

# Bases de datos

**UML**

**Lenguajes de consulta**

UML.

Cálculo relacional de tuplas (CRT). Lenguaje QBE

Cálculo relacional de dominios (CRD). Lenguaje QUEL

## Lenguaje **unificado** de modelado

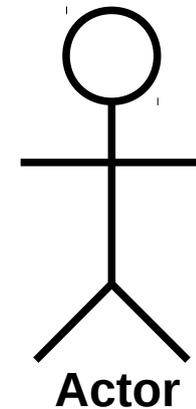
- Modelado visual de sistemas de software
- Especificar, visualizar, construir y documentar los artefactos de un sistema de software.
  
- Capta la información sobre la *estructura estática* y el *comportamiento dinámico* del sistema
  
- Modelado del software desde diferentes perspectivas (vistas)

# Proyecto - 1era fase

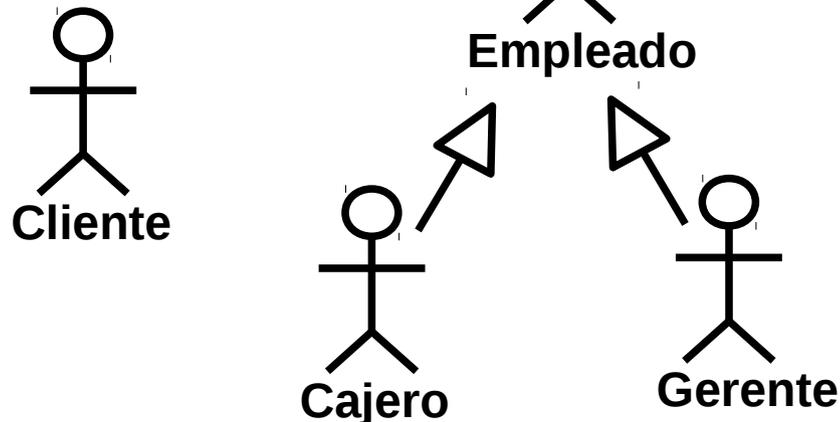
- Modelo de actores (Vista estructural o estática)

**Actor:** Idealización de un rol de una persona externa, proceso o cosa que interactúe con el software.

*Cada actor participa en uno o mas casos de uso.*



## Jerarquías de actores



# Proyecto - 1era fase

- **Diagramas de actividades** de los procesos involucrados (Vista dinámica)

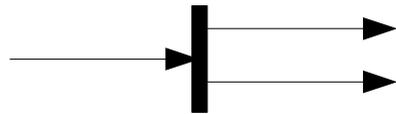
**Grafo de nodos y flujos que muestra el flujo de control (y a veces datos) de un proceso**

*Proceso*: conjunto estructurado de actividades diseñado para alcanzar un objetivo. Es ejecutado por un conjunto de actores que pertenecen a unidades organizativas, cumple con un conjunto de reglas de negocio y cumple un objetivo utilizando a un conjunto de objetos de negocio.

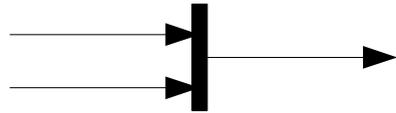
*Acción*: Unidad fundamental de especificación de comportamiento. Es atómica, no se puede descomponer en otras acciones. Puede tomar objetos de entrada y también puede producir objetos.

# Nodos de control

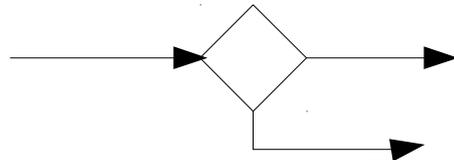
**Sincronización (Fork):** divide un flujo en 2 o mas flujos que se ejecutan en paralelo.



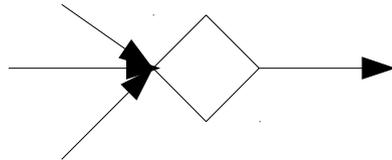
**Concurrencia (Join):** sincroniza múltiples flujos.



