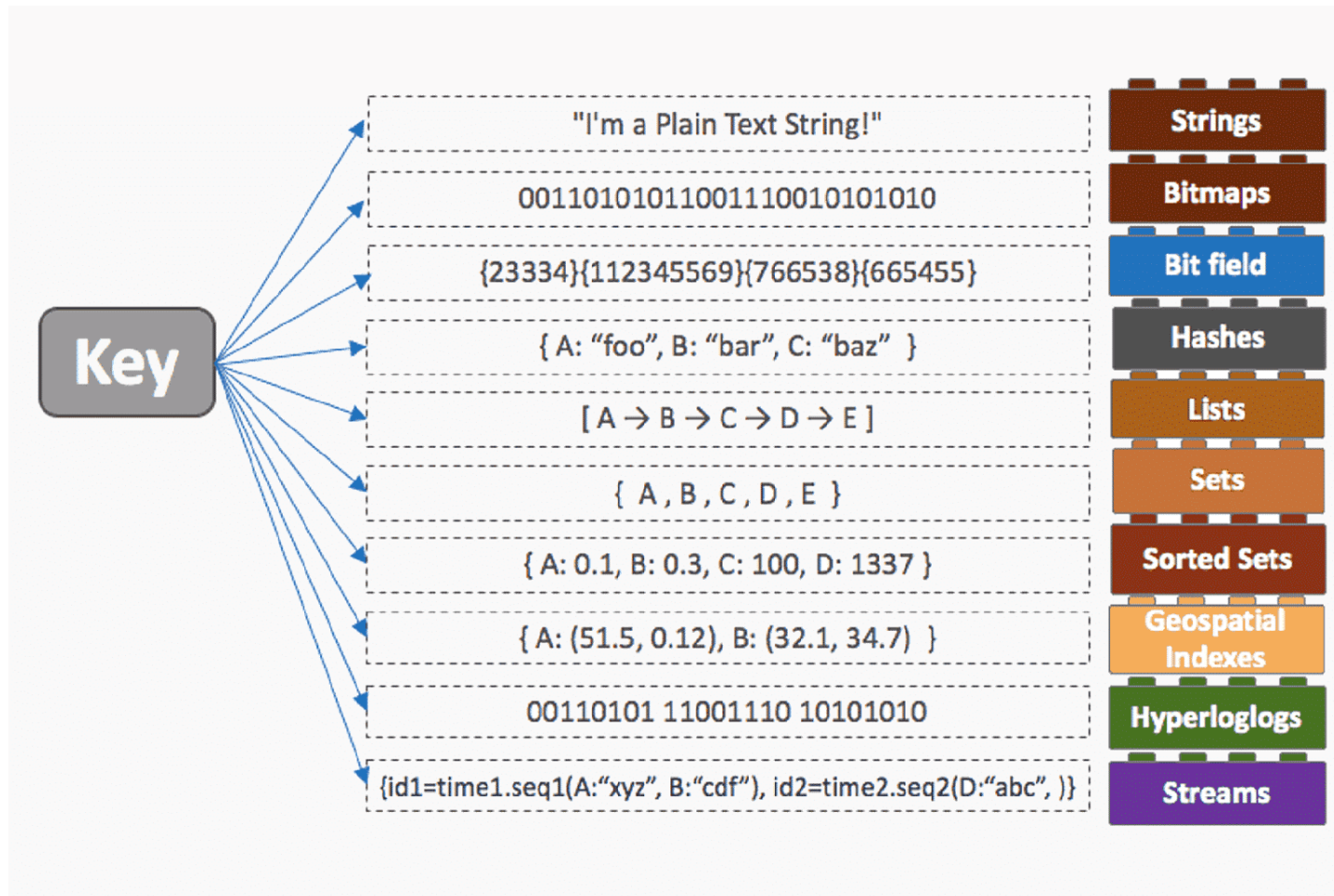


REDIS



redis

Armazenamento de dados em memória de código-aberto usado por milhões de desenvolvedores em banco de dados, cache, streaming e mensageria.



Redis é uma ferramenta de banco de dados em memória (in-memory database) de código aberto, usada principalmente como um armazenamento de dados de estrutura chave-valor (key-value store). Redis é a abreviação de **RE**mote **D**ictionary **S**erver.



Você pode utilizar o Redis como banco de dados, mas ele é um banco de dados volátil estando em memória. O interesse dele não é concorrer com o MySQL, Postgres ou outros relacionais.

