

# Les langages relationnels

- **Algèbre relationnelle** (procédural): Utiliser des opérateurs algébriques pour manipuler des relations (+, -, \*, projection, sélection, jointure)
- **Calcul relationnel** (non procédural): Prédicat utilise une variable de tuple (langages  $\alpha$ , QUEL)
- **Langages de "transformation"** (non procédural)  
    **SQUARE, SEQUEL, SQL, ORACLE, INGRES**  
    **SQL/DS** constitue une norme relationnelle
- **Langages graphiques** (non procédural)
  - QBE (PARADOX), QPE
  - PICT
  - IBS (U de M)



# Langage SQL

- **Langage normalisé par ANSI pour la manipulation des bases de données relationnelles**
- **Standard pour l'échange entre systèmes de BD**
- **Sous langage inclus dans d'autres langages**
- **Résultat est une relation**
  
- **clause SELECT (ne pas confondre avec la sélection)**
- **Projection  
SELECT fnom, ville, ctg**
- **FROM F**

- **SELECT ctg**
- **FROM F**  
va donner une liste avec des éléments en double ---- »
- **SELECT DISTINCT ctg**
- **FROM F**

### Sélection

- **SELECT fnom, ctg**
- **FROM F**
- **WHERE ville = Montreal**
- **Une des structures fondamentales de SQL**
- **SELECT \***
- **FROM P**
- **WHERE couleur = jaune**
- **va donner toutes les colonnes**

- **Plusieurs conditions**
- **SELECT Pnom, Pno**
- **FROM P**
- **WHERE poids > 1500 AND couleur = rouge**
  
- **Ensemble de valeurs**
- **SELECT nom**
- **FROM F**
- **WHERE ville IN [Montreal, Toronto]**  
on peut avoir aussi
- **WHERE ville NOT IN [Montreal, Toronto]**

- **Tri**
- **SELECT Pnom, couleur**
- **FROM P**
- **WHERE poids > 1500**
- **ORDER BY couleur**
- **ou encore**
- **ORDER BY Pnom ASC, poids DESC**
  
- **Fonctions Built-in**
- **COUNT, SUM, AVG, MAX, MIN**
- **SELECT COUNT(\*)**
- **FROM F**

- **On ne peut pas mélanger fonctions et attributs comme:**
- **SELECT couleur, COUNT(\*) (excepté GROUP BY)**
- **SELECT COUNT(poids) FROM P**
- **SELECT COUNT(DISTINCT poids) FROM P**
- **Groupage**
- **SELECT poids, COUNT (\*)**
- **GROUP BY poids**
- **permet de grouper les attributs selon les mêmes valeurs**
- **SELECT couleur, COUNT(\*)**
- **FROM P**
- **GROUP BY couleur**
- **HAVING COUNT (\*) > 3**



- **SELECT nom**
- **FROM F**
- **WHERE Fno IN**
- **(SELECT Fno**
- **FROM F**
- **WHERE ville = Montreal)**
- **Jointure**
- **SELECT Pno, Pnom, couleur**
- **FROM FP, P**
- **WHERE FP.Pno= P.Pno**
- **ou encore avec : AND FP.Qté=250**
- **AND P.Poids>100**

- Pour trois tables:
- **SELECT Fno, Pno, Pnom, Qté**
- **FROM F, FP, P**
- **WHERE F.Fno=FP.Fno**
- **AND FP.Pno= P.Pno**
- **AND F.Fnom= Rona**
- Une jointure peut être utilisée pour réaliser plusieurs sous-requêtes
- **SELECT ville**
- **FROM F, FP**
- **WHERE F.Fno = FP.Fno**
- **AND FP.Fno = 1400**
- **Quelle est la ville des fournisseurs qui fournissent des pièces rouges**
- ...

- **Différences entre sous-requêtes et jointures**
- **No des pièces fournies par des fournisseurs de catégorie diff. de 1**
- **Par une sous requête:**
- **SELECT DISTINCT Pno**
- **FROM FP**
- **WHERE Fno IN (SELECT Fno**
- **FROM F**
- **WHERE catégorie NOT= 1)**
- **Par une jointure:**
- **SELECT DISTINCT Pno**
- **FROM F, FP**
- **WHERE FP.Fno = F. Fno**
- **AND F.catégorie NOT = 1**
- **mais si on veut les no de pièces et les catégories des fournisseurs on doit utiliser une jointure (les résultats viennent de deux tables)**

- Opérateurs EXISTS et NO EXISTS (opérateurs logiques)
- No des fournisseurs qui fournissent plus d'une pièce ?
- **SELECT DISTINCT Fno**
- **FROM FP A**
- **WHERE EXISTS (SELECT \***
- **FROM FP B**
- **Where A.Fno = B.Fno AND A.Pno NOT= B.Pno**
- Noms des fournisseurs qui fournissent toutes les pièces ?
- **SELECT Fnom**
- **FROM F** si non le fournisseur fourni toutes les pieces.
- **WHERE NOT EXISTS (SELECT \***
- **pièces qui n'auraient pas été fournies FROM FP**
- **WHERE NOT EXISTS (SELECT \***
- **FROM P**
- **pièces fournies-----** **WHERE P.Pno = FP.Pno**
- **AND FP.Fno = F.Fno))**

