipulam esse ficheiro.
- Num sistema de BD, a estrutura dos ficheiros está no catálogo do SGBD e portanto separada dos programas de acesso. Conduz à **independência dados/programa**.

Características de um SBD

• Independência dos dados

- O **catálogo** do sistema guarda a descrição da BD (os **meta-dados**).
 - Estrutura de cada ficheiro usado para a BD.
 - Tipo e formato de cada item de dados.
 - Restrições sobre os dados.
- Torna o SGBD independente da BD.
- Permite que o SGBD funcione com diferentes BDs.

Características de um SBD

- **Suporte de visões múltiplas dos dados**
 - Permite fornecer diferentes perspectivas (visões) dos dados para diferentes utilizadores.
 - Uma visão pode ser um subconjunto de dados da BD, ou um subconjunto de dados(virtuais) derivados a partir de dados da BD.

Características de um SBD

- **Partilha de dados e acesso multi-utilizador**

- O SGBD tem de garantir que cada transação ou é executada corretamente ou é abortada por completo.
 - Restaurar o estado da BD quando ocorrem falhas durante a execução de uma transação.
- Controlar **concorrência** para garantir **consistência** e correcção nas actualizações da BD.
- Vários agentes de viagens a tentarem reservar um lugar num mesmo avião. O SGBD tem de garantir que cada lugar só pode ser reservado por apenas um agente.

Funcionalidades Típicas de um SGBD

- **Definição:** tipo de dados, tipo de relações e conjunto de restrições.
- **Manipulação:** inserir dados, apagar dados, alterar dados, fazer consultas, garantir a satisfação das restrições de integridade.
- **Construção:** representação simples e eficientes de relações complexas entre os dados, guardar os dados num local controlado pelo próprio SGBD, persistência dos dados.
- **Rentabilidade:** minimizar o esforço de desenvolvimento e manutenção, controlar a redundância nos dados, mecanismos eficientes para processamento de consultas.

Funcionalidades Típicas de um SGBD

- **Concorrência e partilha:** permitir que vários utilizadores e/ou programas acessem em simultâneo à base de dados, mantendo a consistência dos dados.
- **Protecção:** mecanismos de backup e recuperação para prevenir situações de avaria do hardware e/ou do software.
- **Segurança:** mecanismos para prevenir acessos não autorizados (passwords, permissões, diferentes níveis de acesso).
- **Visualização:** ferramentas gráficas para operações mais comuns.
- **Interacção com outras aplicações:** providenciar múltiplas interfaces com o utilizador.

Quando não utilizar um SGBD

- **Um SGBD não é necessária quando:**
 - A BD é simples, bem definida e não se espera que mude.
 - O acesso multi-utilizador não é necessário.
 - O custo de configuração do SGBD não é compatível com as necessidades imediatas de utilização de determinadas aplicações.
- **Um SGBD não é suficiente quando:**
 - Não é possível modelar um determinado problema.
 - O utilizador necessita de operações não suportadas pelo SGBD.
 - Manipulação de áudio e imagem.
 - Análise de vastos conjuntos de dados resultantes de experiências científicas.

Modelos de Dados

- **Modelo de Dados**

- Conjunto de conceitos que descrevem a estrutura da BD.
- Conjunto de restrições que a BD deve obedecer.

- **Modelo Conceptual:** tipo de modelo onde os conceitos estão perto do modo que os utilizadores percebem os dados (entidades, atributos, relacionamentos).

- Modelos ER, EER, UML.

- **Modelo Representacional:** tipo de modelo normalmente utilizado pelos SGBDs.

- Modelos relacional, hierárquico, network.

- **Modelo Físico:** tipo de modelo que descreve como os dados estão organizados e guardados no computador.

- Formato dos registos, ordem dos registos, caminhos para acesso aos dados.