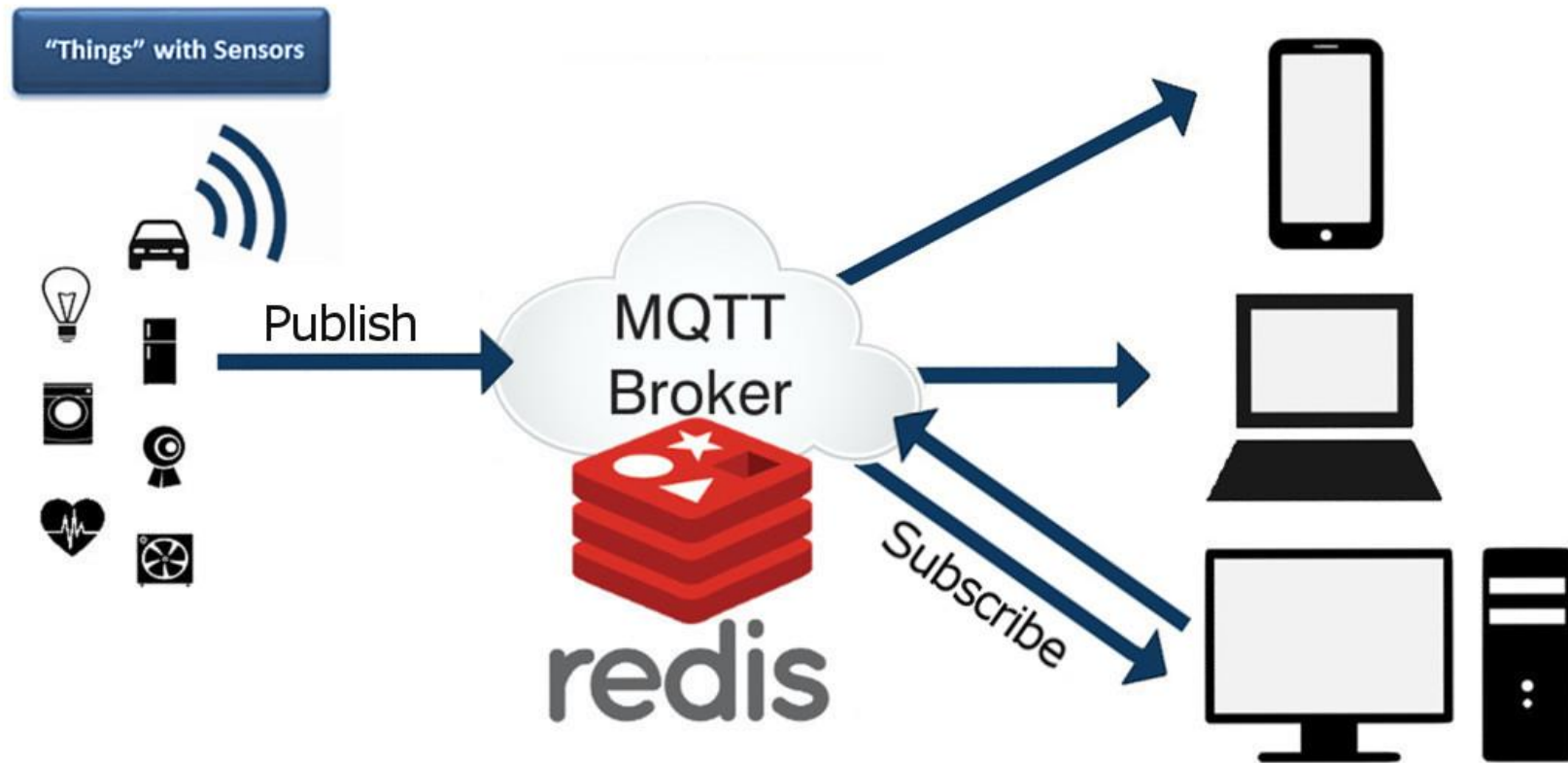


Ele pode funcionar como banco de dados, pode salvar dados, mas sempre em memória.

Ele é muito utilizado como cache sendo essa a principal utilização do Redis. Usado para fazer consultas em memória de algumas das informações recorrentes ou custosas do banco de dados principal.



Naturalmente você vai ter o seu banco e o Redis na frente para guardar parte das informações de acesso mais rápido.



Podemos fazer com que sistemas possam conversar entre si, ou as vezes o mesmo sistema, mas partes diferentes dele.



Redis também pode ser usado para streaming, para fluxo contínuo de dados, também para mensageria.