- **Insertion**
- **INSERT INTO FP**
- **values (1400,500,15)**
- **insertion de tout (sinon 1400, 500) pas autorisé**
- **DELETE FP WHERE F.Fno= 1400**
- **DELETE F WHERE FP.Fno IN (SELECT Fno**  
**FROM F WHERE ville**  
**= Montreal**
- **UPDATE FP**
- **SET Qte= 500 WHERE Fno = 1400**
- **contrôle d 'intégrité :**
- **UPDATE FP SET Pno = 400 WHERE Pno = 450**
- **UPDATE P SET Pno = 400 WHERE Pno = 450**

# Le Langage QBE

- **Nom et numéros des fournisseurs de Montréal avec une catégorie >10**

<b>F</b>	<b>#F</b>	<b>NomF</b>	<b>Ctg</b>	<b>Ville</b>
	<b>P.</b>	<b>P.</b>	<b>&gt; 10</b>	<b>Montréal</b>

==>Le "AND" est implicite, comme les opérateurs existentiels

- **Numéro des fournisseurs et ctg des fournisseurs situés à Montréal *ou* de ctg > 20?**

<b>F</b>	<b>#F</b>	<b>NomF</b>	<b>Ctg</b>	<b>Ville</b>
	<b>P.</b>		<b>P.</b>	<b>Montréal</b>
	<b>P.</b>		<b>P. &gt; 20</b>	

==> doivent être spécifiés dans deux lignes différentes

- **Boîte conditionnelle**

<b>F</b>	<b>#F</b>	<b>NomF</b>	<b>Ctg</b>	<b>Ville</b>
<b>P.</b>			<b>P.x</b>	<b>y</b>

<b>CONDITIONS</b>
<b>y = Montréal OR x &gt; 20</b>

- **Pièces dont le poids est entre 15 et 20 inclusivement?**

<b>P</b>	<b>#P</b>	<b>Pnom</b>	<b>Couleur</b>	<b>Poids</b>	<b>Poids</b>
<b>P.</b>	<b>P.x</b>			<b>&gt; = 15</b>	<b>&lt; = 20</b>

- **Expressions**
- **Donner pour chaque pièce le numéro et le poids (en grammes) (init: pounds)**

P	#P	Poids
P.	<u>P<sub>x</sub></u>	P. 'Poids en g=' <u>P<sub>x</sub></u> *654

- **Donner numéro de fournisseur et numéro de pièce tels que chacun des deux soit dans la même ville**

<b>F</b>	<b>#F</b>	<b>Ville</b>
	<u><b>F<sub>x</sub></b></u>	<u><b>V<sub>x</sub></b></u>

<b>P</b>	<b>#P</b>	<b>Ville</b>
	<u><b>P<sub>x</sub></b></u>	<u><b>V<sub>x</sub></b></u>

<b>P.</b>	<u><b>F<sub>x</sub></b></u>	<u><b>P<sub>x</sub></b></u>

- **Union**
- **Numéros de pièces qui ont soit un poids > a 100 ou sont fournies par F2, ou les 2.**

<b>P</b>	<b>#P</b>	<b>Poids</b>
	<b><u>Px</u></b>	<b>&gt; 100</b>

<b>FP</b>	<b>#F</b>	<b>#P</b>
	<b>F2</b>	<b><u>Py</u></b>

<b>P.</b>	<b><u>Py</u></b>
<b>P.</b>	<b><u>Px</u></b>

- **Groupage**
- **Pour chaque pièce fournie en une quantité moyenne > 200, donner le numéro et la quantité totale.**

<b>FP</b>	<b>#F</b>	<b>#P</b>	<b>Qté</b>
		<b>P.G.*</b>	<b>P. sum.<u>Qx</u></b>

<b>Condition</b>
<b>AVG.<u>Qx</u> &gt; 200</b>

\* analogue au **GROUP by SQL**

- **Jointure d'une table sur elle-même**
- **Donner tous les couples de numéros de fournisseurs tels que deux fournisseurs soient dans la même ville.**

<b>F</b>	<b>#F</b>	<b>ville</b>
	<b>F<sub>x</sub></b>	<b>V<sub>x</sub></b>
	<b>F<sub>y</sub></b>	<b>V<sub>x</sub></b>

<b>F</b>	<b>#F</b>	<b>#F</b>
<b>P.</b>	<b>F<sub>x</sub></b>	<b>F<sub>y</sub></b>

- **Noms des fournisseurs qui fournissent la pièce P2**

<b>F</b>	<b>#F</b>	<b>NOMF</b>
	<b>F<sub>x</sub></b>	<b>P.</b>

<b>FP</b>	<b>#F</b>	<b>#P</b>
	<b>F<sub>x</sub></b>	<b>P2</b>

- Noms des fournisseurs qui fournissent au moins une pièce fournie par F2?**

<b>F</b>	<b>#F</b>	<b>NOMF</b>
	<b><u>Fx</u></b>	<b>P.NOMX</b> (optionnel)

<b>FP</b>	<b>#F</b>	<b>#P</b>
	<b><u>FX</u></b>	<b><u>PX</u></b>
	<b>F2</b>	<b><u>PX</u></b>

- Numéros des fournisseurs et catégorie des fournisseurs de Montréal, en ordre croissant des #F et en ordre décroissant selon les catégories.**

