Database size :

Collection	#int/numbers	#string	#date	#long string	Avg array length	#array int	#array string	#array date	#array long string	#keys	Doc size in B	#doc	Collection size
Size in B	8 B	80 B	20 B	200 B		8 B	80 B	20 B	200 B	12 B			
Prod{[Cat], Supp}	5	5		1	2		1			13 + 2 =15	980	10^5	0.098 GB
St	3	1								4	152	$2*10^7$	3.04 GB
Wa	2	1								3	132	200	26400 B
OL	4		2	1						7	364	$4*10^9$	1456 GB
Cl	1	5	1							7	512	10^7	5.12 G B
Prod{[Cat], Supp, [**St**]}	5	5		1	2 200	2	1 1			613 + 3 = 616	27392	10^5	2.6392 GB
St{Prod{[Cat],Supp}}	7	6		1	2		1			16 + 3 = 19	1108	$2*10^7$	22.16 GB
OL{Prod{[Cat],Supp}}	8	5	2	2	2		1			19 + 3 =22	1328	$4*10^9$	5312 GB
Prod{[Cat], Supp, [**OL**]}	5	5		1	2 $4*10^4$	3	1	2	1	240013 + 3 = 240016	13440992	10^5	1344.0992 GB

Doc size = #int*size int + #string* size string + #date*size date + #longstring * size longstring + avg arra length * (#int*size int + #string* size string + #date*size date + #longstring * size longstring) + #keys* size key

DB1: Prod{[Cat],Supp}, St, Wa, OL, Cl = size(Prod{[Cat],Supp}) + size(ST) + size(Wa) + size(OL) + size (Cl)

DB2: Prod{[Cat],Supp, [**St**]}, Wa, OL, Cl

DB3: **St**{Prod{[Cat],Supp}}, Wa, OL, Cl

DB4: St, Wa, **OL**{Prod{[Cat],Supp}}, Cl

DB5: Prod{[Cat],Supp, [**OL**]}, St, Wa, Cl

Sharding strategy impact :

Sharding Strategy	# doc / server	# Distinct values / server
St - #IDP	$2*10^7$ / 1000 = $2*10^4$ stocks	10^5 / 1000 = 100 distinct products
St - #IDW	$2*10^7$ / 200 = 10^5 stocks for 200 srv	200/1000 = 1 warehouse per srv for 200 srv
OL - #IDC	$4*10^9$ / 1000 = $4*10^6$ orderlines	10^7 / 1000 = 10^4 distinct clients
OL - #IDP	$4*10^9$ / 1000 = $4*10^6$ orderlines	10^5 / 1000 = 100 distinct products
Prod - #IDP	10^5 / 1000 = 100 products	10^5 / 1000 = 100 distinct products
Prod - #brand	10^5 / 1000 = 100 products	5000/1000 = 5 distinct brands of products