

Table of Contents

- Hash objects: quick introduction
- Examples of applying hash objects:
 - Detection of repeat adverse events
 - Flagging of all records from an interval
 - Mapping of baseline flag
- Advantages and disadvantages of hash objects

Hash Objects: Quick Introduction

Hash object is a dynamic data structure - an “array” that can be accessed by a key and uses methods to navigate it

- Resides in memory
- Used only in data step and exists during its execution
- Filled from a data set or from data generated by current data step
- Can contain numeric and character variables
- Modifies its size dynamically
- Has attributes and methods that are written in dot notation: *object.method()*

Examples of applying hash objects

Detection of Repeat Adverse Events

Instantiation of hash object

```
data repeated(keep= subj visn term repeated);  
if _N_ = 1 then do;  
  declare hash events(dataset:'adv_ev(rename=(term=hash_term  
    visn=hash_visn subj=hash_subj))', ordered: 'a');  
    events.definekey('hash_subj', 'hash_visn');  
    events.definedata(all:"yes");  
    events.definedone();  
  declare hiter pointer('events');  
end;  
  length hash_term $20 hash_visn hash_subj 8.;  
  call missing(hash_term, hash_subj);  
  set adv_ev;
```

Defining of
key and
data items

Hash iterator

Adding of hash
items to PDV

Detection of Repeat Adverse Events

Code for navigation through the hash

```
do i = 1 to _N_-1;  
    rc = pointer.next();  
end;  
do while (rc = 0);  
    if subj = hash_subj  
    and hash_term = term  
    then repeated="Y";  
    rc = pointer.prev();  
end;  
  
run;
```