Quando não utilizar um SGBD

- **Um SGBD não é necessária quando:**
 - A BD é simples, bem definida e não se espera que mude.
 - O acesso multi-utilizador não é necessário.
 - O custo de configuração do SGBD não é compatível com as necessidades imediatas de utilização de determinadas aplicações.
- **Um SGBD não é suficiente quando:**
 - Não é possível modelar um determinado problema.
 - O utilizador necessita de operações não suportadas pelo SGBD.
 - Manipulação de áudio e imagem.
 - Análise de vastos conjuntos de dados resultantes de experiências científicas.

Desenho de uma BD

•Fase I: Requisitos e análise

- Entrevistas com os potenciais utilizadores da BD.
- Compreender e documentar os seus requisitos.

•Fase II: Desenho conceptual (ou modelação)

- Definir um modelo de dados conceptual que inclua a descrição das entidades da BD, dos atributos das entidades, dos relacionamentos e das possíveis restrições.
- Evitar detalhes de implementação.

•Fase III: Desenho lógico (ou implementação)

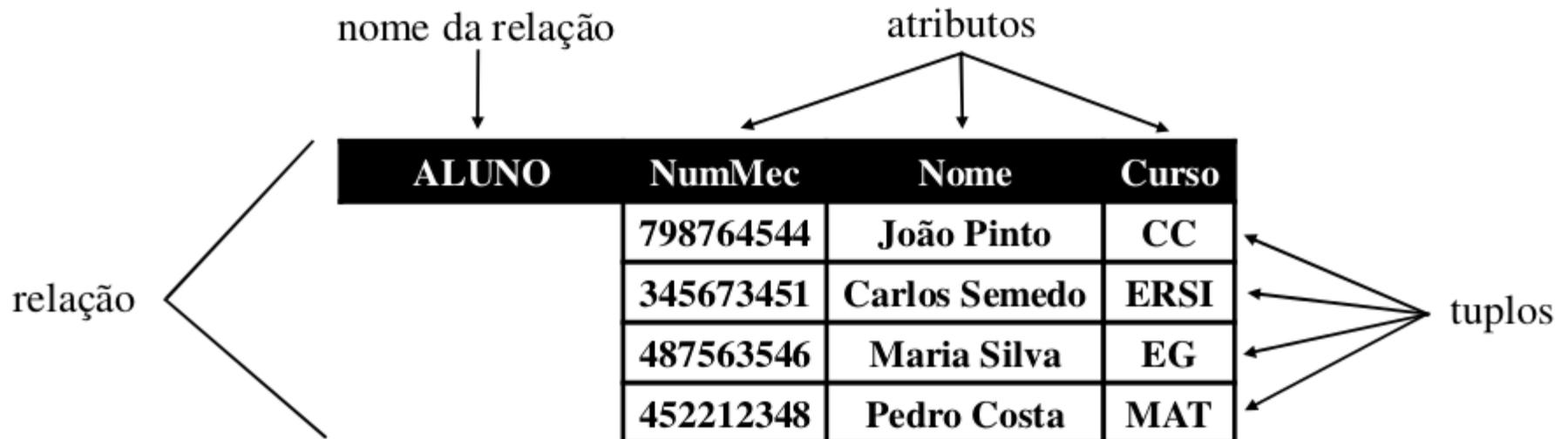
- Mapear o modelo de dados conceptual no modelo de dados representacional.
- Implementação da BD usando um SGBD.

•Fase IV: Desenho físico

- Mapear o modelo de dados representacional no modelo de dados físico.
- Estruturas em memória e organização dos ficheiros da BD (ficheiros de índices).

