

Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

Cet article présente deux exemples d'utilisation pratique des objets de hachage. Le premier illustre l'utilisation des objets de hachage pour la consultation de tables et compare la performance d'un objet de hachage avec celle de code comparable fondé sur des tableaux et formats. Dans le deuxième exemple, une table de hachage est utilisée pour extraire les 80 principaux clients d'une table volumineuse sans tri des données.

Introduction aux objets composants de l'étape DATA :

SAS propose désormais deux objets composants prédéfinis destinés à l'étape DATA : l'objet de hachage et l'objet de hachage itérateur. Ces objets vous permettent de stocker, de rechercher et de récupérer efficacement des données à l'aide de clés de consultation.

L'interface des objets composants de l'étape DATA vous permet de créer et de manipuler ces objets composants à l'aide d'instructions, d'attributs et de méthodes. Vous pouvez utiliser la marque du point pour accéder aux attributs et méthodes de l'objet composant afin :

- D'assurer le stockage en mémoire et la récupération des données à l'aide de l'objet de hachage.
- De définir un composant données et un composant clé (table et index).
- De charger des données dans l'objet de hachage à partir d'une table SAS (les données d'entrée n'ont pas besoin d'être triées).
- De consulter une ligne de données en fonction des valeurs de clé.
- D'extraire des données en ordre de tri à l'aide de l'objet de hachage itérateur.
- D'ajouter ou de supprimer des lignes de données de façon dynamique.

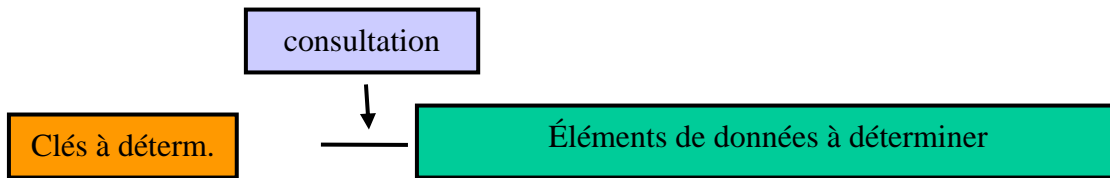
Un attribut, 14 méthodes et 2 instructions sont associés aux objets de hachage et de hachage itérateur. Toute la documentation connexe se trouve dans la documentation en ligne de SAS. Cet article présente des exemples pratiques d'utilisation de ces méthodes et instructions.

Consultation d'une table à l'aide de l'objet de hachage

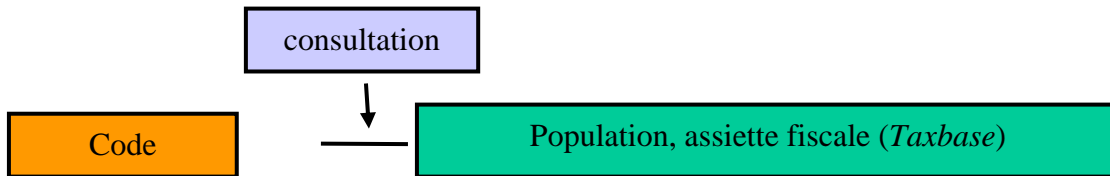
Dans cet exemple, on utilise un jeu de données SAS existant, *work.areadata*, qui comporte des données démographiques associées à des indicatifs régionaux. La clé est constituée de l'indicatif régional stocké dans la variable *area* (région), tandis les données démographiques à extraire sont la taille de la population et l'assiette fiscale (*tax base*).

Pour consulter une table à l'aide d'un objet de hachage :

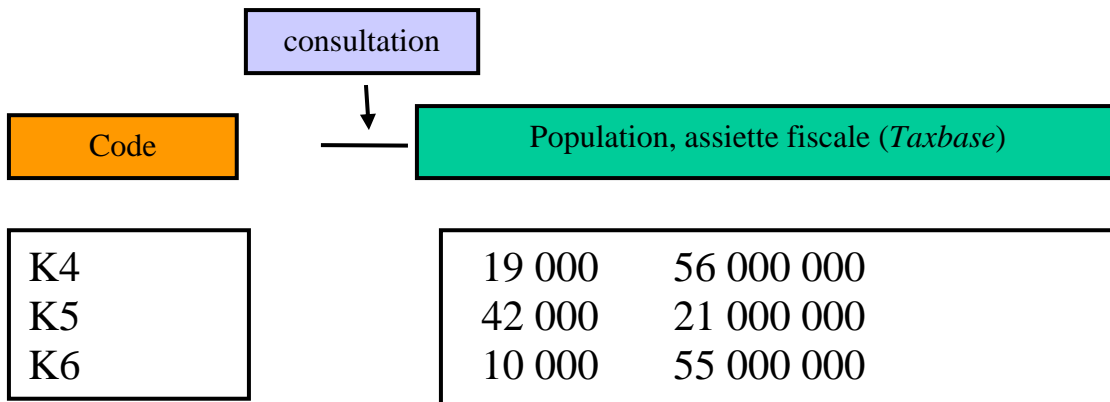
1. Créez un objet que vous nommerez « consultation ».