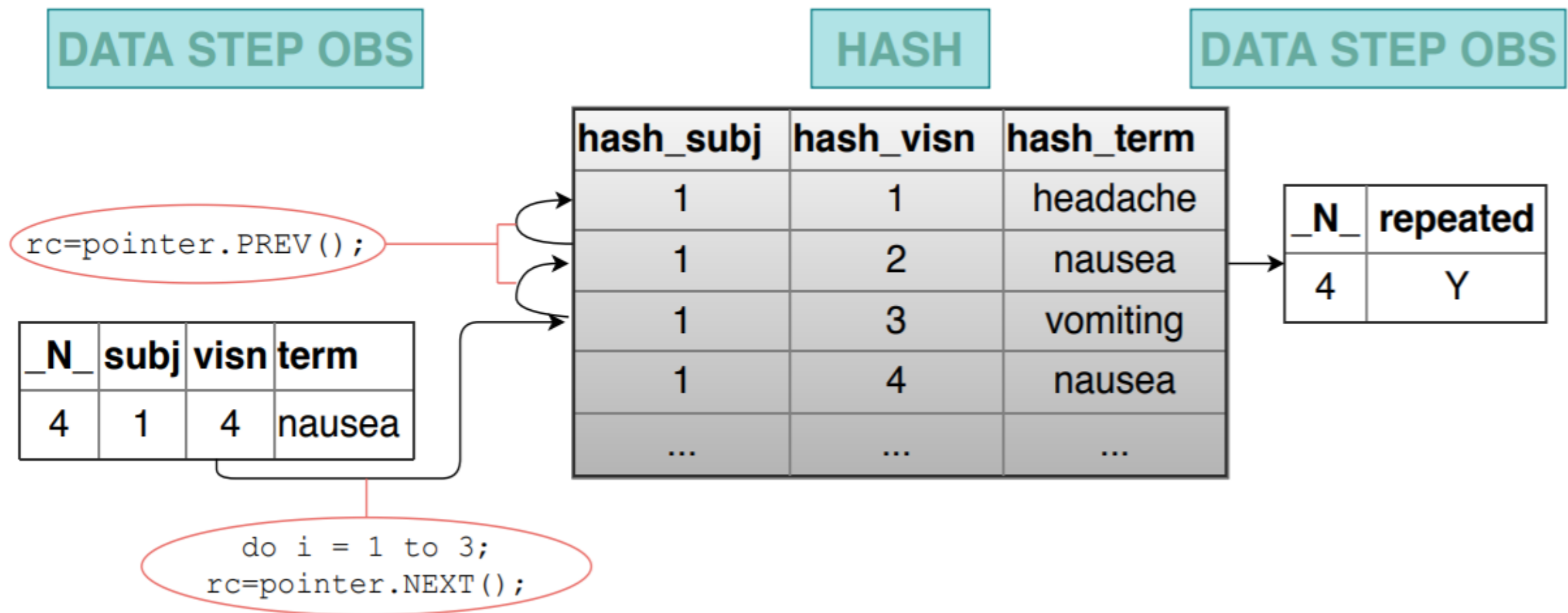
REPEATED ▾

  Filter and Sort  Query Builder  Where | Data ▾ Describe ▾

	 subj	 visn	 term	 repeated
1	1	1	headache	
2	1	2	nausea	
3	1	3	vomiting	
4	1	4	nausea	Y
5	2	1	headache	
6	2	2	rash	
7	2	3	nausea	
8	2	4	dizziness	
9	2	5	rash	Y
10	2	6	rash	Y
11	3	5	fever	
12	3	6	fever	Y
13	3	7	fever	Y

Detection of Repeat Adverse Events

Example of navigation through hash for a particular observation



Flagging of All Records from an Interval

Goal: looking for intervals where FL = "Y" during at least 10 days

```
data intervals(keep=subj fl day stabilization);  
set source;  
length hashfl $1 hashday hashsubj 8. ;  
call missing(hashfl, hashday, hashsubj);  
if _n_=1 then do;  
  declare hash src  
    (dataset: 'source(rename=(fl=hashfl  
      day=hashday subj=hashsubj))',  
    ordered: 'a');  
  src.definekey('hashsubj', 'hashday');  
  src.definedata(all:"yes");  
  src.definestone();  
  declare hiter tracer("src");  
end;
```

INTERVALS ▾

	 Filter and Sort	 Query Builder	 Where	
	 SUBJ	 FL	 DAY	 STABILISATION
1	1	N	2	
2	1	Y	12	
3	1	Y	15	
4	1	N	22	
5	1	Y	24	Y
6	1	Y	27	Y
7	1	Y	35	Y
8	2	Y	2	
9	2	N	15	
10	2	N	20	
11	2	Y	22	Y
12	2	Y	35	Y

Flagging of All Records from an Interval

Navigation through the hash

```
rc1 = tracer.first();  
do i = 1 to _n_; rc1 = tracer.next(); end;  
if fl = "Y" and first.subj=0 then  
    do while (subj = hashsubj and hashfl = "Y" and rc1 = 0);  
        if hashday - day > 10 then do;  
            stabilization = "Y";  
            retain stabilization;  
            leave;  
        end;  
        rc1 = tracer.next();  
    end;  
else call missing(stabilization);  
run;
```

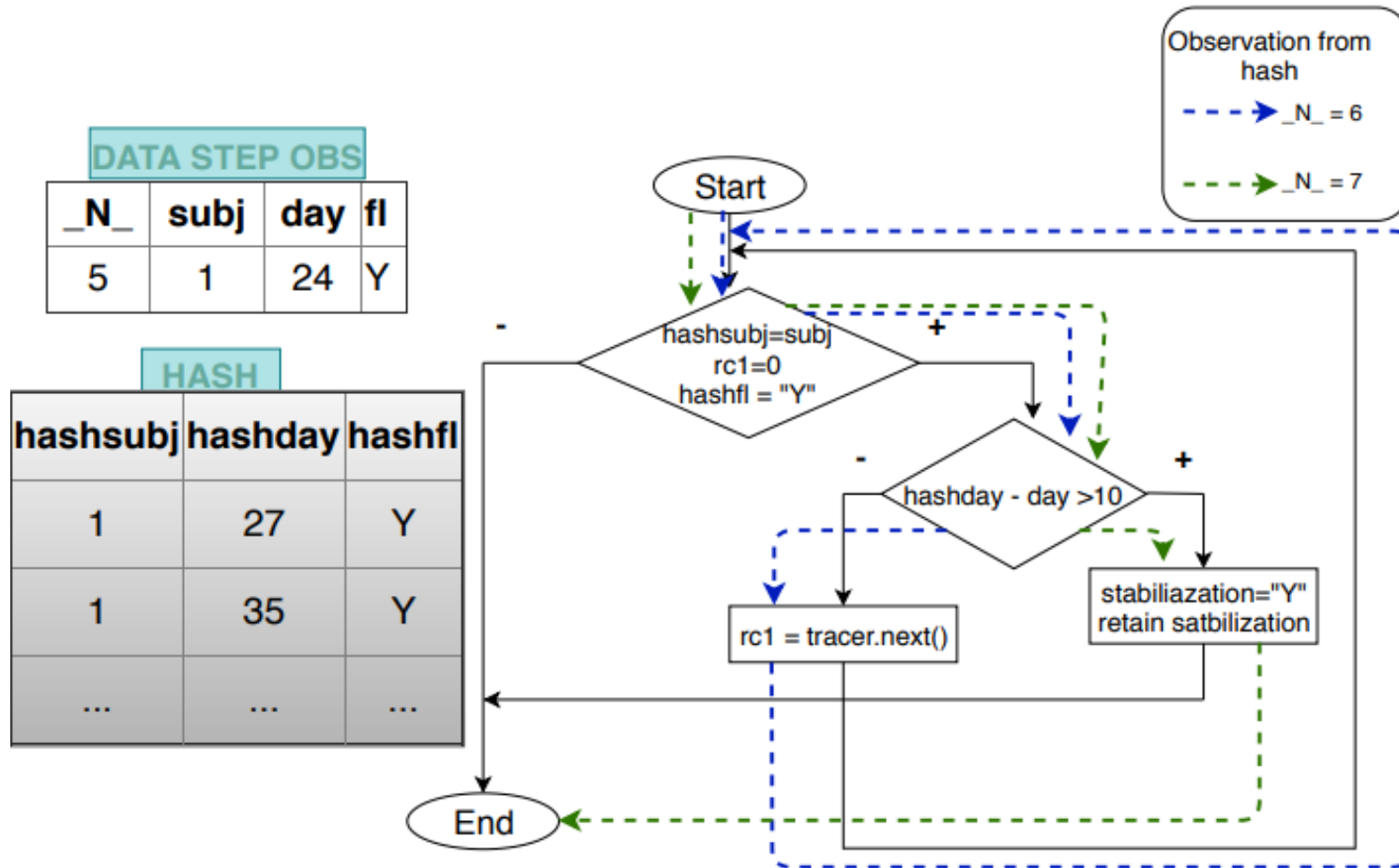
HITER reaches the record in SRC next to the current observation

