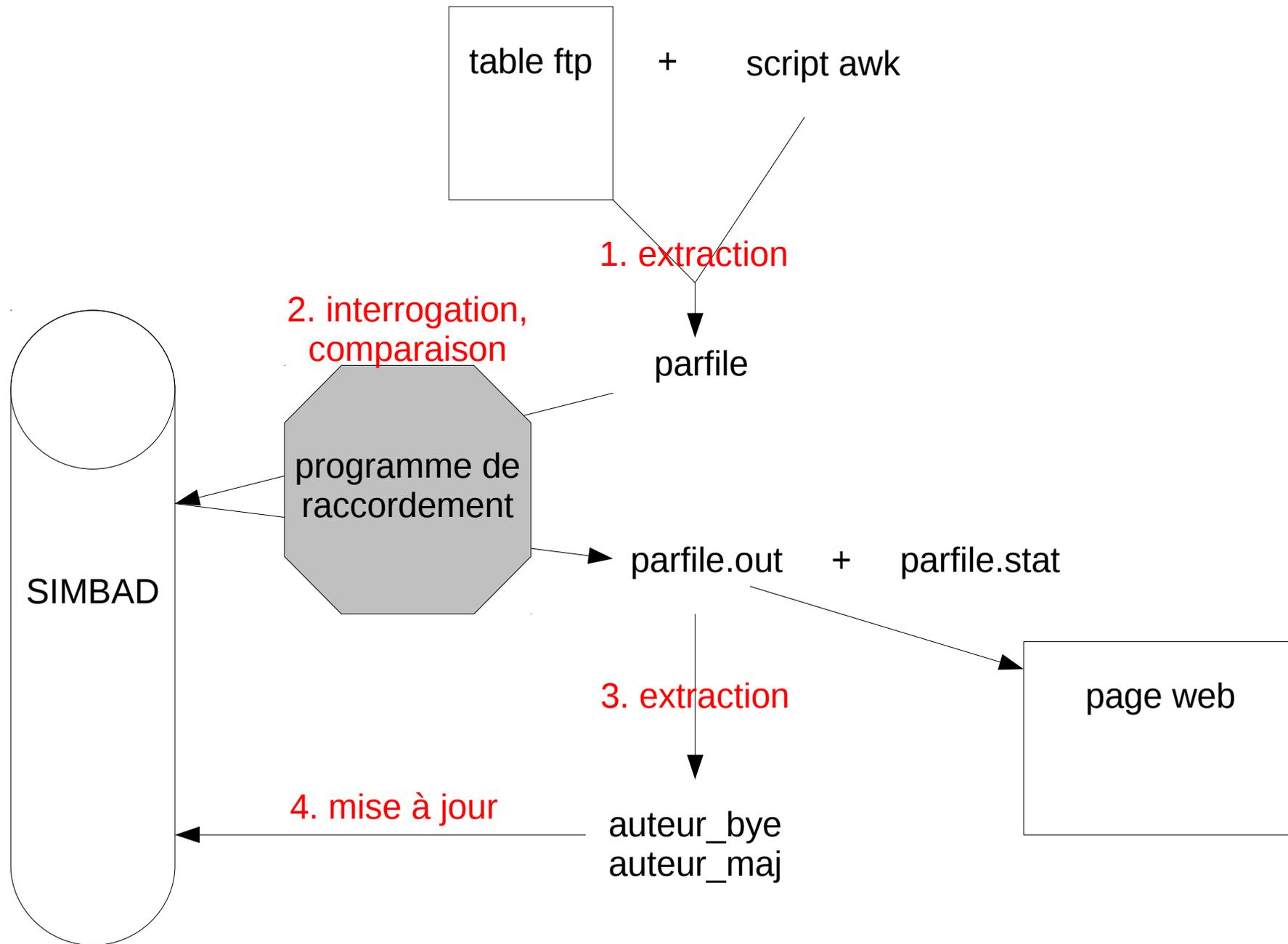


# Le nouveau programme de raccordement pour SIMBAD

Anais Oberto, Catherine Brunet  
janvier 2014

# Processus d'intégration des grandes tables dans SIMBAD



# Etapes du programme

et/ou

1. Recherche d'objets dans SIMBAD

a. par identificateur

b. par coordonnées