Modelo Relacional

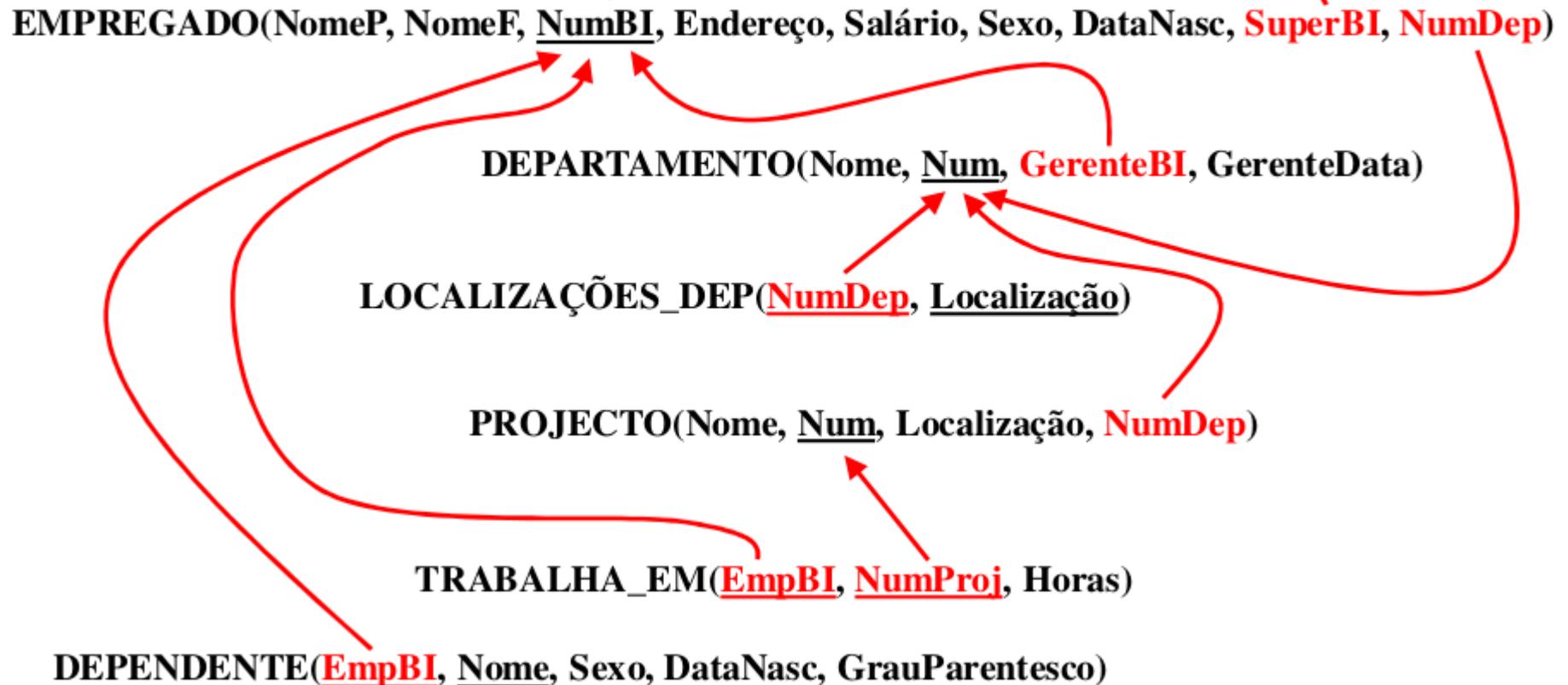
- **Relação:** uma tabela retangular
 - **Atributo:** uma coluna na tabela
 - **Tuplo:** uma linha na tabela



Chaves

- **Chave:** conjunto de atributos de uma relação para a qual todos os os tuplos são diferentes
 - Uma relação pode ter várias chaves mas apenas uma deve ser designada como **chave primária** (usualmente representada sublinhada)
- **Chave externa:** conjunto de atributos que referenciam a chave de outra relação
 - Integridade referencial: garante consistência

Esquema relacional de uma BD de empresa



Restrições de Integridade

- **Integridade de domínio**

- Os valores de um atributo devem pertencer ao domínio do atributo (tipo de dados).

- **Integridade da chave**

- Não podem existir dois tuplos de uma relação com valores iguais na chave primária.

- **Integridade de entidade**

- Os valores da chave primária não podem ser nulos.

- **Integridade referencial**

- Um tuplo que referencia outra relação tem de referenciar um tuplo existente nessa outra relação (chave externa).