<b>F</b>	<b>#F</b>	<b>NOM</b>	<b>catégorie</b>	<b>ville</b>
	<b>P.AO(2)</b>		<b>P.DO(1)</b>	<b>Montréal</b>

**ordre des opérations**

**On peut spécifier aussi la commande dans F (impression de tous les attributs)**

<b>F</b>	<b>#F</b>	<b>NOMF</b>	<b>CTG</b>	<b>VILLE</b>
<b>Domain I.</b>	<b>#F</b>	<b>NOMF</b>	<b>CTG</b>	<b>VILLE</b>
<b>TYPE I.</b>	<b>CHAR(5)</b>	<b>CHAR(15)</b>	<b>FIXED</b>	<b>CHAR(20)</b>
<b>KEY</b>	<b>Y</b>	<b>U.N.</b>	<b>U.N.</b>	<b>U.N.</b>

- **Création d'un cliché (Snapshot)**

- **no. de pièce et nom des villes où cette pièce est fournie.**

<b>F</b>	<b>#F</b>	<b>NOMF</b>	<b>CTG</b>	<b>VILLE</b>
--->	<u><b>Fx</b></u>			<u><b>FV</b></u>
--->	<u><b>15</b></u>			<u><b>10</b></u>

<b>FP</b>	<b>#F</b>	<b>#P</b>	<b>QTE</b>
	<u><b>Fx</b></u>	<u><b>PX</b></u>	
	<u><b>15</b></u>	<u><b>20</b></u>	

<b>I. RESULT I.</b>	<b>#P</b>	<b>VILLE</b>
<b>P.I.</b>	<u><b>Px</b></u>	<u><b>FV</b></u>
	<u><b>20</b></u>	<u><b>10</b></u>

(autre forme d'expression)

Indépendante des autres relations  
**≠ VIEW**

- **Insertion**
- **Ajout d'une pièce**

<b>P</b>	<b>#P</b>	<b>NOMP</b>	<b>COULEUR</b>	<b>POIDS</b>
<b>I.</b>	<b>24</b>	<b>robinet</b>	<b>blanc</b>	<b>25</b>

- **Effacement**

<b>F</b>	<b>#F</b>	<b>NOMF</b>	<b>CTG</b>	<b>VILLE</b>
<b>D.(1)</b>				<b>TO</b>

- **Création d'une relation**

<b>I.FI.*</b>	<b>#F</b>	<b>NOMF</b>	<b>CTG</b>	<b>VILLE</b>
---------------	-----------	-------------	------------	--------------

entrée dans le dictionnaire des relations

entrée dans le dictionnaire des attributs

- **Doubler le poids de toutes les pièces de couleur rouge**

<b>P</b>	<b>#P</b>	<b>NOMP</b>	<b>COULEUR</b>	<b>POIDS</b>
<b>U.</b>	<b>P<sub>x</sub></b>	<b>P<sub>x</sub></b>	<b>rouge</b>	<b>P<sub>y</sub></b> <b>2*P<sub>y</sub></b>

- **Annuler la quantité de pièces fournies par les fournisseurs de Toronto.**

<b>FP</b>	<b>#F</b>	<b>#P</b>	<b>QTE</b>
<b>U.</b>	<b>F<sub>x</sub></b>		<b>0</b>

<b>F</b>	<b>#F</b>	<b>NOMF</b>	<b>CTG</b>	<b>VILLE</b>
	<b>F<sub>x</sub></b>			<b>TO</b>

**No. de pièces fournies par plus d'un fournisseur: donner le nombre de fournisseurs.**

<b>FP</b>	<b>#F</b>	<b>#P</b>	<b>QTE</b>
-----------	-----------	-----------	------------

**P. CNT.ALL.Fx P.G.Px**

**G.:** groupage

**CONDITIONS**  
**CNT.ALL.Fx > 1**

**a Mise à jour**

<b>P #P</b>	<b>NOMP</b>	<b>COULEUR</b>	<b>POIDS</b>
-------------	-------------	----------------	--------------

**U.P5** rouge

**P5** Px (ancien)

**U.P5** Px+ 10 (nouv.) augmenter le poids de 10

- **Quantité totale de pièces P2 fournies**

<b>FP</b>	<b>#F</b>	<b>#P</b>	<b>Qté</b>
		<b>P</b>	<b>P.SUM.ALL.<u>Q<sub>x</sub></u></b>

- **No des fournisseurs dont la catégorie est inférieure à la valeur maximum de la catégorie dans F. Donner aussi cette valeur maximum.**

<b>F</b>	<b>#F</b>	<b>NOMF</b>	<b>CTG</b>
	<b>P.<u>F<sub>x</sub></u></b>		<b><u>C<sub>x</sub></u></b>
		<b>P.MAX.ALL.<u>C<sub>y</sub></u></b>	

**CONDITIONS**

**C<sub>x</sub> < MAX.ALL.C<sub>y</sub>**

**P.MAX.ALL.C<sub>y</sub>**

**Groupage**

- **Pour chaque pièce fournie donner no. et la quantité totale**

<b>FP</b>	<b>F*</b>	<b>#P</b>	<b>Qté</b>
-----------	-----------	-----------	------------