2. Evaluation de chaque objet trouvé:  
comparaison avec l'objet donné en  
entrée

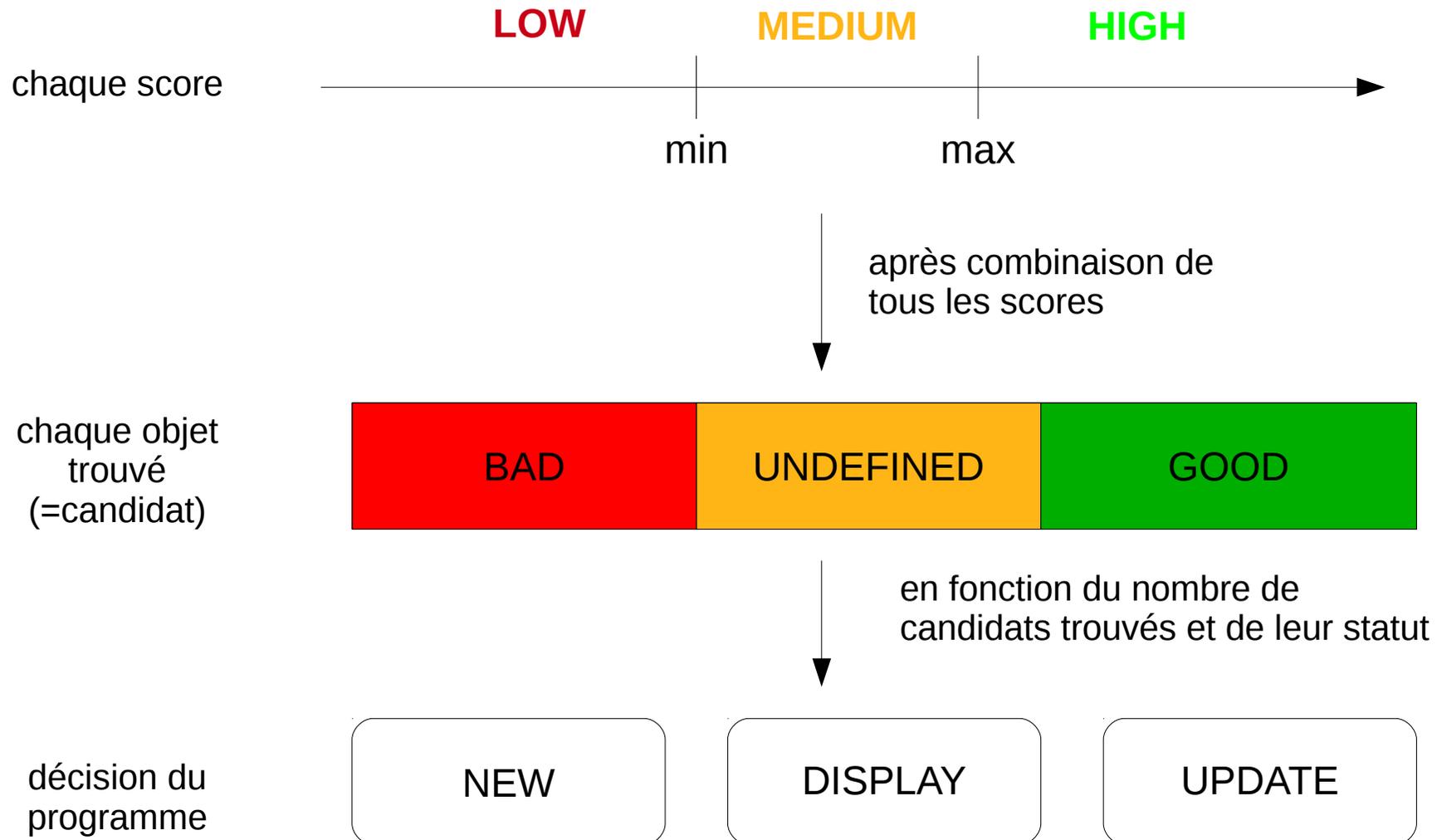
2a. 1. Mise en place de différents scores,  
un par paramètre comparable  
2. Calcul de chacun de ces scores  
3. Classement de chacun de ces  
scores selon les valeurs min et max  
données: **LOW**, **MEDIUM** ou **HIGH**

