create table algorithmisation.person\_tower\_days as select M.caller\_msisdn, M.bts\_nodeb, M.longitude, M.latitude, count(DISTINCT M.call\_date) as tot\_days from (select S.caller\_msisdn, S.call\_date, S.call\_time, T.bts\_nodeb, T.longitude, T.latitude from algorithmisation.subset\_of\_cdr\_november S inner join algorithmisation.antennas T on (S.ms\_location=T.id\_cellule)) M group by M.caller\_msisdn, M.bts\_nodeb, M.latitude, M.longitude;

create table algorithmisation.person\_towers as select caller\_msisdn, count(DISTINCT bts\_nodeb) as total\_towers from algorithmisation.person\_tower\_days group by caller\_msisdn;

create table algorithmisation.person\_tower\_days\_final as select R.\*, rank() over (partition by R.caller\_msisdn order by R.tot\_days desc, rand() desc) as rnk from (select S.caller\_msisdn, S.bts\_nodeb, S.longitude, S.latitude, S.tot\_days, M.total\_towers from algorithmisation.person\_tower\_days S inner join algorithmisation.person\_towers M on (S.caller\_msisdn=M.caller\_msisdn)) R;

create table algo\_july.person\_tower\_days\_final as select R.\*, rank() over (partition by R.caller\_msisdn order by R.tot\_days desc, rand() desc) as rnk from (select S.caller\_msisdn, S.bts\_nodeb, S.longitude, S.latitude, S.tot\_days, M.total\_towers from algo\_july.person\_tower\_days S inner join algo\_july.person\_towers M on (S.caller\_msisdn=M.caller\_msisdn)) R;

create table algorithmisation.subset\_of\_cdr\_larger as select distinct M.caller\_msisdn from (select S.caller\_msisdn, T.region from algorithmisation.subset\_of\_cdr\_july S left join algorithmisation.antennas T on (S.ms\_location=T.id\_cellule) where region=”DK”) M where rand()<=0.1 distribute by rand() sort by rand() limit 100000;

create table algorithmisation.subset\_of\_cdr\_November as select M.\* from subset\_of\_cdr\_july as M right join subset\_of\_cdr\_larger SC on (M . caller\_msisdn= SC. caller\_msisdn) ;

create table algo\_july.subset\_of\_cdr\_July as select M.\* from algo\_july.subset\_of\_cdr\_july as M right join algorithmisation.subset\_of\_cdr\_larger as SC on (M . caller\_msisdn= SC. caller\_msisdn) ;

Algo\_July : 87368 caller\_msisdn, 1185450 entrées;

Algorithmisation: 100000 caller\_msisdn, 2910663 entrées;