Questões no Design Relacional

- **Evitar múltiplos conceitos dentro de uma relação**
 - Pode levar a dados redundantes
 - A eliminação de um tuplo poderia também eliminar informações necessárias mas não relacionadas

Melhorar o Design Relacional

- **Decomposição**

- Dividir as colunas de uma relação em duas ou mais relações, duplicando as colunas necessárias para manter as relações

Uma relação contendo redundância

Empl Id	Name	Address	SSN	Job Id	Job Title	Skill Code	Dept	Start Date	Term Date
25X15	Joe E. Baker	33 Nowhere St.	111223333	F5	Floor manager	FM3	Sales	9-1-2009	9-30-2010
25X15	Joe E. Baker	33 Nowhere St.	111223333	D7	Dept. head	K2	Sales	10-1-2010	*
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999	F5	Floor manager	FM3	Sales	10-1-2009	*
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555	S25X	Secretary	T5	Personnel	3-1-1999	4-30-2010
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555	S26Z	Secretary	T6	Accounting	5-1-2010	*
.
.
.

Uma base de dados de empregados com três relações

EMPLOYEE relation

Empl Id	Name	Address	SSN
25X15	Joe E. Baker	33 Nowhere St.	111223333
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555

JOB relation

Job Id	Job Title	Skill Code	Dept
S25X	Secretary	T5	Personnel
S26Z	Secretary	T6	Accounting
F5	Floor manager	FM3	Sales
•	•	•	•
•	•	•	•
•	•	•	•

ASSIGNMENT relation

Empl Id	Job Id	Start Date	Term Date
23Y34	S25X	3-1-1999	4-30-2010
34Y70	F5	10-1-2009	*
23Y34	S26Z	5-1-2010	*
•	•	•	•
•	•	•	•
•	•	•	•

Encontrar os departamentos em que o empregado 23Y34 trabalhou

EMPLOYEE relation

Empl Id	Name	Address	SSN
25X15	Joe E. Baker	33 Nowhere St.	111223333
34Y70	Cheryl H. Clark	563 Downtown Ave.	999009999
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555
•	•	•	•
•	•	•	•
•	•	•	•

JOB relation

Job Id	Job Title	Skill Code	Dept
S25X	Secretary	T5	Personnel
S26Z	Secretary	T6	Accounting
F5	Floor manager	FM3	Sales
•	•	•	•
•	•	•	•
•	•	•	•