**P.G.P<sub>x</sub>P.SUM.ALL.Q<sub>x</sub>**

**G: GROUP B<sub>y</sub>**

## FONCTIONS

CNT.ALL CNT.UNQ.ALL SUM.ALL SUM.UNQ.ALL

AVG.ALL AVG.UNQ.ALL MAX.ALL MIN.ALL

\*\* UNQ: Éimine les doubles

**Nb total de fournisseurs**

F	#F	NOMF	CTG	VILLE
---	----	------	-----	-------

P.CNT.ALL.F<sub>x</sub>

... qui fournissent des pièces

FP	#F	#P	Qté
----	----	----	-----

P.CNT.UNQ.ALL.F<sub>x</sub>

P.CNT.ALL.FP<sub>x</sub>

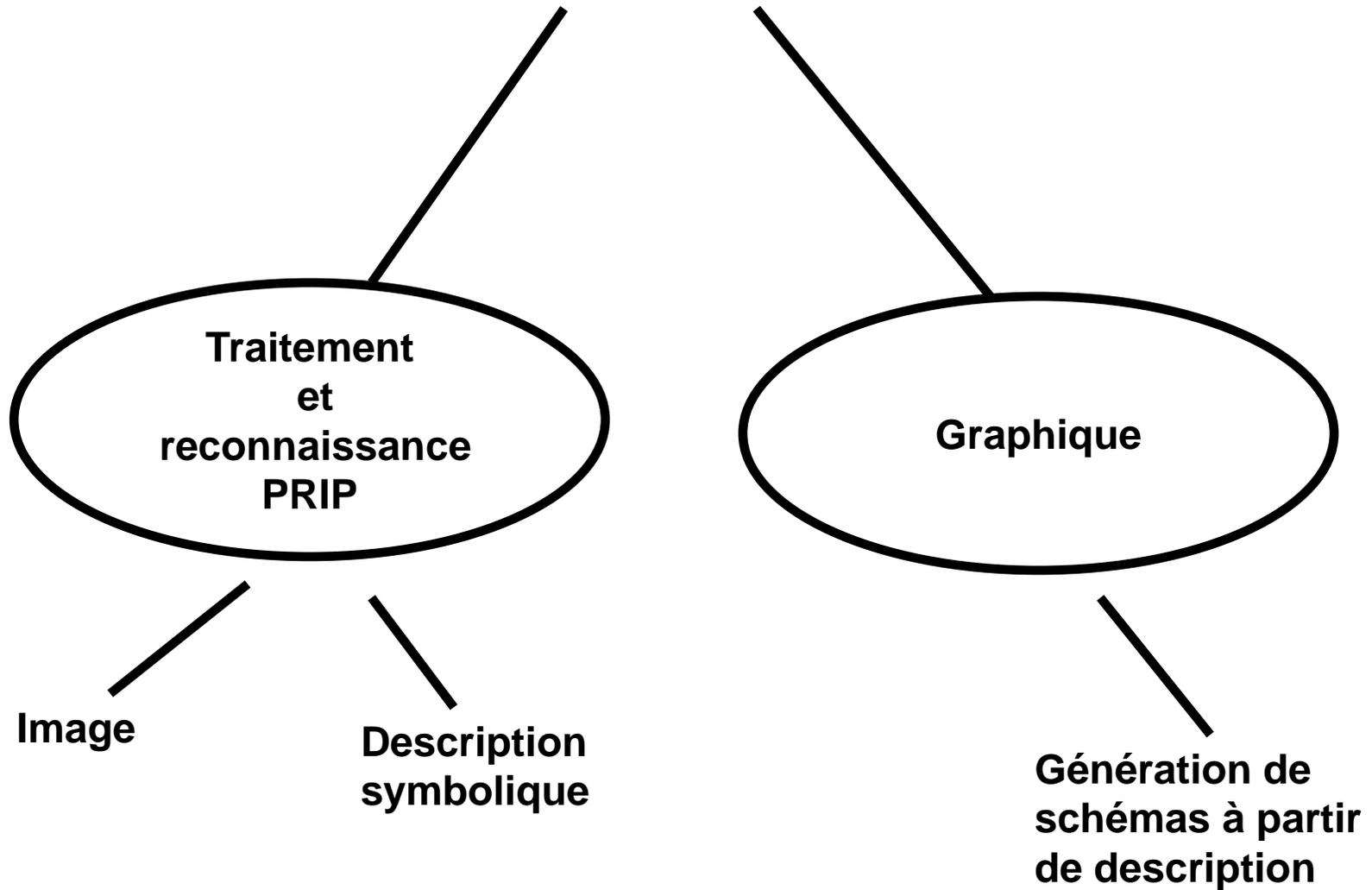
P2

P: Nb. de fournisseurs de la pièce P2



# Bases de données d'images

# Traitement d'images



# Importance de l'image

- **Contient une grande quantité d'informations**
- **Caractérise**
  - **Disposition spatiale**
  - **Apparence**
  - **Texture**
  - **Relief**
  - **Composition**
- **80% de l'acquisition de la connaissance humaine**
- **Sous ensemble symbolique : l'icône**