**Decisión:** selecciona entre dos o mas flujos de salida

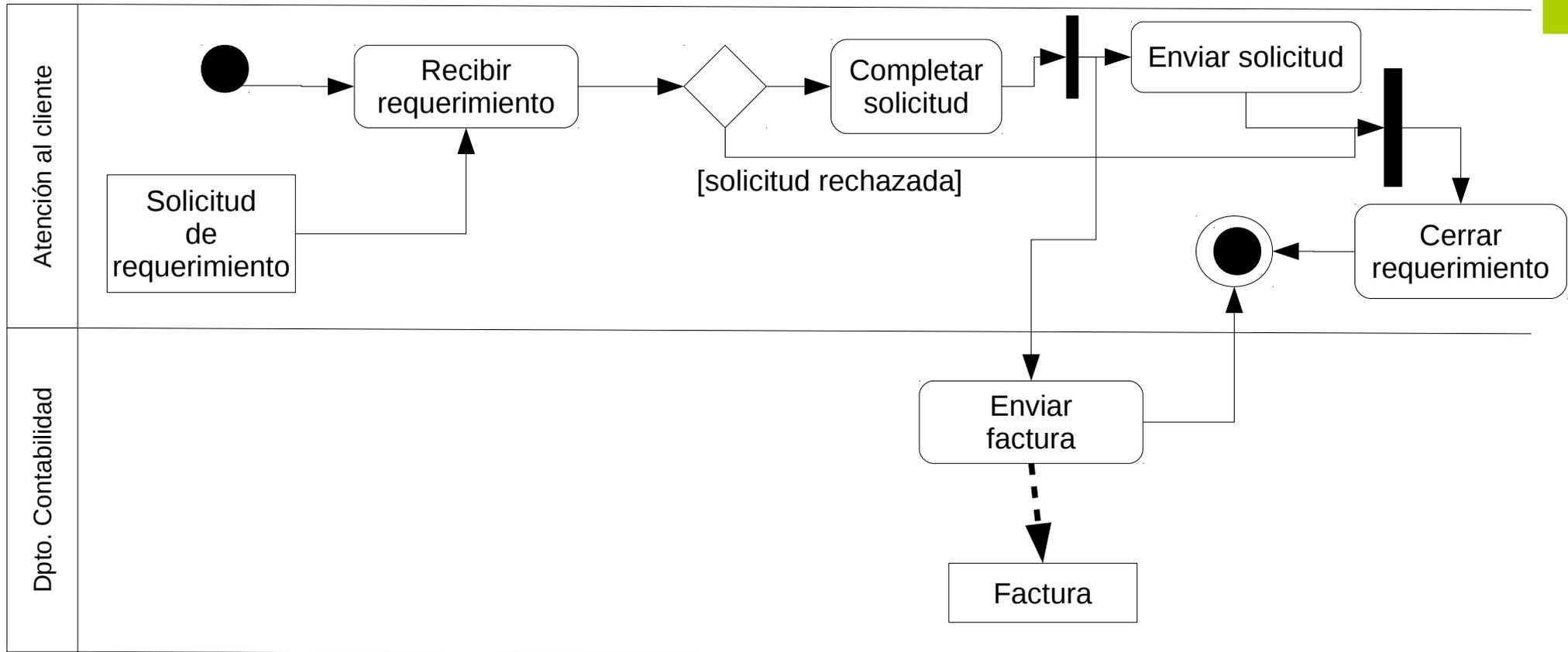


**Mezcla (fusión):** une dos o mas flujos alternativos



# Diag. de actividades

## Proceso: Atención de solicitudes



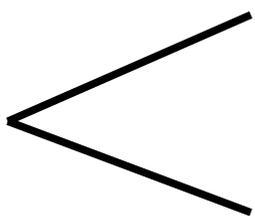
*Acciones  
Cuando y donde ocurren*

# Cálculo relacional

Lenguaje formal para consultar el modelo relacional (Codd, 1972)

Lenguaje declarativo - ¿Qué quiero?