Navigate through hash records within current subject till flag = "N" and checks if difference in days >= 10

Break of the interval

Flagging of All Records from an Interval

Algorithm for a particular observation



Mapping of Baseline Flag

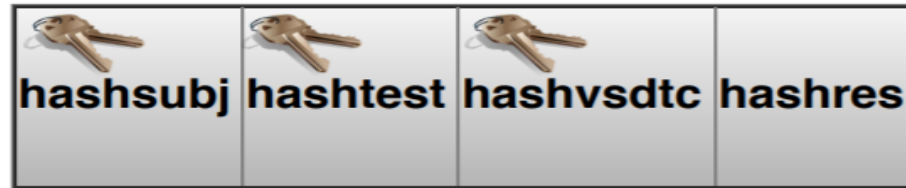
Creation of the needed hash objects

```
data baseln(drop= hash: rc: i);  
    /*Declaring of hash objects STTRT and TESTS*/
```

STTRT



TESTS



```
set vs;  
if sttrt.find() = 0;  
rc2 = tracer.first();  
do i = 1 to _n_;  
    rc2 = tracer.next();  
end;
```

DM merging

TRACER goes to the next
(relative to current current
read from VS) observation

Mapping of Baseline Flag

Finding of baseline using hash

```
if vsdte < rfxstdte and not missing(res)
then do;
    pot_bl = "Y";
    do while(rc2 = 0 and subj = hashsubj and test = hashtest);
        if hashvsdte<rfxstdte and nmiss(hashres)=0
            then do;
                pot_bl="N";
                leave;
            end;
        rc2 = tracer.next();
    end;
    if pot_bl = "Y" then blfl = "Y;
end;
```

Check if current VS record can potentially be baseline one

Define if there is any potential baseline record after current one

Speed of Execution

N obs in hash	Example 1		Example 2		Example 3	
	hash	no hash	hash	no hash	hash	no hash
1 000	0.01	0.03	0.04	0.08	0.12	0.02
10 000	0.04	0.05	0.06	0.19	0.15	0.05
100 000	0.05	0.27	0.09	0.48	0.22	0.34
1 000 000	0.08	2.38	1.12	4.15	0.25	2.99
5 000 000	1.35	26.08	4.21	44.31	6.31	19.81

Advantages and Disadvantages of Hash Objects



- Performing a table lookup using unsorted datasets
- Low and almost consistent runtime when scaled
- Can return more than one data item
- Items can be a mix of char and num data



- Size too large for available memory
- Unusual approach for SAS users
- Limited analyses features compared to PROC step capabilities

Thank you for your attention!

Any questions?