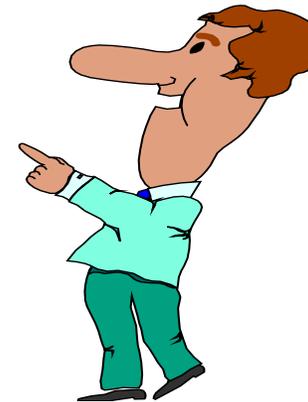
# Applications

- **Télédétection**

- Images de satellites
- Photos aériennes
- Traitement des informations géographiques, météos
- Cartographies

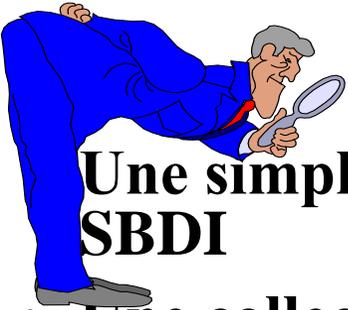
- **Médecine**

- Rayons X
- Tomographies
- Karyotypes
- Cellules, tissus



- **Industrie**
  - **Conceptions graphiques (CAD/CAM, prototypes, circuits VLSI)**
  - **Vision (3D), inspection, robotique, navigation**
- **Science**
  - **Chambres à bulles, astronomie**
  - **Ethnologie**
- **Sécurité**
  - **Empreintes digitales**
  - **Plaques de voitures**
  - **Accès protégés**

# Systeme de base de données d'images SBDI



**Une simple collection d'images sans système de gestion n'est PAS un SBDI**

- **Une collection d'images de types variés pour l'analyse et le traitement n'est PAS un SBDI car**
  - Ils dépendent des pgms d'application
  - Ne possèdent pas d'intégrité
  - Dépendent du stockage
- **On a recherché une extension des SGBD avec des fichiers d'images et des descripteurs d'images**

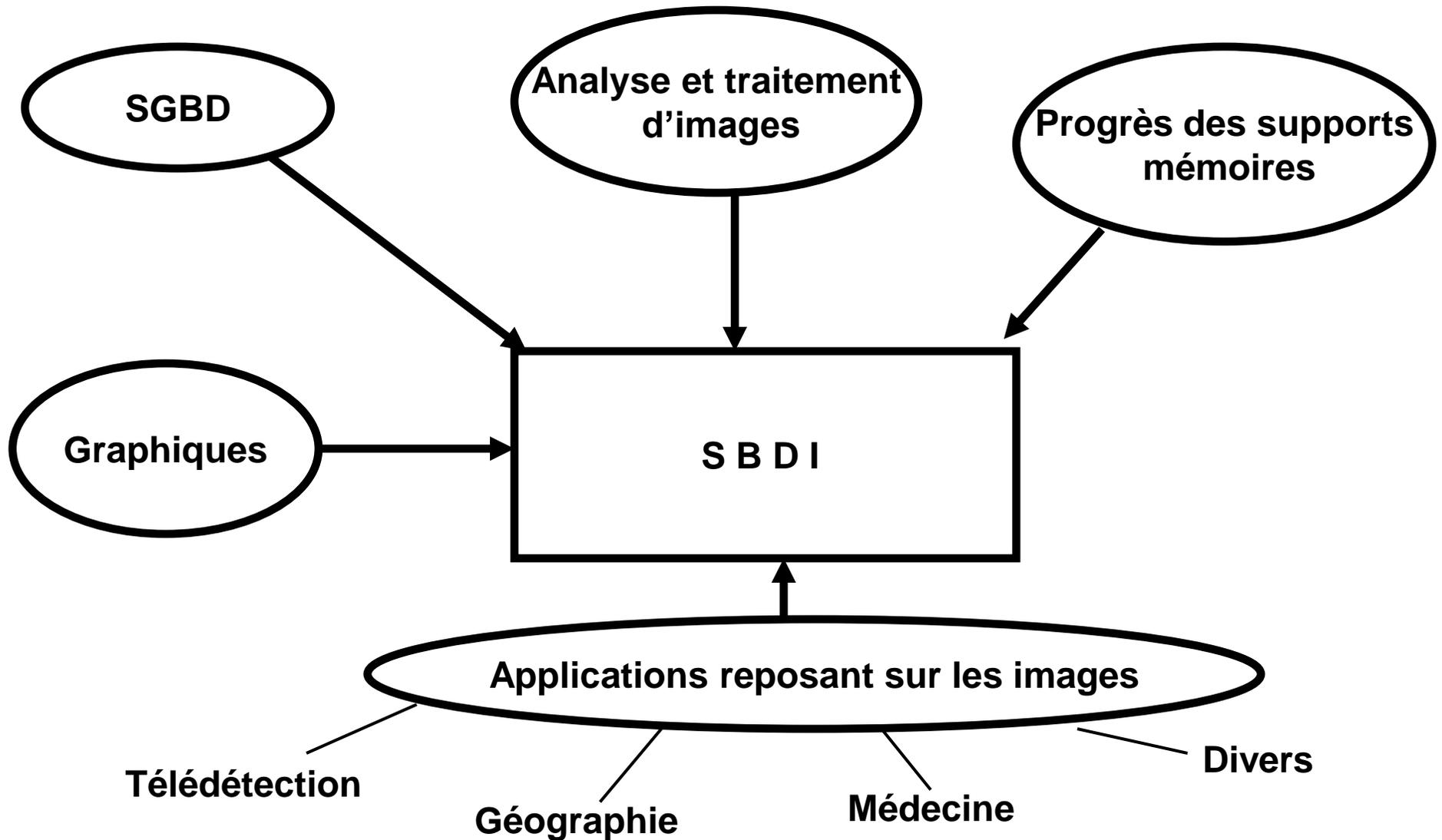
# Problèmes avec les SGBD actuels

- **Objets à représentation discrète**
  - Les images sont des objets continus
  - Les images ont des transformations spatiales
- **Dans les SGBD on a un ordre total; donc tri possible**
- **Dans les images on a un ordre partiel**
- **Le volume d'une image est important (min 64k)**
- **Le temps de traitement nécessite une architecture spéciale**