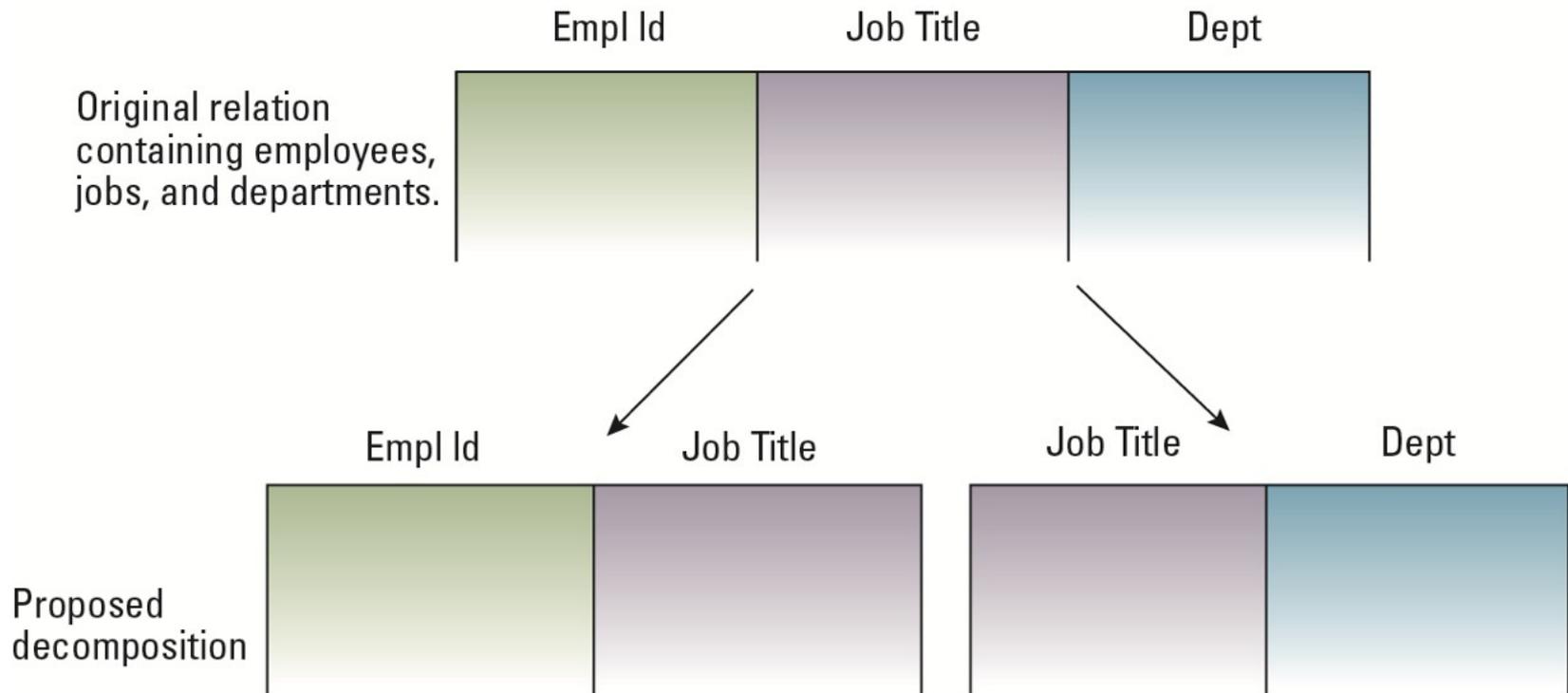
are contained in the personnel and accounting departments.

ASSIGNMENT relation

Empl Id	Job Id	Start Date	Term Date
23Y34	S25X	3-1-1999	4-30-2010
34Y70	F5	10-1-2009	*
23Y34	S26Z	5-1-2010	*
•	•	•	•
•	•	•	•
•	•	•	•