2. Définissez les clés et les éléments de données.



3. Chargez les données à partir du jeu de données SAS existant.



Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

Le code SAS suivant permettra de mettre en œuvre les étapes 1 à 3 :

```
data combinedA ;  
  if _n_ = 1 then do;  
    if 0 then set work.areadata(keep=area population taxbase);  
    declare hash lookup(dataset: 'work.areadata');  
    lookup.definekey('area');  
    lookup.definedata('population', 'taxbase');  
    lookup.definedone();  
  end;
```

Voici le reste de l'étape DATA qui permet d'effectuer la consultation :

```
  set work.basetable;  
  lookup.find();  
run;
```

où le jeu de données SAS *work.basetable* comporte également la variable *area* (région) qui fournit la valeur de la clé en vue de la consultation de la table.

Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

Comparaison entre les objets de hachage et les tableaux :

- Les objets de hachage peuvent utiliser une variable de caractère, une variable numérique ou une combinaison de ces variables en tant que clé.
- La taille de l'objet de hachage n'est pas spécifiée au moment de sa création.
- Les objets de hachage peuvent stocker de multiples éléments de données dans chaque clé.
- Un même objet de hachage peut stocker des éléments de données de type caractère et numérique.

Voici le code SAS nécessaire pour charger un tableau et l'utiliser pour consulter une table :

```
data makeformat;
  retain fmtname 'areaid' type 'IN';
  set work.areadata(keep=area rename=(area=start));
  label = put(_n_,8.);
run;

proc format cntlin=makeformat fmtlib;
run;

data combinedC;
  array values (27,2) _temporary_;
  if _n_ = 1 then do loop = 1 to totobs;
    set work.areadata nobs=totobs;
    areaid = input(area,areaid.);
    values(areaid,1) = population;
    values(areaid,2) = taxbase;
  end;
  set work.basetable;
  areaid = input(area,areaid.);
  population = values(areaid,1);
  taxbase = values(areaid,2);
run;
```

L'informat qui en résulte établit la correspondance entre un code de caractères désignant la région et un numéro séquentiel, ce qui permet de contourner les limites des tableaux à l'égard des valeurs entières d'index.

Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

Comparaison entre les objets de hachage et les formats :