- **SBDI : système où une grande quantité d'images et d'informations associées sont intégrées (stockage et gestion) avec :**
  - **Intégrité**
  - **Indépendance des données**
  - **Accès et manipulation des images sans connaître la structure logique ou physique**



# Évolution du contexte



# Exemple : QPE

## Query by Pictorial Example

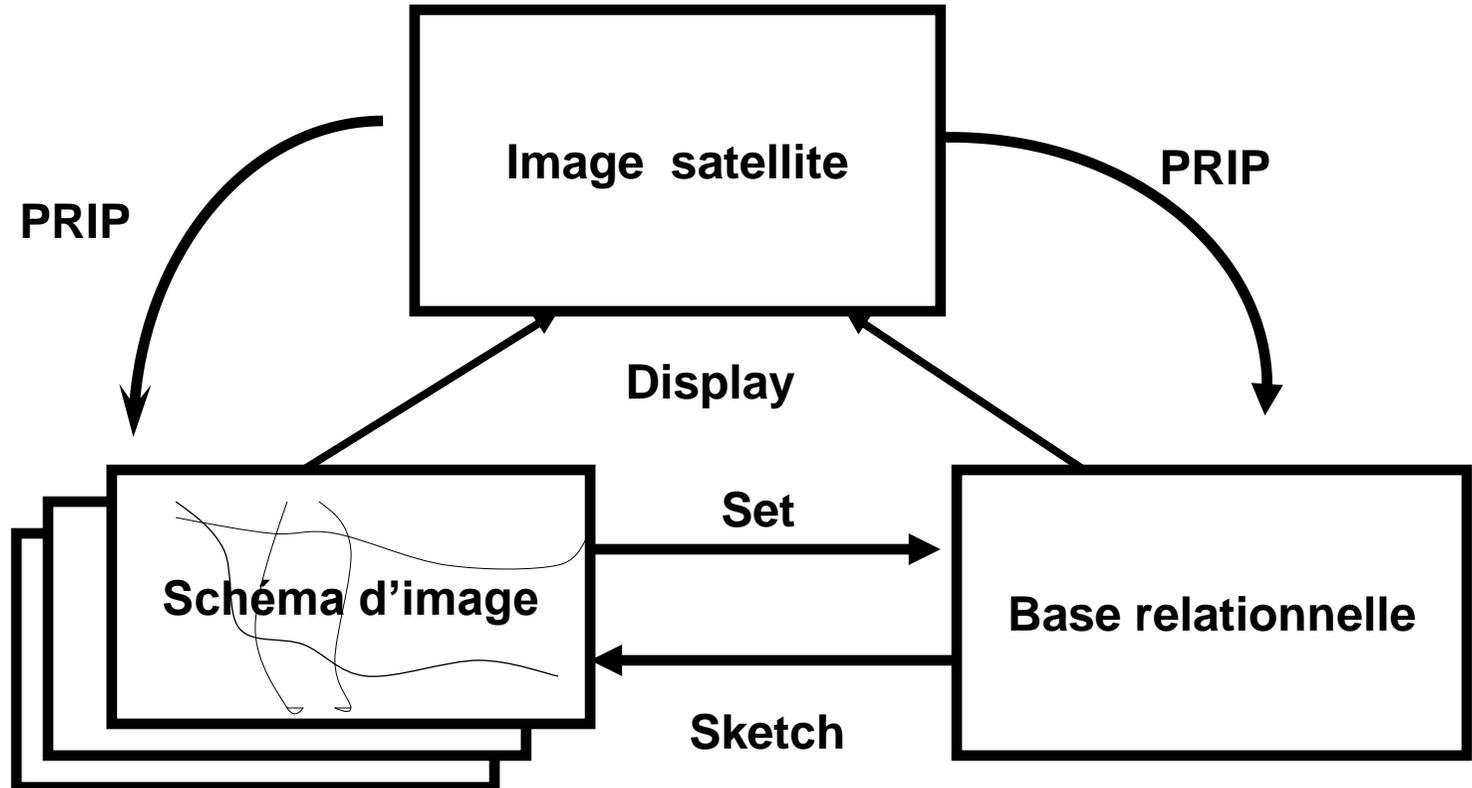
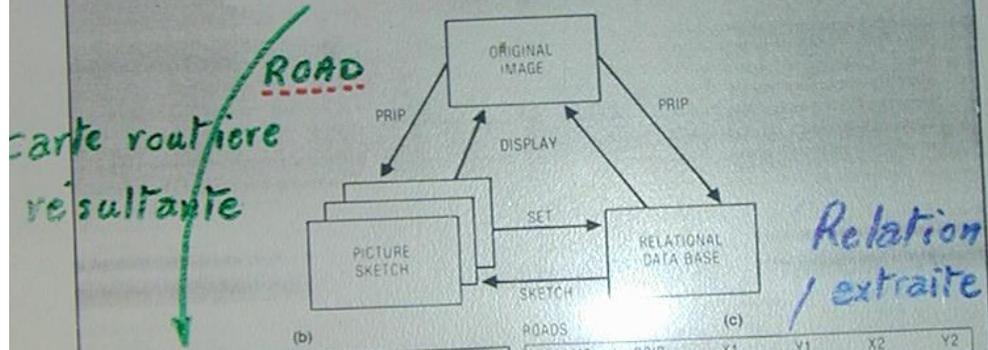
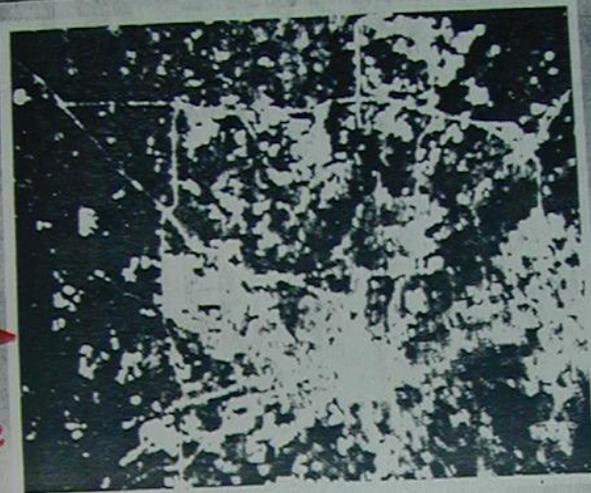
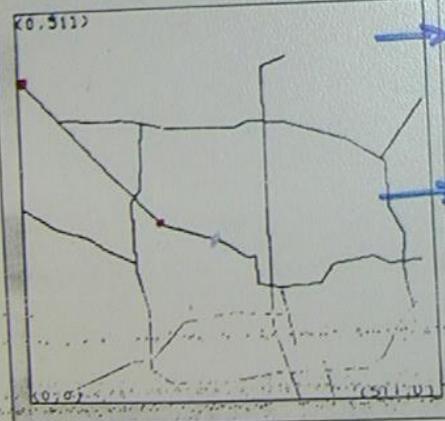