- Una expresión de cálculo relacional especifica QUE se va a recuperar en vez de COMO recuperarlo (lenguaje no procedimental)

Cálculo relacional  Basado en tuplas  
Basado en dominios

# Cálculo relacional

## Una expresión de cálculo relacional

- crea una nueva relación
- No tiene un orden de operaciones para especificar como se recuperan los datos resultantes de la consulta, *solo va que información debe contener el resultado.*
- El estándar SQL tiene algunas de sus bases en una variación del cálculo relacional conocida como Cálculo Relacional de Tuplas (CRT)

- **Cálculo Relacional de Tuplas**, es una variación del cálculo relacional
- *Las variables abarcan tuplas*. Cada variable de tupla abarca una relación determinada.
- Forma general de consulta:

Expresión del CRT

**$\{ t \mid \text{condicion}(t) \}$**

$t$  es una tupla y  $\text{condicion}(t)$  es una expresión condicional donde interviene  $t$

El resultado de esta consulta es el conjunto de todas las tuplas  $t$  que satisfacen la  $\text{condicion}(t)$

**Ej:** Lista de empleados con salario mayor a 200.000 Bs

$\{ t \mid \text{Empleado}(t) \text{ AND } t.\text{salario} > 200000 \}$

Trae todos los atributos de la tupla

Condición: relación abarcada por la variable tupla t es Empleado

Condición: atributo salario de la Variable tupla t sea mayor a 200.000 Bs

$\{ t.\text{nombre}, t.\text{apellido} \mid \text{Empleado}(t) \text{ AND } t.\text{salario} > 200000 \}$

¿Equivalente en SQL?

```
SELECT nombre, apellido FROM Empleado WHERE salario > 200000
```

## Una expresión del CRT contiene:

- Para cada variable de tupla, la **relación R** que abarca o relación de intervalo. Se especifica como  $R(t)$
- Una **condición** para seleccionar combinaciones específicas de tuplas. Si la condición es Verdadera la tupla se incluye en el resultado
- Un **conjunto de atributos** que se solicitan en la consulta para cada tupla seleccionada.

*SELECT-FROM-WHERE*

# Condición en CRT

**condicion**( $t_1, t_2, \dots, t_n$ ) es una fórmula bien formada constituida por *átomos de tres tipos* que pueden conectarse mediante AND, OR y NOT:

- $R(t_i)$
- $t_i.att\_a \text{ operador } t_j.att\_a$ , donde operador puede ser alguno del conjunto  $\{=, \text{diferente}, <, <=, >, >=\}$
- $t_i.att\_a \text{ operador } c$  o viceversa, donde operador puede ser alguno del conjunto  $\{=, \text{diferente}, <, <=, >, >=\}$  y  $c$  es un valor constante.

# Consultas en CRT

**Producto**(codPro, nombre, cantidad, color, ubicacion\_almacen)

**Compra**(codigo, fecha, *producto*, cantidad, monto, rif\_proveedor)

**Cliente**(rif, nombre, direccion, ciudad, telefono, celular)

**Venta**(codigo, fecha, *rif\_cliente*, *producto*, cantidad, monto)

1. Listado con nombre y color de cada producto en el almacén

**{ t.nombre, t.color | Producto(t) }**

2. Listado con nombre y cantidad en existencia de cada producto de color azul en el almacén

**{ x.nombre, x.cantidad | Producto(x) AND x.color='azul' }**

3. ¿Cuál es el rif del proveedor de cada producto en el almacén?

**{ x.nombre, c.rif\_proveedor | Producto(x) AND Compra(c) AND x.codPro=c.producto }**

# Cuantificadores en CRT

En la condición de una expresión del CRT pueden aparecer *dos tipos* de cuantificadores:

- Cuantificador existencial ( $\exists$ )
- Cuantificador universal ( $\forall$ )

Variable de tupla libre

Variable de tupla ligada = variable de tupla que aparece en una cláusula ( $\exists t$ ) o ( $\forall t$ )

# Cuantificador existencial

$(\exists t)(F)$  es una fórmula donde  $t$  es una variable de tupla.