2b. Combinaison de ces scores pour donner  
la valeur de chaque candidat: BAD,  
UNDEFINED ou GOOD pour la cross-  
identification



3. Décision finale pour l'édition ou non de commandes de mise à jour

# Le vocabulaire utilisé



# 1. Recherche des objets dans SIMBAD

- Recherche par identificateur
- Si aucun objet trouvé par identificateur, recherche par coordonnées
- Le rayon de recherche est fonction de la précision des coordonnées en entrée (comme dans l'ancien raccord)

1° → 3600"

0,1° → 1440"

1' → 300"

0,1' → 36"

1" → 6"

0,1" → 2"

0,01" → 1,2"

1mas → 0,6"

puis borné entre 30" et 3600".

- Pour agir sur le rayon de recherche:

*.search 5"*

*.search 300mas*

*.search 2'*

*.search \*2*

- Pour forcer la recherche par coordonnées dans tous les cas:

*.search coo*

- Pour interdire la recherche par coordonnées:

*.search nocoo*

## 2a1. Mise en place des scores à calculer

- Presque chaque paramètre en entrée va chercher à être comparé et générer un score.

%J	→ COO
%C	→ OT
%V	→ V
%M	→ M
%G	→ GDIM
%P	→ PM
%X	→ X

- Dans l'impossibilité de comparer un paramètre, le score correspondant est ignoré.
- Le cas échéant, deux autres scores peuvent être générés: ACRO (pour tester la présence/absence d'un acronyme) et B (pour tester la présence/absence d'un bibcode).

- Pour empêcher le calcul d'un score:

*.IGNORE COO*

*.IGNORE M*

*.IGNORE V*

- Pour tenir compte des scores ACRO et B:

*.ACRO+1 2MASS,2MASX*

*.ACRO-1 IRAS*

*.ACRO++1 PMN*

*.B+1 2012A&A...537A.147H*

*.B-1 2013PASJ...65...65T*

(voir plus loin pour la définition de ces options)

# Ecriture des abrégés de champs

		parfiles (derrière % ou !)	scores de Raccord	commandes de mise à jour dans SIMBAD
type d'objet	principal	<b>C</b>	<b>OT</b>	<b>c o</b>
	secondaire	<b>CL</b>		<b>a o</b>
identificateur	le premier	<b>I.O</b>		o/a <b>i</b> ou <b>id</b>
	les suivants	<b>I</b>		
	pour le raw-id	<b>I.R</b>		
coordonnées		<b>J</b>	<b>COO</b>	a/c <b>c</b> ou <b>coo</b>
vitesse		<b>V</b>	<b>V</b>	a/c <b>v</b>
magnitude		<b>F</b> ou <b>M</b>	<b>M</b>	a/c <b>m</b>
dimensions		<b>G</b>	<b>GDIM</b>	a/c <b>g</b>
mouvement propre		<b>P</b>	<b>PM</b>	a/c <b>p</b>
parallaxe		<b>X</b>	<b>X</b>	a/c <b>x</b>
type spectral		<b>S</b>		a/c <b>s</b>
type morphologique		<b>T</b>		a/c <b>t</b>
référence bibliographique		<b>B</b>	<b>B</b>	a <b>r</b>
hiérarchie	parent	<b>H.p</b>		a <b>parent</b>
	enfant	<b>H.c</b>		a <b>child</b>
note	courte	<b>NC</b>		a/c <b>nc</b>
	longue	<b>NL</b>		a/c <b>nl</b>
	de service	<b>NS</b>		a/c <b>ns</b>

 attention! différences  
 sans objet

## 2a2. Calcul des scores: type d'objet

- Le score du type d'objet se base sur un tableau de comparaison des types entre eux utilisé dans SIMBAD.
- La comparaison n'est faite qu'entre types d'objets principaux.
- Si les deux types sont compatibles, il prend la valeur de 1;

Si les deux types ne sont pas compatibles, il prend la valeur de -1;

Si les deux types sont neutres, il prend la valeur de 0.