Image Landsat digitalisée



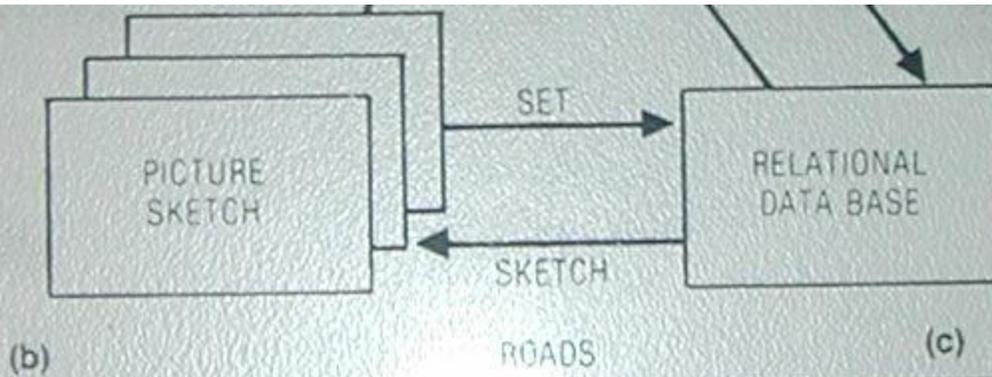
carte routiere resultante



FRAME	ROID	X1	Y1	X2	Y2
405	1	0	432	117	304
405	1	117	304	176	243
405	1	176	243	252	218
405	1	252	218	218	284
405	1	284	197	295	198
405	1	295	198	198	296
405	1	296	197	324	197
405	1	324	197	341	71
405	1	341	71	374	53
405	1	374	53	384	4
405	2	0	265	40	238
405	2	40	238	85	224
405	2	85	224	106	209
405	2	106	209	142	191
405	3	33	6	134	56
405	3	134	56	159	59
405	3	159	59	201	111
405	3	201	111	267	122

