# Systems Design Interviews at Facebook

## The systems design interview at Facebook focuses on your ability to design a large-scale system end to end. A strong performance in this interview indicates that you would be successful applying your design process to many systems at Facebook.

## What we ask

Some example questions are:

* Design a key-value store
* Design Google search
* Architect a world-wide video distribution system
* Build Facebook chat

We may focus on specific types of systems (search, Ads, distributed learning systems) if you claim expertise in there. We may be more product-focused if that’s what you’ve been working on. We don't expect you to know crazy algorithms that you likely wouldn’t know off the top of your head (like quad trees or Paxos).

## Expectations:

What we’re looking for:

* Can you arrive at an answer in the face of unusual constraints?
* Can you visualize the entire problem and solution space?
* Can you make trade-offs like consistency, availability, partitioning, performance?
* Can you give ballpark numbers on QPS supported, # of machines needed using a modern computer?
* How much have you thought about Facebook and some of the unique problems we face?

A good design shows that you:

* Clearly understand the problem
* Propose a design for a system that breaks the problem down into components, that can be built independently, and you can drill into any piece of the design and talk about it in detail
* Identify the bottlenecks as the system scales and understand the limitations in your design
* Understand how to adapt the solution when requirements change
* Draw diagrams that clearly describe the relationship between the different components in the system
* Calculate (back-of-the-envelope) the physical resources necessary to make this system work

## How to study

This is the hardest interview to study for.

To practice, take any well-known app and imagine you work for a competitor. Your job is to figure out 1) where are the most important problems for their system and 2) what are the constraints for their system. Answering these questions will help you focus on the most important problems and not worry about solving problems that are not relevant.

For example, Google’s index building layer has many more components for document understanding. It would need components for extracting deep links, contact information, referrals (for page rank). On the other hand, Twitter’s index building should be simpler due to small size tweets and some rich media information for attached media. Twitter’s search is head heavy. So a bulk of engineering efforts in designing their search should go to rapidly indexing new tweets and making them searchable.

Work out the above problems on a paper and just think about the ways to break them down. It also helps to read up on common large scale systems, like watch the public videos about memcached and learn how search engines work. But during the interview, don’t parrot back what you read; make sure your solution actually answers the question being asked.