The jobs held by employee 23Y34

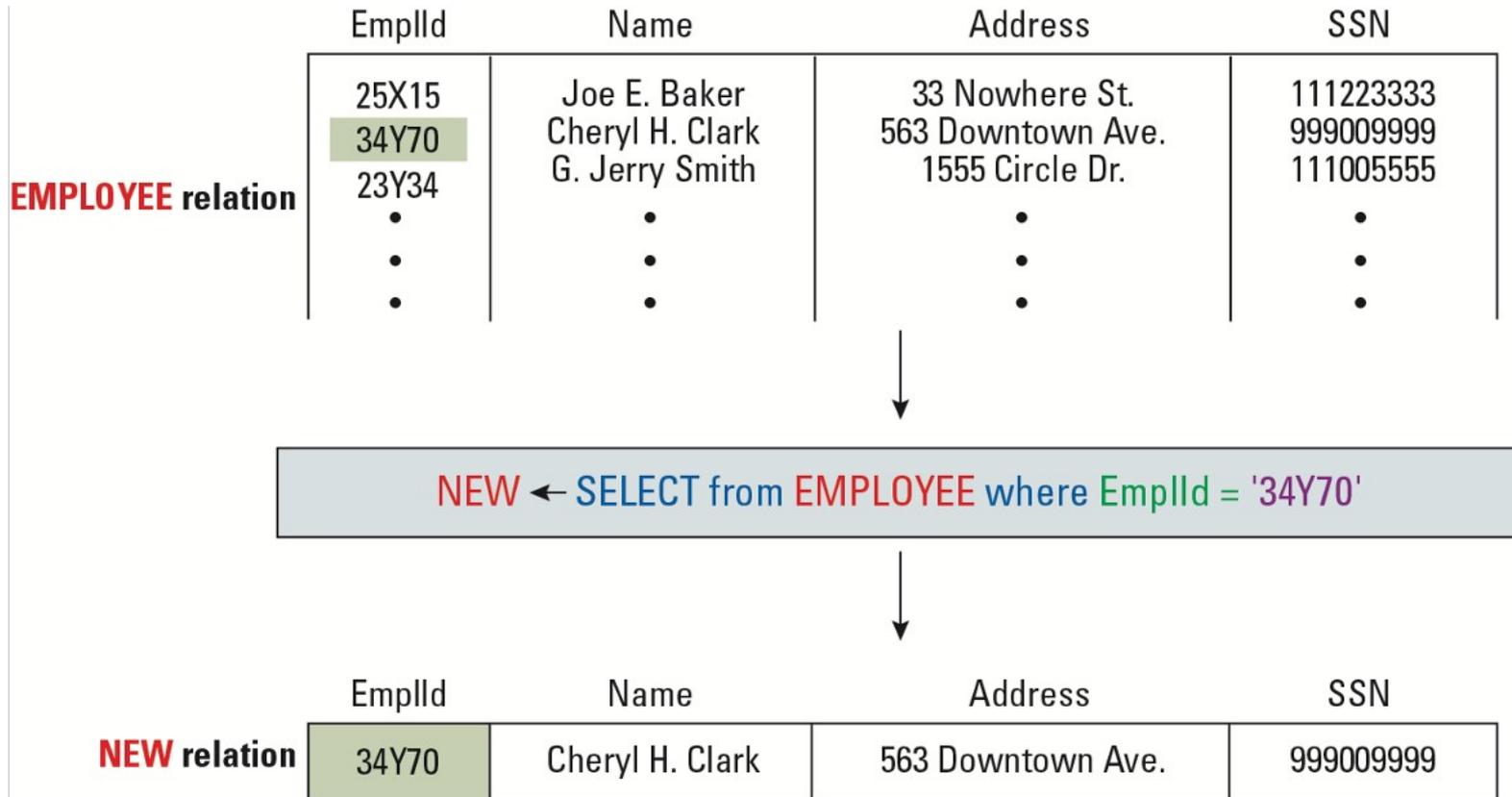
Uma relação e uma proposta de decomposição



Operações relacionais

- “Álgebra relacional”
- *Select*: escolher linhas
- *Project*: escolher colunas
- *Join*: juntar informação de duas ou mais relações

SQL: A operação SELECT



SQL: A operação PROJECT

EMPLOYEE relation

EmplId	Name	Address	SSN
25X15	Joe E. Baker	33 Nowhere St.	111223333
24Y70	Cheryl H. Clark	563 Downtown Ave.	999009999
23Y34	G. Jerry Smith	1555 Circle Dr.	111005555
•	•	•	•
•	•	•	•
•	•	•	•