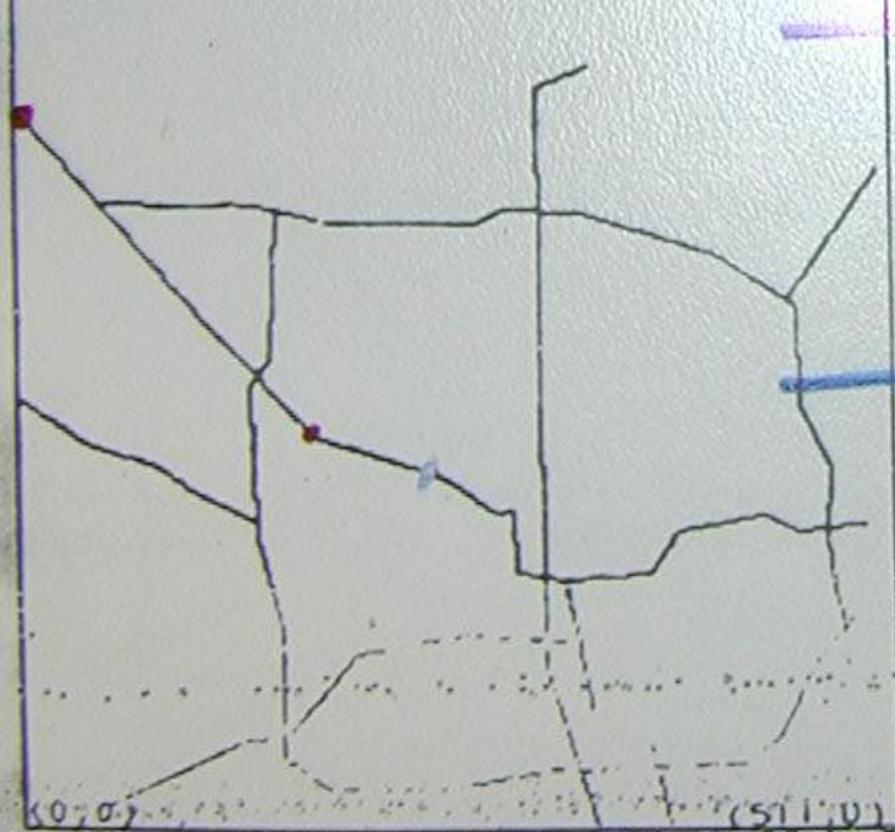
Figure 1. Data-flow diagram of IMAID.

resultante



Relation  
/ extraite

(0,511)



ROADS	NAME	ROID	X1	Y1	X2	Y2
	405	1	0	432	117	304
	405	1	117	304	176	243
	405	1	176	243	252	218
	405	1	252	218	284	197
	405	1	284	197	295	198
	405	1	295	198	296	161
	405	1	296	161	324	157
	405	1	324	157	341	71
	405	1	341	71	374	53
	405	1	374	53	384	4
	405	2	0	265	40	238
	405	2	40	238	85	224
	405	2	85	224	106	209
	405	2	106	209	142	191
	405	3	33	6	134	56
	405	3	134	56	159	59
	405	3	159	59	201	111
	405	3	201	111	267	122

# Principes

- **Extraction d'un croquis (ou schémas) à partir d'une image satellite selon une technique PRIP (reconnaissance de forme)**
- **Extraction de routes, villes, ponts, rivières,...par des algorithmes spécifiques**
- **Transformation du schéma en base relationnelle (set) par succession de segments de droite**
- **Inverse : lecture de la base pour reconstituer un schéma (sketch)**
- **Display : affiche l'image de la relation ou du schéma**

# Exemple

**ROADS**

Frame	ROID	X1	Y1	X2	Y2
405	1	0	432	117	304
405	1	117	304	176	243
405	1	176	243	252	218
405	1	252	218	284	197
405	1	284	197	285	198
405	1	285	198	296	161
405	2	0	265	40	238
405	2	40	238	85	224
405	2	85	224	106	209

# Modèle de relations

- **ROADS (Frame, ROID, X1, Y1, X2, Y2)**
- **RONAME (Frame, ROID, Name)**
- **POS (Frame, Xsize, Ysize, Xcen, Ycen, Loc)**
- **CITIES (Frame, CIID, X1, Y1, X2, Y2)**
- **CINAME (Frame, CIID, Name)**
- **Certaines relations comme ROADS, POS, CITIES sont construites automatiquement par application de l'opérateur Set sur l'image.**

# Capacité de QPE

- **Constructions de dessins à partir d'intersection, de négation de dessins**
- **Calcul de surface, longueur, périmètre**
- **Conversion : Image ----Schéma----relation au moyen d'opérateurs**
- **Inversement : attributs ---schéma --- image**
- **Requêtes à partir d'exemples**
- **Manipulation de recherche par similarité**

**Q1 : Trouver le nom des routes qui apparaissent dans le même cliché (Frame) que la ville de Laval ?**

**ROADS**

CINAME	Frame	CID	Name
	<u>X</u>		Laval

RONAME	Frame	ROID	Name
	<u>X</u>		P. Name

**On dispose aussi d'opérateurs d'attributs**

- longueur d'une ligne                      **Length-L**
- Périmètre d'une région                      **Length- R**
- Surface    **Area-R**
- Distance entre points                      **Length-PP**

**Q2 : Appliquer le traitement ROAD à l'image 301 et placer le résultat dans la relation ROADS**

POS	Frame	Xsize	Ysize	XCen	YCen	LOC
ROAD	301					

ROADS	Frame	ROID	X1	Y1	X2	Y2
I.(ROAD)	301					

**Q3 : Faire le schéma de la relation ROADS**

ROADS	Frame	ROID	X1	Y1	X2	Y2
S.	301					

#### Q4 : Afficher l'image de la ville de Laval

CINAME	Frame	CID	Name
	<u>X</u>		Laval

POS	Frame	Xsize	Ysize	XCen	YCen	LOC
DIS.	<u>X</u>					

#### Q5 : Trouver le nom des routes pointées sur l'écran

ROADS	Frame	ROID	X1	Y1	X2	Y2
	*	<u>X</u>	@	@	@	@

RONAME	Frame	ROID	Name
		<u>X</u>	P. Name