Limites d’une base de données orientée Document (ici, mongoDB) :

* Il n’y a plus de gestion de jointures
* Il se peut qu’il y ait une incohérence des données
* En cas de synchronisation latente, cette situation peut faire apparaître des incohérences poussées
* La charge du contrôle des données est reportée du côté de l’application, ce qui peut s’avérer très pénalisant.
* MongoDB possède un langage d’interrogation qui lui est propre, donc sans standard (pratique mais limité)
* Il n’offre aucun support transactionnel
* MongoDB est aussi utilisé pour stocker toutes les erreurs et transactions lentes, mais le log d’erreur est limité en taille
* La taille des documents ne peut pas dépasser 16 MB
* Il ne peut y avoir que 64 index max par collection

Redis :

Les données sont stockées sous forme <clé, valeur>.

On peut y stocker pluseurs structures, à savoir :

* Les listes de String (LISTs)
* Les collections non ordonnées de Strings (SETs)
* Les Maps de String (Hashs)
* Les index (SortedSet ou ZSET)

Les données stockées peuvent être limitées dans le temps, il est possible que Redis les supprime à partir d’un temps donné

Ensuite, Redis est un cache puissant. Des données peuvent être optimisées et l’on peut donc placer en frontal plusieurs serveurs sur un même redis.

Il est aussi pratique lorsque l’on souhaite gérer une queue asynchrone de message.

Il est possible d’effectuer plusieurs requêtes, comme :

* SET : rajouter une valeur à une clé
* GET : Lire une valeur associée à une clé
* NCR : Créer un compteur
* Fixer une limite de temps
* Enregistrer plusieurs valeurs dans une seule clé