Esta fórmula es *Verdadera* si existe “**alguna**” tupla  $t$  que haga que  $F$  sea *Verdadera*.

# Cuantificador universal

$(\forall t)(F)$  es una fórmula donde  $t$  es una variable de tupla.

Esta fórmula es *Verdadera* si  $t$  se sustituye por cualquier tupla posible que se pueda asignar en las ocurrencias libres de  $t$  en  $F$ , y  $F$  es *Verdadera* después de hacer las sustituciones.

Todas las tuplas del universo de tuplas deben hacer que  $F$  sea *Verdadera*.

# Consultas en CRT

1. ¿Cuáles son los clientes que han comprado al menos un producto de color amarillo?

**{ c.nombre, t.nombre |  $\exists t$  Venta(v) AND Producto(t) AND Cliente(c)  
AND v.producto=t.codPro AND t.color='amarillo' AND  
v.rif\_cliente=c.rif }**

2. ¿Cuáles son los nombres de los productos comprados a todos los proveedores y vendidos a por lo menos un cliente?

**{ c.nombre, t.nombre |  $\forall t \exists v \exists a$  Venta(v) AND Producto(t) AND  
Compra(c) AND Compra(a) AND Cliente(cl) AND  
t.codPro=v.producto AND t.codPro=a.producto AND  
a.rifProv=c.rifProv AND c.rifProv=cl.rif }**

**Berkeley**  
UNIVERSITY OF CALIFORNIA

## Principios de 70s - 1985

5 años como  
proyecto de código  
abierto de la startup  
Ingres Corp.



Fue el primer SGBD relacional que no estuvo atado a hardware específico de IBM.  
**Postgres** fue creado como sucesor de *Ingres*

Lenguaje implementado para el SGBDR Ingres basado en el CRT.

## Especificar variables de tupla en QUEL:

- Range of <listaDeVariables> is <listaDeTablas>

## Consultas en QUEL:

- Retrieve [[into] <nombreTabla>] <listaVariableAtributo> [where <condición>]

## Manipulación de datos en QUEL:

- Append [to] <variable> (at=valor,...) [where <condición>]
- Replace <variable> (at=expresión,...) [where <condición>]
- Delete <variable> [where <condición>]

Lacroix y Pirotte

- Otro lenguaje formal para consultar BDR basado en el cálculo de predicados.
- En este lenguaje las variables abarcan valores individuales del dominio de los atributos.
- Una relación se denota como :  
**Relación(atributo1: valor1, atributo2: valor 2,...)**  
**Carro(modelo:m, marca:c, placa:p)**
- Los predicados se construyen igual que para el cálculo relacional de tuplas

# Consultas en CRD

**Producto**(codPro, nombre, cantidad, color, ubicacion\_almacen)

**Compra**(codigo, fecha, *producto*, cantidad, monto, rif\_proveedor)

**Cliente**(rif, nombre, direccion, ciudad, telefono, celular)

**Venta**(codigo, fecha, *rif\_cliente*, *producto*, cantidad, monto)

1. Listado con nombre y color de cada producto en el almacén  
**{ n, c | Producto(nombre:n, color:c) }**
2. Listado con nombre y cantidad en existencia de cada producto de color azul en el almacén  
**{ n, t | Producto(nombre:n, cantidad: t, color='azul')}**
3. ¿Cuál es el rif del proveedor de cada producto en el almacén?  
**{n, r | Producto(codPro: p, nombre:n) AND Compra(producto:p, rif\_proveedor:r)}**
4. ¿Cuáles son los clientes que han comprado *al menos* un producto de color amarillo?  
**{nc, n | ∃p Venta(rif\_cliente:r, producto:p) AND Producto(codPro:p, nombre:n, color='amarillo') AND Cliente(rif:r, nombre:nc)}**