- Pour favoriser ou défavoriser certains types d'objets:

*.OT+2 LM\*,\**

*.OT-1 CIG,GrG*

## 2a2. Calcul des scores : coordonnées

- Selon la formule:

$$1.5 - 0.5 \times (|\text{valeurSimbad} - \text{valeurTable}| / \sqrt{(\sigma_{\text{Simbad}}^2 + \sigma_{\text{Table}}^2)})$$

- Les limites du scores sont définies par la formule: de moins l'infini à +1.5
- Le sigma est fonction de l'erreur donnée; sinon il est estimé d'après la lettre de qualité:
  - E 10 000 mas
  - D 3 000 mas
  - C 1 000 mas
  - B 300 mas
  - A 100 mas
- Si une taille est donnée, le sigma est égal au minimum au dixième du grand axe.

- Pour borner le sigma, à la fois pour l'objet en entrée et pour celui de SIMBAD, toutes lettres confondues (agit aussi sur les erreurs données entre crochets):

*.SIGMA COO 300,1000*

## 2a2. Calcul des scores : vitesse

- Selon la formule:

$$1.5 - 0.5 \times (|\text{valeurSimbad} - \text{valeurTable}| / \sqrt{(\sigma_{\text{Simbad}}^2 + \sigma_{\text{Table}}^2)})$$

- Le sigma est systématiquement égal au dixième de la valeur, qu'il y ait ou non une erreur donnée. Ce qui donne:

$$1.5 - 0.5 \times (|\text{valeurSimbad} - \text{valeurTable}| / \sqrt{((\text{valeurSimbad}/10)^2 + (\text{valeurTable}/10)^2)})$$

- Le résultat de la formule est borné entre -1 et +1.
- Si une des deux qualités est E ou si un des deux redshifts est déclaré photométrique, aucune comparaison n'est faite.

## 2a2. Calcul des scores : magnitude

- Selon la formule:

$$1.5 - 0.5 \times (|\text{valeurSimbad} - \text{valeurTable}| / \sqrt{(\sigma_{\text{Simbad}}^2 + \sigma_{\text{Table}}^2)})$$

- Le sigma est une valeur fixe de 0,3 pour les magnitudes en entrée et celles de SIMBAD, qu'il y ait ou non une erreur donnée entre crochets. Ce qui donne:

$$1.5 - 0.5 \times (|\text{valeurSimbad} - \text{valeurTable}| / \sqrt{(0.3^2 + 0.3^2)})$$

- Le résultat de la formule est borné entre -1 et +1.
- Quand plusieurs magnitudes sont comparées, un score moyen est calculé à partir des différents scores obtenus.
- Si une des deux qualités est E, aucune comparaison n'est faite.

## 2a2. Calcul des scores : acronyme

- Par défaut, le score ACRO n'existe pas. Il existe dès qu'on utilise l'option *.ACRO+*, *.ACRO-* ou *.ACRO++*.
  - Il devient alors un score comme les autres qui va entrer en jeu dans la combinaison des scores.
  - Il ne prend que des valeurs entières.
- Pour favoriser ou défavoriser certains acronymes:
    - .ACRO+1 2MASS,2MASX*  
(augmente le score ACRO de 1 pour chaque acronyme 2MASS ou 2MASX présent dans l'objet SIMBAD; si un identificateur 2MASS ou 2MASX est donné en entrée, le score ACRO ne sera pas augmenté pour un identificateur différent trouvé avec le même acronyme)
    - .ACRO-1 IRAS*  
(diminue le score ACRO de 1 pour chaque acronyme IRAS présent dans l'objet SIMBAD; si un identificateur IRAS est donné en entrée, le score ACRO ne sera pas diminué si le même identificateur IRAS est trouvé)
    - .ACRO++1 PMN*  
(augmente le score ACRO de 1 pour chaque acronyme PMN présent dans l'objet SIMBAD<sub>1,2</sub> dans tous les cas)

## 2a2. Calcul des scores : bibcode

- Par défaut, le score B n'existe pas. Il existe dès qu'on utilise l'option .B+ ou .B-.
  - Il devient alors un score comme les autres qui va entrer en jeu dans la combinaison des scores.
  - Il ne prend que des valeurs entières.
- Pour favoriser ou défavoriser certaines références:  
*.B+1 2012A&A...537A.147H*  
(augmente le score B de 1 chaque fois que ce bibcode est présent dans la bibliographie de l'objet SIMBAD)  
*.B-1 2013PASJ...65...65T*  
(diminue le score B de 1 chaque fois que ce bibcode est présent dans la bibliographie de l'objet SIMBAD)

## 2a3. Classement des scores

- A chaque score sont attribués un minimum et un maximum.