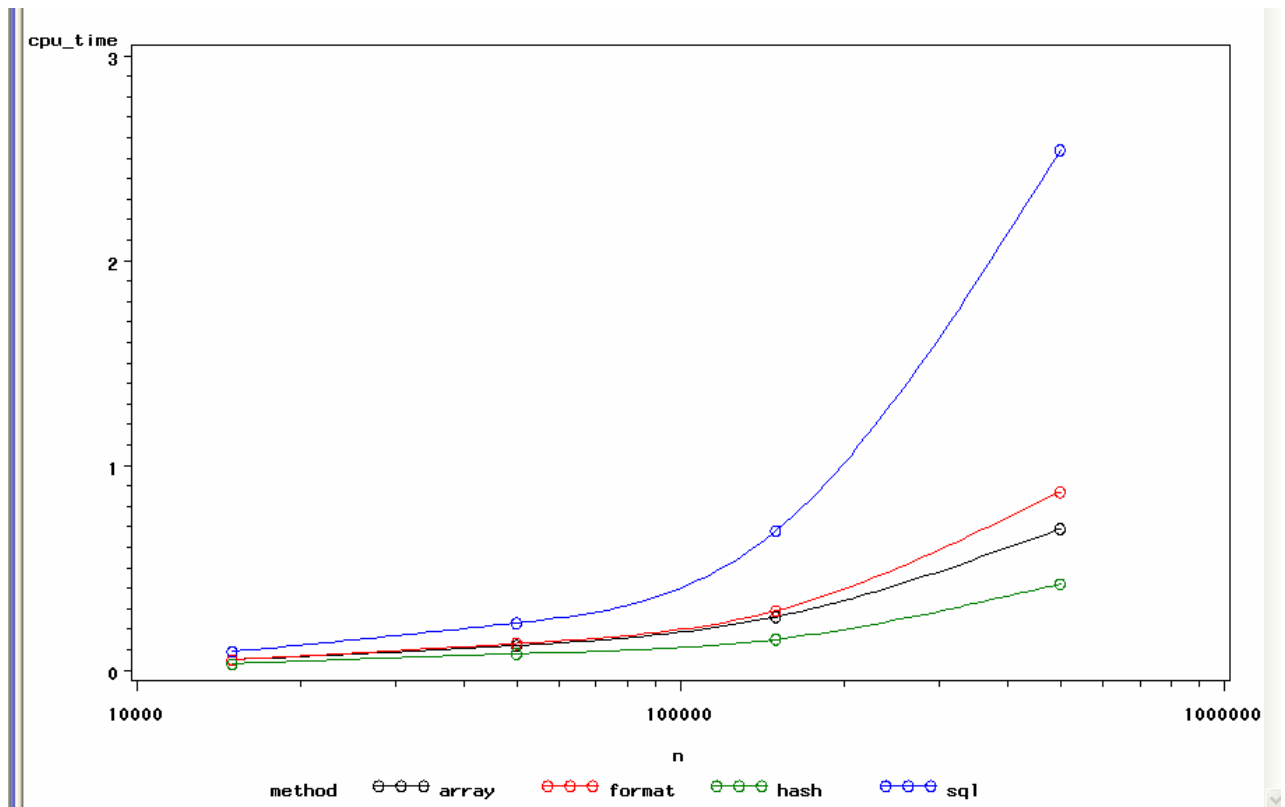
- Un objet de hachage est plus rapide qu'un format.
- Un objet de hachage utilise moins de mémoire qu'un format.
- Un objet de hachage peut stocker de multiples éléments de données par clé.
- Lorsque les données sont volumineuses, les formats occupent beaucoup d'espace disque, ce qui n'est pas le cas des objets de hachage.

Voici le code SAS qui permet d'utiliser directement les formats SAS :

```
data combinedB;  
  set work.basetable;  
  population = input(area,pop.);  
  taxbase = input(area,tax.);  
run;
```

Dans ce cas, vous devez créer deux informats, car un seul informat ne retourne qu'une valeur à la fois.

Notez que le code associé à l'objet de hachage est plus bref et moins complexe que dans le cas du format. Il est également plus efficace. Le banc d'essai ci-dessous compare le coût d'utilisation des objets de hachage avec celui des tableaux, des formats et des liaisons internes (*inner join*) SQL standard.



Étude de cas : Recherche des 80 principaux clients

Une entreprise possède une table historique comportant de l'information sur environ 20 millions de clients. Celle-ci comprend le montant des achats effectués chaque mois pendant 48 mois, de même que des données démographiques. Les travailleurs du savoir doivent extraire les 80 principaux clients répondant à un profil particulier en fonction des données démographiques. Les clients sont classés selon la valeur de leurs achats antérieurs.

Voici les premières lignes de données historiques disponibles :

customerID	phone	address	purchase1
1000001	513 9201233	17639 street name, city name, ON, L5P 1M7	120
1000002	(905)6061405	42639 street name, city name, ON, M2P 1M7	170
1000003	(905)6988306	43660 street name, city name, ON, K1P 1M7	231
1000004	905-463-4117	13910 street name, city name, ON, M6P 1M7	294
1000005	513 7864257	26883 street name, city name, ON L8P 1M7 Canada	336
1000006	(416) 973 7142	27844 street name, city name, ON, K8P1M7	373
1000007	513 404 5075	32202 street name, city name, ON, L4P1M7	421

- L'ID du client (*CustomerID*) constitue la clé primaire.
- Achat1 (*Purchase1*) est l'une des 48 colonnes indiquant le total des achats effectués au cours de chacun des 48 derniers mois.
- Numéro de téléphone avec indicatif régional sous différentes formes.
- Adresse, y compris le code postal sous différentes formes.

Un certain nombre de règles d'affaires régissent l'affectation d'une valeur totale à chaque client.