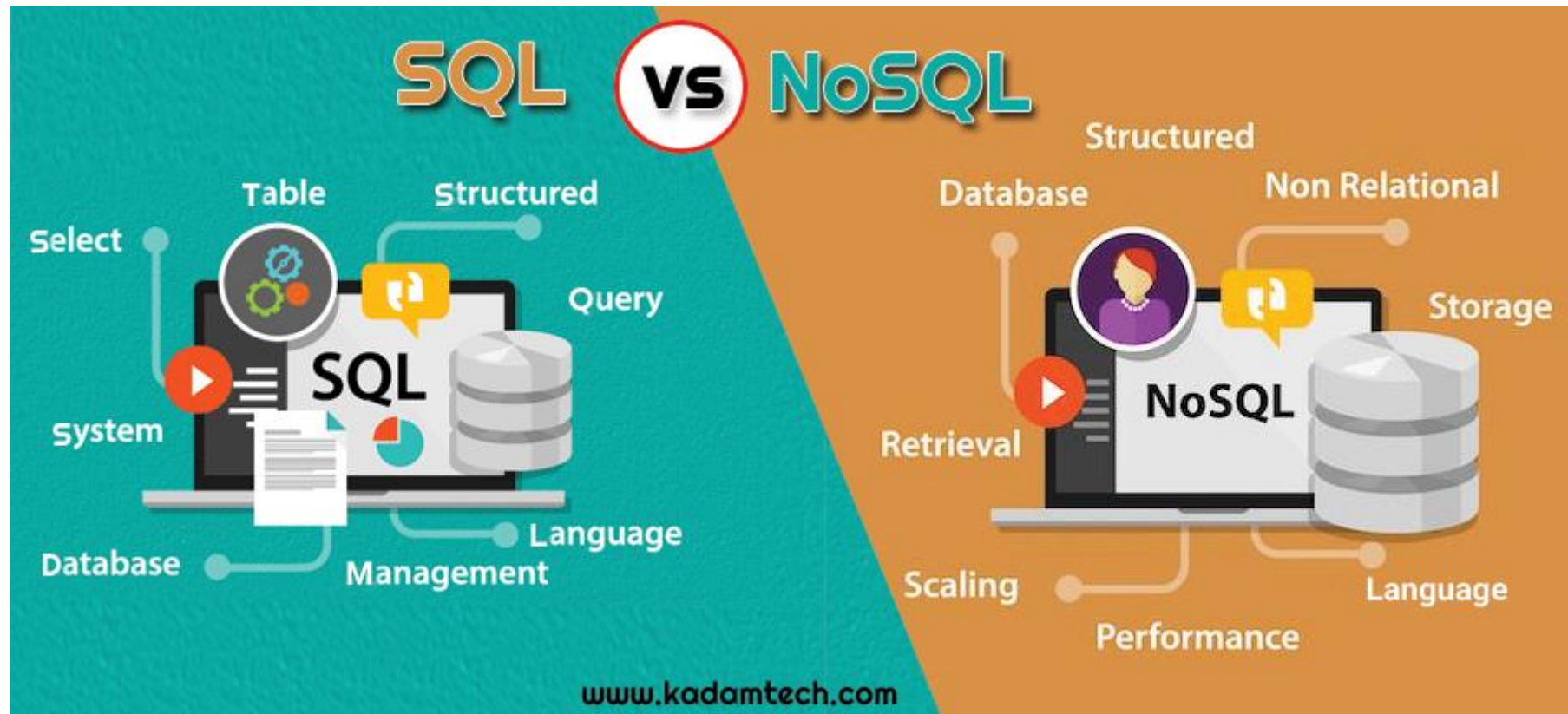
Se você não precisa de grande performance ou tem um sistema de poucos usuários, usar Redis e matar uma mosca com canhão.

Pub/Sub	<p>1. SUBSCRIBE (from Consumer A and Consumer B to Redis) 2. PUBLISH (from Producer to Redis) 3. data (from Redis to Consumer A and Consumer B)</p>	<ul style="list-style-type: none"> 1. at-most-once delivery 2. no data persistence
List	<p>RPUSH (from Producer to Redis) BLPOP (from Redis to Consumer A and Consumer B)</p> <p>Consumer Group (enclosing Consumer A and Consumer B)</p>	<ul style="list-style-type: none"> 1. at-most-once delivery 2. can be persisted on disk 3. can form a consumer group
Stream	<p>1. XADD (from Producer to Redis) 2. XREAD <id> (from Consumer A to Redis) 2. XREAD \$ (from Consumer B to Redis) 2. XREAD 0-0 (from Consumer C to Redis) 3. data (from Redis to Consumer A, B, and C)</p>	<ul style="list-style-type: none"> 1. at-least-once delivery 2. can be persisted in Redis RDB 3. can form a consumer group

Com um sistema com dezenas, milhares ou milhões de usuários ou processando milhares de requisições por minuto, lidando com tabelas envolvendo milhões de registros que precisam ser acessados muito rapidamente, começa ficar interessante você buscar apoio em tecnologias como a do Redis.



O Redis faz parte de uma família de tecnologia que nós chamamos de NoSQL. O NoSQL é uma família, um grupo de mecanismos de persistência focados em outro aspectos, que não aquela persistência mais rígida de dados que estamos acostumados nos bancos de dados tradicionais.



Nos SGBD's tradicionais, estamos preocupados com o armazenamento integro, durável, seguro, atômico. Agora na família do NoSQL, geralmente tem outras características que você está buscando, como no caso do Redis, a velocidade, nem tanto a persistência permanente da informação. Então o Redis é um dos muitos softwares da família NoSQL.