Au dessous de min: le score est dit **LOW** (score a priori pour un candidat BAD).

Au dessus de max: le score est dit **HIGH** (score a priori pour un candidat GOOD).

Entre les deux, le score est dit **MEDIUM**.



- Valeurs par défaut:

.OT -1,1

.COO 0,1

.V 0,0

.M 0,0

(.ACRO -1,1)

(.B 0,1)

- Quand l'objet est trouvé par identificateur, tous les scores **MEDIUM** deviennent **HIGH**, car il arrive qu'on ne puisse pas départager nettement les scores M ou V pour tous les objets.

- Pour modifier les valeurs min et max de chaque score:

.COO 0.5,0.8

.V 0.2,0.2

.M -0.1,0.1

.ACRO 0,2

.B 0,2

## 2b. Evaluation du candidat

- Pour chaque objet trouvé, le programme va décider s'il est bon ou mauvais pour une cross-identification: c'est l'évaluation du candidat.

<u>BAD</u>	<u>UNDEFINED</u>	<u>GOOD</u>
------------	------------------	-------------

- Tous les scores calculés et non ignorés entrent en compte pour l'évaluation du candidat.
- Dès que OT est **LOW**, ou dès que COO est **LOW**, le candidat est étiqueté BAD.
- Pour qu'un candidat soit étiqueté GOOD, tous ses scores calculés doivent être **HIGH**.

- Pour écarter des candidats avec un score **LOW** (autre que OT ou COO):

*.TRUST V*

*.TRUST M*

*.TRUST ACRO*

- Pour écarter des candidats si un score n'est pas disponible (autre que OT ou COO):

*.EXPECT V*

*.EXPECT M*

- Rappel: pour ignorer un score:

*.IGNORE COO*

*.IGNORE V*

*.IGNORE M*

- Pour accepter tous les candidats trouvés par identificateur, même si *a priori* ils auraient été étiquetés BAD:

*.FORCE FROM ID*





# 3. Décision finale

- Si aucun objet n'est trouvé, ou si tous les objets trouvés par coordonnées sont BAD, un nouvel objet est créé (NEW).
- Si il y a un et un seul candidat GOOD (même si il y en a d'autres UNDEFINED), cet objet est mis à jour (UPDATE).
- Dans tous les autres cas, il n'y a pas d'édition de commandes de mise à jour et tout reste affiché (l'objet en entrée, les objets SIMBAD trouvés) (DISPLAY).

### 3. Décision finale, cas particuliers

- Si un candidat trouvé par identificateur est BAD, il reste en DISPLAY
- Si plusieurs candidats sont GOOD, on reste en DISPLAY, avec le message **Possible Merge**.
- Si la recherche par identificateur a donné plusieurs objets dans SIMBAD, on reste en DISPLAY, avec le message **Conflicting**, quelle que soit la valeur de chaque candidat.
- Si un objet en entrée contient des erreurs (identificateur mal écrit, ou tout autre champ mal écrit qui générerait un message d'erreur en mise à jour de SIMBAD), on reste en DISPLAY.
- Si un candidat GOOD a déjà été trouvé GOOD pour une autre entrée de la table, on reste en DISPLAY, avec le message **Already Connected**.
- Pour permettre un « UPDATE », même si l'objet en entrée contient un identificateur mal écrit:  
*.OK INCORRECT ID*
- Pour permettre un « UPDATE » plusieurs fois sur un même objet:  
*.OK ALREADY CONNECTED*

# 3. Commentaires sur les commandes de mise à jour

- A propos de la référence bibliographique:

La référence bibliographique mise en .B dans le parfile sera ajoutée sur chaque objet mis à jour ou créé. Derrière cette référence sont ajoutées automatiquement trois informations supplémentaires:

- La lettre d, qui indique que l'objet apparaît dans une table
- Le ou les raw-id, c'est-à-dire la liste des identificateurs tels qu'ils sont écrits dans la table. C'est ce qui est donné dans le parfile par %l.R. Si %l.R n'est pas précisé, le programme va prendre le contenu de %l.0 comme raw-id.
- L'occurrence +1

- A propos du type d'objet:

Si le type d'objet principal ne donne pas lieu à la commande 'c o ', il est systématiquement ajouté en type secondaire (commande 'a o ').