- La valeur totale d'un client correspond à la moyenne pondérée des achats effectués au cours des 48 mois, les achats des 12 derniers mois faisant l'objet d'une pondération double.
- Seuls les clients se trouvant dans la région téléphonique 905 sont inclus.
- Seuls les clients dont le code postal débute par « L » sont inclus.

L'extraction finale contiendra les 80 principaux clients en ordre décroissant selon la valeur totale :

	totalvalue	customerID	phone	address
1	5413	1001299	(905)4058224	27627 street name, city name, ON, K7P1M7
2	5411	1097199	(905) 767 9094	52279 street name, city name, ON K3P1M7
3	5405	1093399	905 2106868	17896 street name, city name, ON, K9P 1M7
4	5403	1078799	905-523-1200	5491 street name, city name, ON, L9P1M7
5	5403	1026999	513 6651082	1681 street name, city name, ON K2P1M7

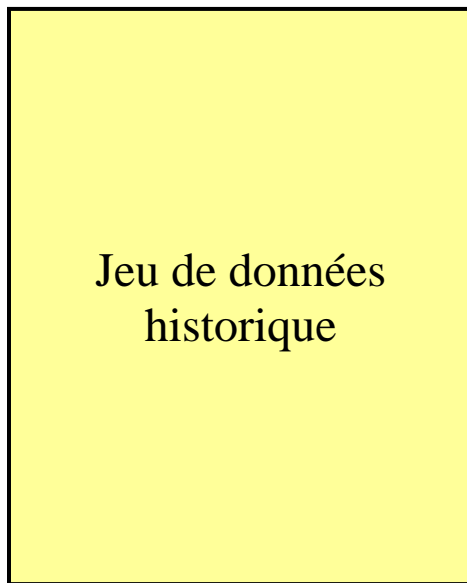
Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

75	5388	1101199	(513) 905 8283	48860 street name, city name, ON, L8P 1M7
76	5388	1132299	513-521-7283	28575 street name, city name, ON K6P 1M7 Canada
77	5388	1025899	416-424-8155	54921 street name, city name, ON L8P 1M7 Canada
78	5388	1129999	(905)9606406	20864 street name, city name, ON M6P1M7
79	5388	1147299	(905) 797 5730	57641 street name, city name, ON, M1P1M7
80	5388	1009999	(513)8246037	40992 street name, city name, ON, L7P 1M7

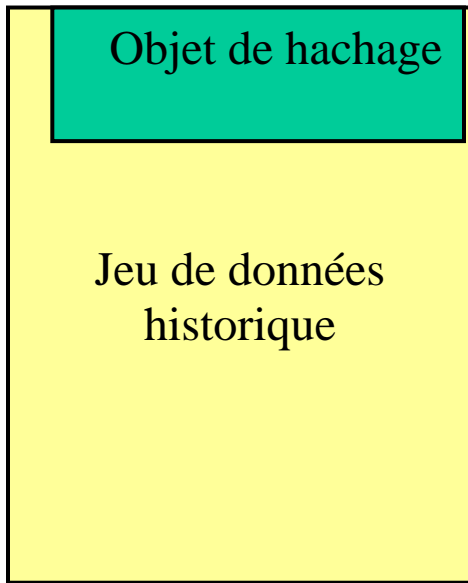
Au moins deux stratégies permettent de rechercher les 80 principaux clients :

- SAS^{MD} 9 : En utilisant une seule étape DATA et une lecture séquentielle des données.
- L'autre possibilité : En calculant la valeur dans une étape DATA puis en triant la table résultante.

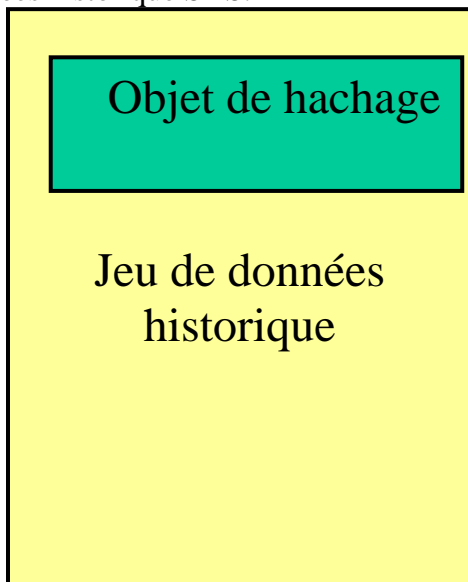
Le processus qui permet d'utiliser un objet de hachage en tant que fenêtre mobile est le suivant :