- **QBE=Query By Example**
- Lenguaje relacional gráfico, amigable para sus usuarios, basado en el cálculo relacional de dominios.
- Idea base: formular la consulta mediante un ejemplo de la respuesta
- Es el único lenguaje que existe en el mercado que guarda cierta relación con el CRD (aunque se creó antes que la especificación formal de CRD)

Zloff, 1977

- Para consultar se invocan los esquemas de las tablas y estas se despliegan de forma gráfica en la pantalla. Luego se pone el ratón en la o las columnas deseadas y se indica la operación a realizar.

Producto	codPro	nombre	cantidad	color	ubicacion_almacen

- Las variables se indican colocando el símbolo de subrayado como prefijo o subrayándolas (s o s).
- Las constantes se colocan directamente en la columna que se desea precedida del operador de comparación que necesita.
- Toda variable desplegable está cuantificada ímplicitamente por el cuantificador existencial.
- Todas las operaciones deben tener como sufijo un punto.

# Operaciones en QBE

P.	Desplegar o seleccionar
ALL	Cuantificador universal
CNT.	Contar
AVG.	Promedio
SUM.	Sumar o acumular
MIN.	Calcular el valor mínimo
MAX.	Calcular el valor máximo
G.	Agrupar tuplas
AO.	Ordenar ascendentemente
DO.	Ordenar descendientemente
$\neg$	Negación lógica
OR.	Disyunción lógica
AND.	Conjunción lógica

# Operaciones en QBE

Las funciones CNT, AVG, SUM, MIN y MAX deben aplicarse a variables precedidas con ALL.

Si no se desea eliminar las tuplas duplicadas en una proyección, se coloca P.ALL.\_v

# Consultas en QBE

1. Listado con nombre y color de cada producto en el almacén

Producto	codPro	nombre	cantidad	color	ubicacion_almacen
		P.		P.	

2. Listado con nombre y cantidad en existencia de cada producto de color azul en el almacén

Producto	codPro	nombre	cantidad	color	ubicacion_almacen
		P.	P.	azul	

# Consultas en QBE

3. ¿Cuál es el rif del proveedor de cada producto en el almacén?

Producto	codPro	nombre	cantidad	color	ubicacion_almacen
	_pr	P.			

Compra	codigo	fecha	producto	cantidad	monto	rif_proveedor
			_pr			P.

# Consultas en QBE

4. ¿Cuáles son los clientes que han comprado al menos un producto de color amarillo?

Producto	codPro	nombre	cantidad	color	ubicacion_almacen
	_pr	P.		amarillo	

Venta	codigo	fecha	rif_cliente	producto	cantidad	monto
			_rc	_pr		

Cliente	rif	nombre	direccion	ciudad	telefono	celular
	_rc	P.				

# Consultas en QBE

5. Haga un listado de los productos de color verde ordenados descendientemente por nombre de producto

Producto	codPro	nombre	cantidad	color	ubicacion_almacen
		P.ALL.DO._n		verde	

6. Teniendo la relación *Prelacion*(codMateria, *codMateriaPrelacion*) liste las materias que prelan al Proyecto de grado

Prelacion	codMateria	codMateriaPrelacion
	'Proyecto de grado'	_m
	_m	P._mp

# Manipulación de datos en QBE

## Inserción

Insertar un producto nuevo de 10 pares de zapatos negros en el almacén de la Av. 4

Producto	codPro	nombre	cantidad	color	ubicacion_almacen
I.	MAX.ALL._n+1	zapatos	10	negro	Av. 4

## Eliminación

Elimine todos los productos de color gris

Producto	codPro	nombre	cantidad	color	ubicacion_almacen
D.				gris	

# Manipulación de datos en QBE

## Actualización

Añadir 100 en cantidad de la existencia de todos los productos de color verde

Producto	codPro	nombre	cantidad	color	ubicacion_almacén
U.	_p		_c	verde	
	_p		_c+100		