- Chaque donnée du parfile donne lieu à une mise à jour éventuelle dans SIMBAD. Dans le cas des UPDATE, le programme va choisir entre garder l'ancienne valeur de SIMBAD ou écraser cette valeur. Dans les deux cas, une ligne contenant la commande de mise à jour est générée.

- Dans le cas où l'ancienne valeur SIMBAD va être écrasée, la ligne sera effective. Une deuxième ligne au-dessous, en commentaire et contenant « **!!Simbad replaced** », rappelle cette ancienne valeur.
- Dans le cas où l'ancienne valeur SIMBAD va être conservée, la ligne est aussi présente, mais en commentaire. Une deuxième ligne au-dessous, en commentaire et contenant « **!!Simbad kept** », rappelle cette ancienne valeur.

Ceci permet d'une part de vérifier rapidement ce que le programme va écraser ou ne pas écraser, d'autre part de récupérer si besoin des commandes de mise à jour qui n'avaient pas été prévues par le programme.

# Présentation du parfile.out : cas d'un DISPLAY

numéro d'ordre du paragraphe (automatique)

rappel des données d'entrée

affichage des données du candidat dans SIMBAD

```

!PAR #1
!### Possible Merge
!=== DISPLAY 12 36 56.3222+62 13 21.805 - MODS deep 2859
!### Identifier not found in the database : MODS deep 2859
!I.0 MODS deep 2859
%J 189.2346758 62.2227235 (Opt) C 2011PASJ...63S.379K
%V z:p 3.24 E 2011PASJ...63S.379K
%M H (AB) 25.3 [0.3] D 2011PASJ...63S.379K
%M K (AB) 24.39 [0.05] C 2011PASJ...63S.379K
%C G
%B 2011PASJ...63S.379K
!
!+++ 1/4: G / G( 1) 0.1"D ( 1.5) ; from coo
! !@ = @483537 ~
!C.0 G
!J 189.2346666667 +62.22269444444 (6) == 12 36 56.32 +62 13 21.7
!J.E (~) [~ ~ ~] D 2003ApJ...596..748T
!M V 30.0 (Vega) [~] D 1999AJ....117..102B
!M I 27.2 (Vega) [~] D 1999AJ....117..102B
!M J 23.0 (Vega) [~] D 1999AJ....117..102B
!V z:~ 4.00 (~) D [~] 2003ApJ...596..748T
!I.0 Cowie 155
!I [T2003] 1723
!#B 2
!
!+++ 2/4: G / G( 1) 0.7"D ( 1.2) ; from coo
! !@ = @481596 ~
!C.0 G
!J 189.23507916667 +62.22273305556 (8) == 12 36 56.4190 +62 13 21.839
!J.E (~) [3 3 90] D ~
!I.0 NAME HDFN-JD1
!I [LYF98] A
!#B 9
!
!--- 3/4: G / G( 1) 2.4"D ( 0.6) ; from coo
! !@ = @478235 ~
!C.0 G
!J 189.23437500000 +62.22338888889 (6) == 12 36 56.25 +62 13 24.2
!J.E (~) [1000 1000 90] D 1996AJ....112.1335W
!I.0 [wBD96] 3-60.0
!#B 1
!
!--- 4/4: G / G( 1) 2.5"D ( 0.6) ; from coo
! !@ = @481073 ~
!C.0 G
!J 189.23483333333 +62.22202777778 (6) == 12 36 56.36 +62 13 19.3
!J.E (~) [1000 1000 90] D 1996AJ....112.1335W
!I.0 [wBD96] 3-111.0
!#B 1