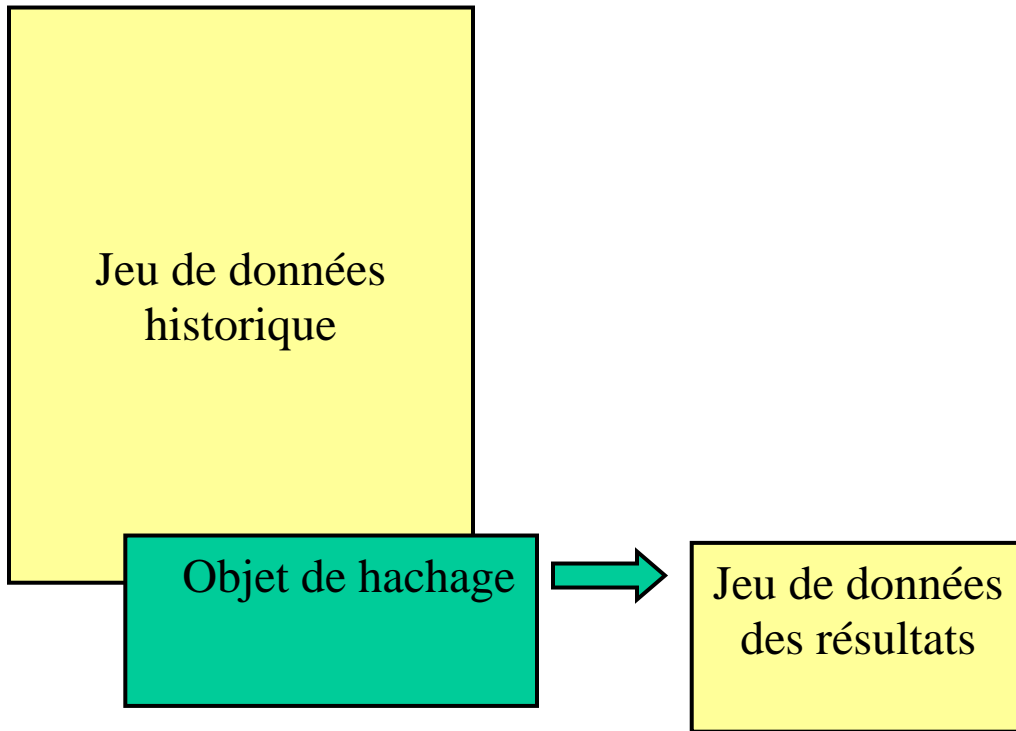
Étape 1 : Utilisez le jeu de données historique SAS comme point de départ.



Étape 2 : Définissez un objet de hachage en y incluant les 80 premières lignes du jeu de données historique SAS.

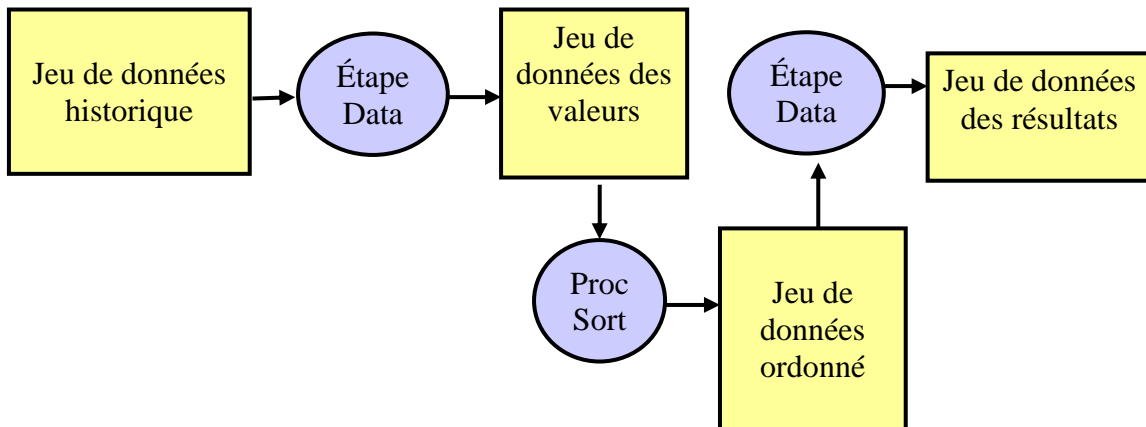


Étape 3 : Déplacez l'objet de hachage vers le bas, une ligne à la fois. Si la nouvelle ligne est plus longue que le plus petit élément se trouvant actuellement dans l'objet de hachage, remplacez ce plus petit élément par la nouvelle ligne.