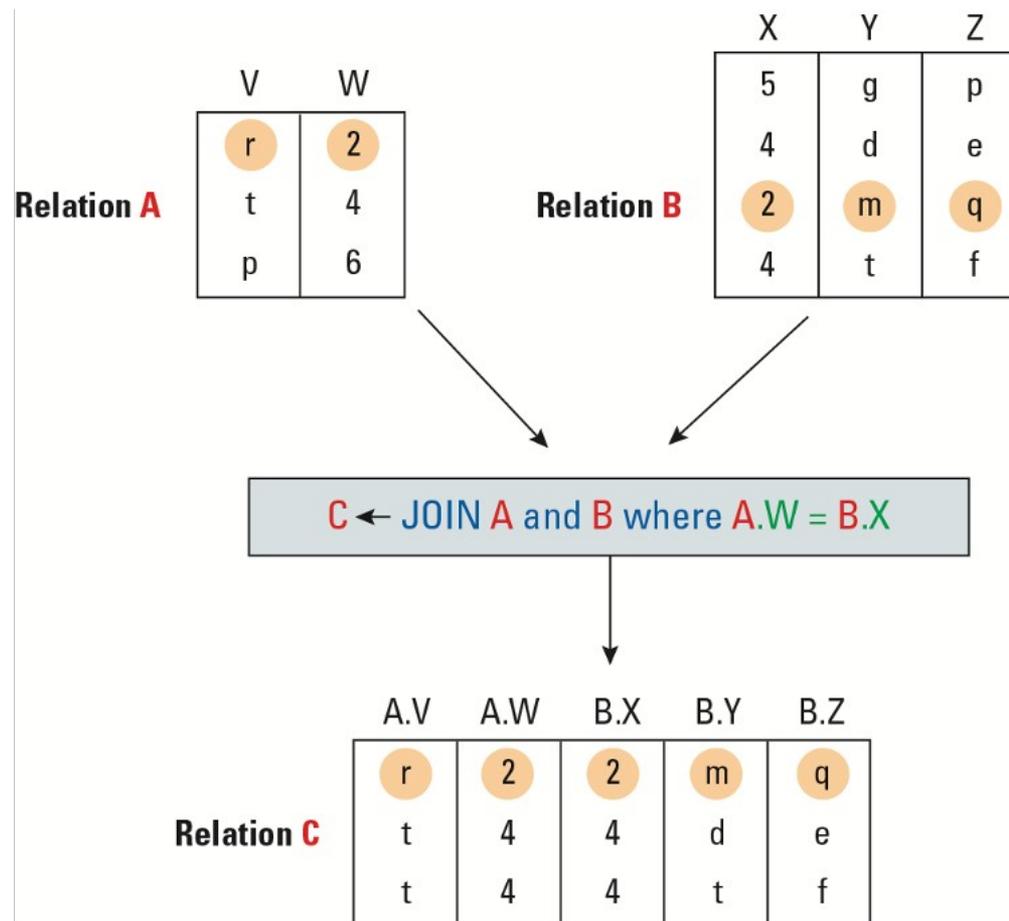
MAIL ← PROJECT Name, Address from **EMPLOYEE**