```

décision finale

ligne de résumé des scores pour le candidat 1/4

candidats listés du plus proche au plus éloigné

!+++ candidat GOOD

!... candidat UNDEFINED

!--- candidat BAD

# Présentation du parfile.out : cas d'un UPDATE

numéro d'ordre du  
paragraphe (automatique)

```
!PAR #6
!=== UPDATE from coo : 12 37 07.2645+62 12 13.987
!### Identifrier not found in the database : MODS deep 2864
!+++ 1/1: G / G ( 1) 0.3"C ( 1.4) V( 1.0) M( 0.9); from coo
$o [CBH2004] 189.280273+62.20398
$c c 189.2802689 62.2038853 (Opt) C 2011PASJ...63S.379K
$!!c 12 37 07.2661+62 12 14.310 (Opt) [~ ~ ~] C 2004ApJ...600L..93G !!Simbad kept
$c V z:s 0.6551 D 2011PASJ...63S.379K
$!!V z:spectroscopic 0.655 (~) D [-] 2000ApJ...538...29C !!Simbad kept
$c m U (AB) 26.8 [0.3] D 2011PASJ...63S.379K
$!!m U (AB) 26.61 D 2008ApJ...689..687B !!Simbad kept
$c m B (AB) 26.1 [0.1] D 2011PASJ...63S.379K
$!!m B (AB) 25.61 [0.09] C 2004ApJ...600L..93G !!Simbad kept
$c m V (AB) 23.83 [0.01] C 2011PASJ...63S.379K
$!!m V (AB) 23.33 [0.01] C 2004ApJ...600L..93G !!Simbad kept
$a m i (AB) 22.466 [0.006] B 2011PASJ...63S.379K
$c m z (AB) 22.032 [0.005] B 2011PASJ...63S.379K
$!!m z (AB) 21.547 [0.005] B 2004ApJ...600L..93G !!Simbad kept
$a m J (AB) 21.419 [0.004] B 2011PASJ...63S.379K
$a m H (AB) 21.105 [0.006] B 2011PASJ...63S.379K
$c m K (AB) 20.850 [0.003] B 2011PASJ...63S.379K
$!!m K (AB) 20.38 D 2008ApJ...689..687B !!Simbad replaced
$a o G!
$!!o Galaxy ~ !!Simbad kept
$a id MODS deep 2864
$a r 2011PASJ...63S.379K, d, MODS deep 2864, +1
$bye!
```

décision finale

ligne de résumé des scores  
pour le candidat 1/1

commandes de mise à  
jour pour SIMBAD +  
lignes d'information  
(Simbad kept ou  
Simbad replaced)

# Présentation de parfile.stat

problèmes d'écriture  
(la liste peut être plus longue le cas échéant)

```
#####  
* Identifier incorrect      .  
* NAME                      .  
#####
```

problèmes de cross-identification

```
Conflicting                .  
Possible Merge             47  
Already Connected          2  
#####
```

occurrences pour chaque type de décision

```
* id non presents dans Simbad 154  
#####  
* UPDATE from id :          7  
* UPDATE from coo :         40  
NEW :                        90  
DISPLAY                       64  
#####
```

```
total : Entree : 201 / Sortie : 201  
#####  
* see REJECTED
```

les astérisques \* indiquent les contenus détaillés dessous (pour les trouver, chercher la chaîne de caractères écrite)

les REJECTED sont tous les premiers candidats rejetés de la liste dans les cas de UPDATE ou NEW

## Les fichiers

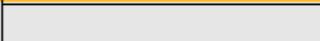
- Le programme a besoin d'un fichier d'entrée, le ***parfile***. Ce fichier contient les données extraites de la table.
- La ligne de commande à écrire:  
***raccord parfile***  
(les options ne s'écrivent pas sur la ligne de commande; elles sont écrites dans le parfile)
- Deux fichiers sont alors créés automatiquement : ***parfile.out***, qui contient entre autres les commandes de mise à jour et ***parfile.stat***, un fichier de statistiques sur les résultats du programme.

# Ecriture du parfile

- Le fichier parfile contient :
  - Une en-tête, constituée de :
    - \_ Une ligne obligatoire avec le bibcode de la table traitée :  
.B 2012A&A...544A..81H
    - \_ Des lignes pour les options (une ligne par option), commençant toujours par un point.
  - Un paragraphe par objet, avec une ligne pour chaque champ, commençant toujours par un % :
    - %l.0 pour l'id principal
    - %l pour un autre id
    - %J pour les coordonnées
    - %V v pour une vitesse
    - %V z:s pour un redshift spectroscopique
    - %M B pour la magnitude B...