Étape 4 : Lorsque l'objet de hachage atteint le bas, il contient les 80 principaux éléments du jeu de données historique SAS. Transférez le contenu de l'objet de hachage vers le jeu de données SAS des résultats.

L'autre possibilité consiste à utiliser la procédure SORT (*Proc Sort*), selon le flux de données suivant :



La technique faisant appel à PROC SORT nécessite plus de mémoire, plus d'espace sur le disque de travail et plus de cycles d'UC que la technique utilisant un objet de hachage.

Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

Le banc d'essai ci-dessous compare le coût de la technique utilisant un objet de hachage avec celui de la technique faisant appel à PROC SORT.

Nombre total de lignes	Taille du groupe	UC hachage	UC SORT	Ratio
10 001	80	0,05	0,09	1,80
50 001	80	0,12	0,24	2,00
250 001	80	0,50	1,12	2,24
1 000 001	80	1,88	5,03	2,67
4 000 001	80	6,95	19,56	2,81

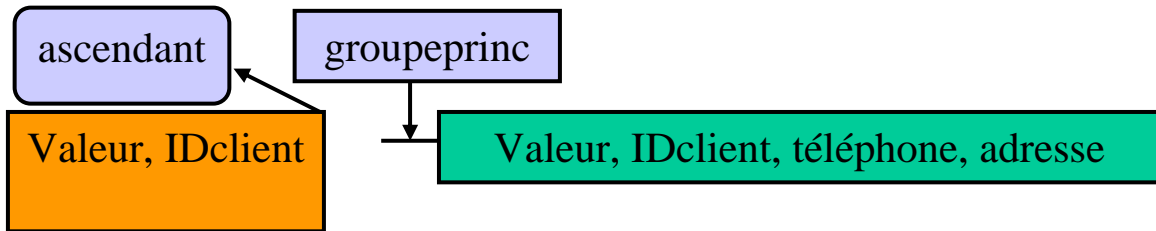
Chaque valeur d'UC constitue une moyenne de trois exécutions.

Notons un autre avantage important conféré par l'objet de hachage dans ce contexte : contrairement à l'utilisation de la procédure SORT, la quantité de mémoire et d'espace sur le disque de travail requis par la technique utilisant l'objet de hachage n'augmente pas à mesure que la taille du jeu de données historique SAS augmente.

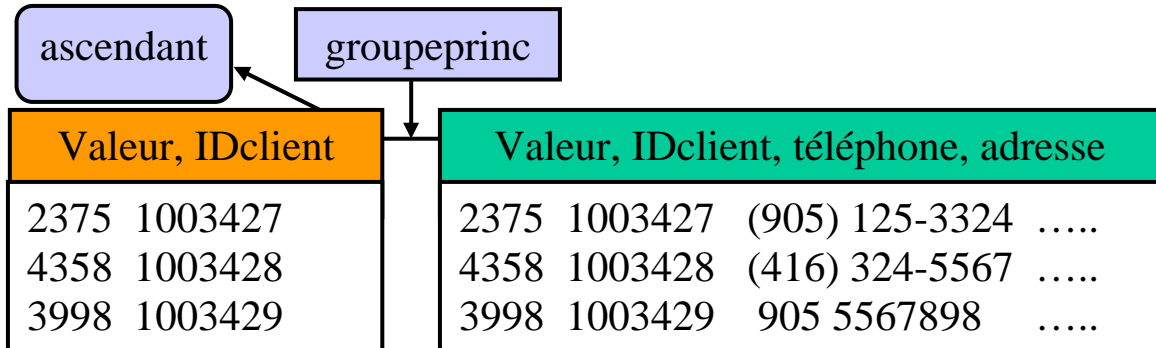
Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

Les premières étapes d'une recherche des 80 principaux clients à l'aide d'un objet de hachage s'apparentent à l'utilisation d'un objet de hachage comme table de consultation :