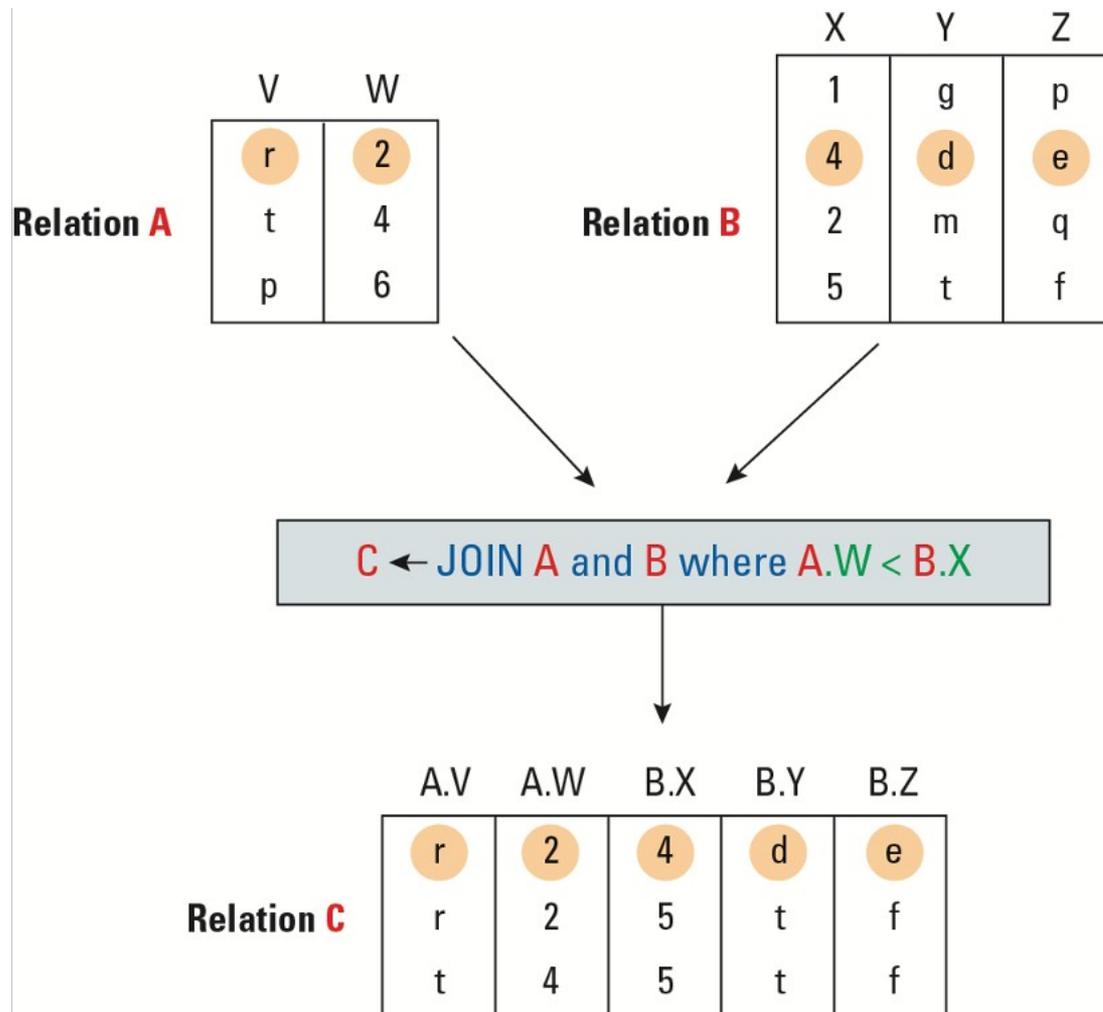
MAIL relation

Name	Address
Joe E. Baker	33 Nowhere St.
Cheryl H. Clark	563 Downtown Ave.
G. Jerry Smith	1555 Circle Dr.
•	•
•	•
•	•

SQL: A operação JOIN



SQL: Outro exemplo da operação JOIN



SQL: Uma aplicação da operação JOIN

ASSIGNMENT relation

Empl Id	Job Id	Start Date	Term Date
23Y34	S25X	3-1-1999	4-30-2010
34Y70	F5	10-1-2009	*
25X15	S26Z	5-1-2010	*
•	•	•	•
•	•	•	•
•	•	•	•

JOB relation

Job Id	Job Title	Skill Code	Dept
S25X	Secretary	T5	Personnel
S26Z	Secretary	T6	Accounting
F5	Floor manager	FM3	Sales
•	•	•	•
•	•	•	•
•	•	•	•

NEW1 ← JOIN ASSIGNMENT and JOB where ASSIGNMENT.JobId = JOB.JobId

NEW1 relation

ASSIGNMENT Empl Id	ASSIGNMENT Job Id	ASSIGNMENT StartDate	ASSIGNMENT TermDate	JOB Job Id	JOB JobTitle	JOB SkillCode	JOB Dept
23Y34	S25X	3-1-1999	4-30-2010	S25X	Secretary	T5	Personnel
34Y70	F5	10-1-2009	*	F5	Floor manager	FM3	Sales
25X15	S26Z	5-1-2010	*	S26Z	Secretary	T6	Accounting
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•

Structured Query Language (SQL)

- Operações para manipular tuplos

- insert

- update

- delete

- select ... from ... where

SQL exemplos

- ```
SELECT EmpId, Dept
FROM Assignment, Job
WHERE Assignment.JobId = Job.JobId
 AND Assignment.TermData = '*';
```
- ```
INSERT INTO Employee
VALUES ('43212', 'Sue A. Burt',
      '33 Fair St.', '444661111');
```

SQL exemplos

- `DELETE FROM Employee
WHERE Name = 'G. Jerry Smith';`
- `UPDATE Employee
SET Address = '1812 Napoleon Ave.'
WHERE Name = 'Joe E. Baker';`

Bases de dados NoSQL

- Não aderentes ao uso do SQL
- Otimizadas para muito larga escala
- Providenciam essencialmente *retrieval* e *appending*
- Não necessitam das relações
- Ex: Apache Casandra, Voldemort, Amazon Dynamo, Google BigTable