# Ecriture des abrégés de champs

		parfiles (derrière % ou !)	scores de Raccord	commandes de mise à jour dans SIMBAD
type d'objet	principal	<b>C</b>	<b>OT</b>	<b>c o</b>
	secondaire	<b>CL</b>		<b>a o</b>
identificateur	le premier	<b>I.O</b>		o/a <b>i</b> ou <b>id</b>
	les suivants	<b>I</b>		
	pour le raw-id	<b>I.R</b>		
coordonnées		<b>J</b>	<b>COO</b>	a/c <b>c</b> ou <b>coo</b>
vitesse		<b>V</b>	<b>V</b>	a/c <b>v</b>
magnitude		<b>F</b> ou <b>M</b>	<b>M</b>	a/c <b>m</b>
dimensions		<b>G</b>	<b>GDIM</b>	a/c <b>g</b>
mouvement propre		<b>P</b>	<b>PM</b>	a/c <b>p</b>
parallaxe		<b>X</b>	<b>X</b>	a/c <b>x</b>
type spectral		<b>S</b>		a/c <b>s</b>
type morphologique		<b>T</b>		a/c <b>t</b>
référence bibliographique		<b>B</b>	<b>B</b>	<b>a r</b>
hiérarchie	parent	<b>H.p</b>		a <b>parent</b>
	enfant	<b>H.c</b>		a <b>child</b>
note	courte	<b>NC</b>		a/c <b>nc</b>
	longue	<b>NL</b>		a/c <b>nl</b>
	de service	<b>NS</b>		a/c <b>ns</b>

 attention! différences  
 sans objet

# Résumé des options (1/2)

- **Pour modifier le rayon de recherche:**

*.search 5"*

*.search 300mas*

*.search 2'*

- **Pour forcer la recherche par coordonnées dans tous les cas:**

*.search coo*

- **Pour interdire la recherche par coordonnées :**

*.search nocoo*

- **Pour borner le sigma, dans la formule du calcul du score sur les coordonnées** (agira à la fois pour l'objet en entrée que pour celui de SIMBAD, toutes lettres confondues, même sur les erreurs entre crochets):

*.sigma COO 300,1000*

- **Pour modifier les valeurs min et max de chaque score:**

*.COO 0.5,0.9*

*.V 0.2,0.2*

*.M -0.1,0.1*

- **Pour favoriser/défavoriser certains types d'objets par rapport à la grille de comparaison par défaut :**

*.OT+2 LM\*,\**

*.OT-1 CIG,GrG*

- **Pour favoriser/défavoriser certains acronymes :**

*.ACRO+1 2MASS*

(ajoute 1 au score ACRO si un id 2MASS est dans l'objet SIMBAD, sauf si un autre est donné en entrée)

*.ACRO-1 IRAS*

(retire 1 au score ACRO si un id IRAS est dans l'objet SIMBAD, sauf s'il est donné en entrée)

*.ACRO++1 PMN*

(ajoute 1 au score ACRO dès qu'un id PMN est dans l'objet SIMBAD)

- **Pour favoriser/défavoriser certaines références :**

*.B+1 2012A&A...537A.147H*

(ajoute 1 au score B si ce bibcode est dans la bibliographie de l'objet SIMBAD)

*.B-1 2013PASJ...65...65T*

(retire 1 au score B si ce bibcode est dans la bibliographie de l'objet SIMBAD)

## Résumé des options (2/2)

- **Pour ne pas tenir compte d'un score** (par ex. j'accepte de cross-identifier deux galaxies même si leur redshift est significativement différent):

*.ignore V*

*.ignore M*

- **Pour donner plus d'importance à un score** = je refuse la cross-id dès que ce score est mauvais (par ex. si j'ai deux galaxies à la même position : je veux qu'elles soient considérées comme différentes si elles n'ont pas le même redshift) (N.B.: c'est le comportement par défaut pour OT et COO):

*.trust V*

*.trust M*

- **Pour donner encore plus d'importance à un score** = j'accepte la cross-id seulement si j'ai une valeur à comparer et que le score est bon (par ex. si j'ai deux galaxies à la même position : non seulement je veux qu'elles soient différentes si elles n'ont pas le même redshift, mais aussi si je n'ai pas de redshift à comparer):

*.expect V*

*.expect M*

- **Pour permettre l'édition de commandes de mise à jour même si un identificateur est mal écrit:**

*.ok incorrect id*

- **Pour passer outre la sécurité « already connected »** (=pour mettre à jour une deuxième fois sur le même objet):

*.ok already connected*

- **Pour forcer le passage de tous les objets trouvés par identificateur:**

*.force from id*