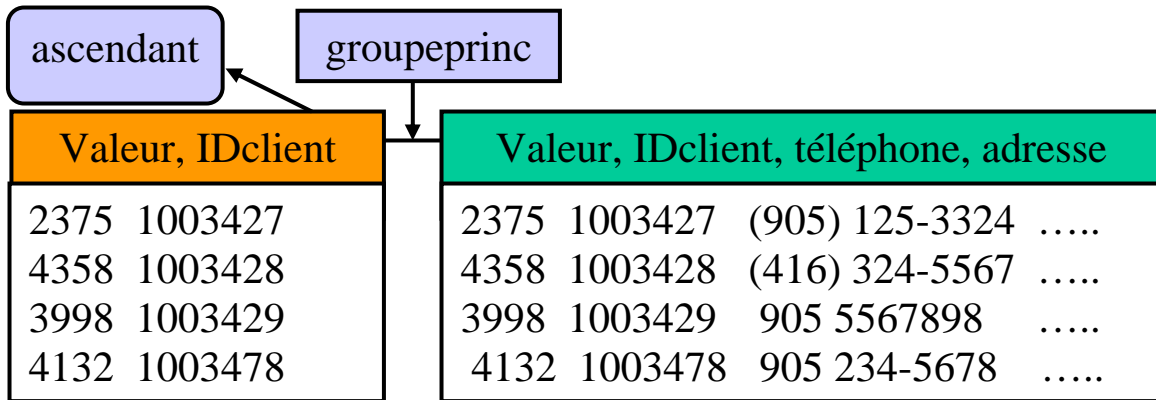
1. Créez un objet et nommez-le « groupeprinc ».
2. Définissez les clés et les éléments de données.
3. Créez un objet de hachage itérateur pour permettre l'accès aux données dans l'objet de hachage selon l'ordre des clés.



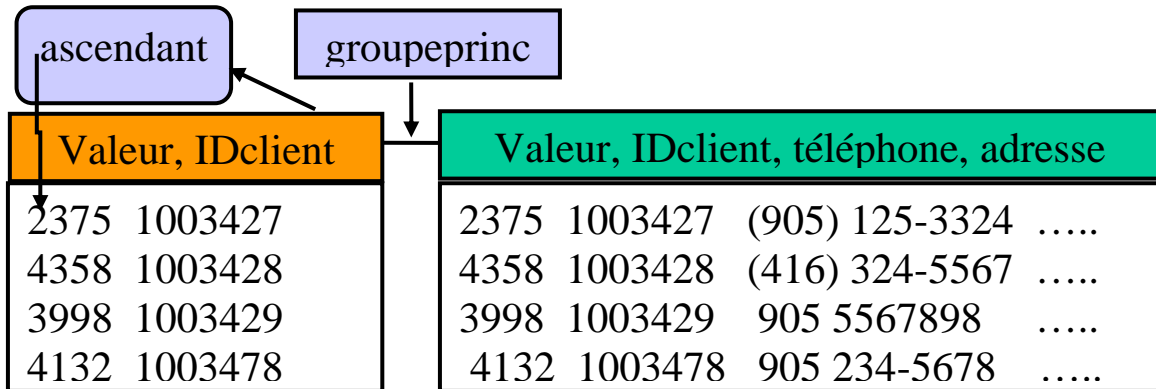
4. Chargez les 80 premières lignes de données provenant du jeu de données historique SAS.



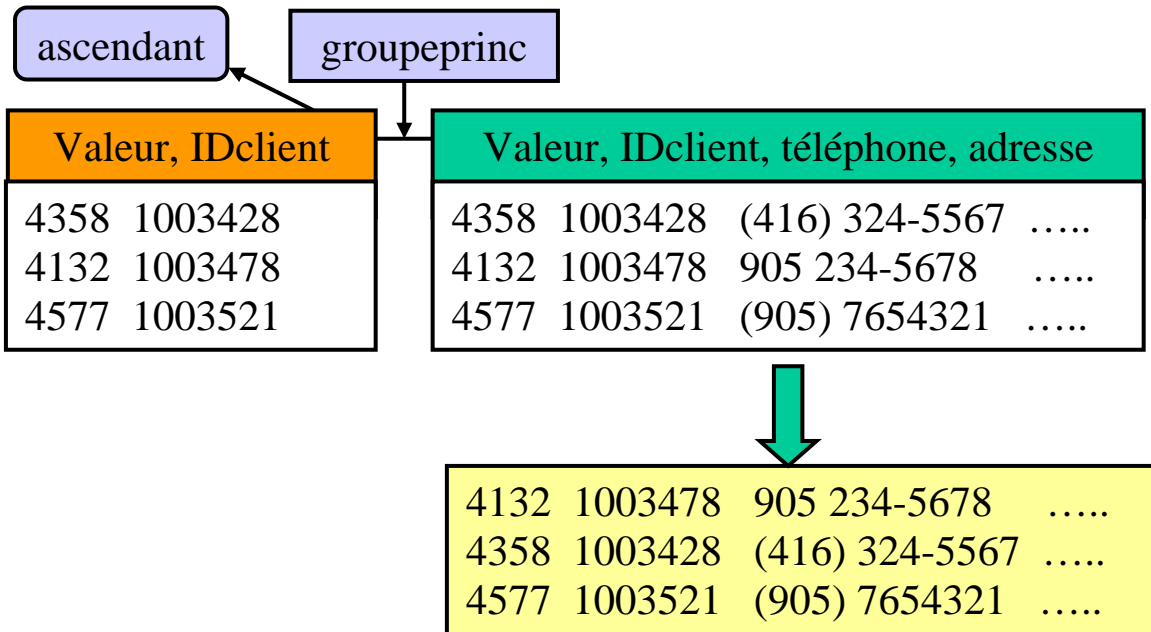
5. Ajoutez les données du vecteur de données du programme lorsque la valeur calculée est supérieure à la plus petite valeur actuellement stockée.



- Repérez la plus petite valeur actuelle.
- Supprimez l'élément correspondant à cette valeur.



- Continuez d'ajouter et de supprimer des éléments jusqu'à ce que toutes les données historiques aient été traitées.
- Transférez les résultats dans un jeu de données SAS en ordre ascendant.



Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

Voici le code qui permet de mettre en œuvre le processus décrit ci-dessus. Les commentaires indiquent les segments du code qui correspondent aux étapes décrites ci-dessus.

```
data _null_ ;
  length rc 8;
  retain minvalue mincustomerid;
  /* declare variables in hash object */
  length totalvalue 8 customerID 8 phone $ 15 address $50;
  format totalvalue 10.;
  /* initialize hash object and hash iterator (steps 1 to 3) */
  if _n_=1 then do;
    if 0 then set work.history(keep=totalvalue customerid phone address);
    declare hash topgroup(ordered='ascending');
    declare hiter ascend('topgroup');
    topgroup.definekey('totalvalue','customerid');
    topgroup.definedata('totalvalue', 'customerID','phone','address');
    topgroup.definedone();
  |
    /* load first block of rows into hash object (step 4)*/
  do loop = 1 to 80;
    link getdatarow;
    link calctotalvalue;
    rc = topgroup.add();
  end;
  rc = ascend.first();
  minvalue = totalvalue;
  mincustomerid=customerid;
  rc = ascend.next(); /* avoid pointing to a row you may remove later*/
end; /* the _n_=1 loop */

  /* process remaining data rows (step 8) */
  link getdatarow;
  link calctotalvalue;

  /* test if value is greater than minimum (step 5, 6, and 7) */
  if totalvalue gt minvalue then do;
    rc = topgroup.add();
    rc = topgroup.remove(KEY:minvalue,KEY:mincustomerid);
    rc = ascend.first();
    minvalue = totalvalue;
    mincustomerid = customerid;
    rc = ascend.next(); /* avoid pointing to a row you will remove later*/
  end;
```

Utilisation d'objets de hachage dans SAS 9
par Bill Fehlner, SAS Institute
Juillet 2005

```
        /* output the results (step 9) */  
    if eof then do;  
        rc = topgroup.output(dataset:"work.results");  
    end;  
    return;  
  
    getdatarow: /* read row from input dataset */  
        set work.history end=eof;  
    return;  
  
    calctotalvalue: /* use simple business rule to calculate total value */  
        totalvalue = (2 * sum(of purchase37-purchase48) + sum(of purchase1-purchase36))/  
            (2 * n(of purchase37-purchase48) + n(of purchase1-purchase36));  
    return;  
  
run;
```

Conclusion :

L'objet de hachage est un mécanisme efficace et pratique pour le stockage et la récupération rapides des données. Le code nécessaire à la mise en œuvre d'un objet de hachage n'est pas plus complexe (et l'est même parfois moins) que pour d'autres stratégies utilisées aux mêmes fins dans SAS 8.