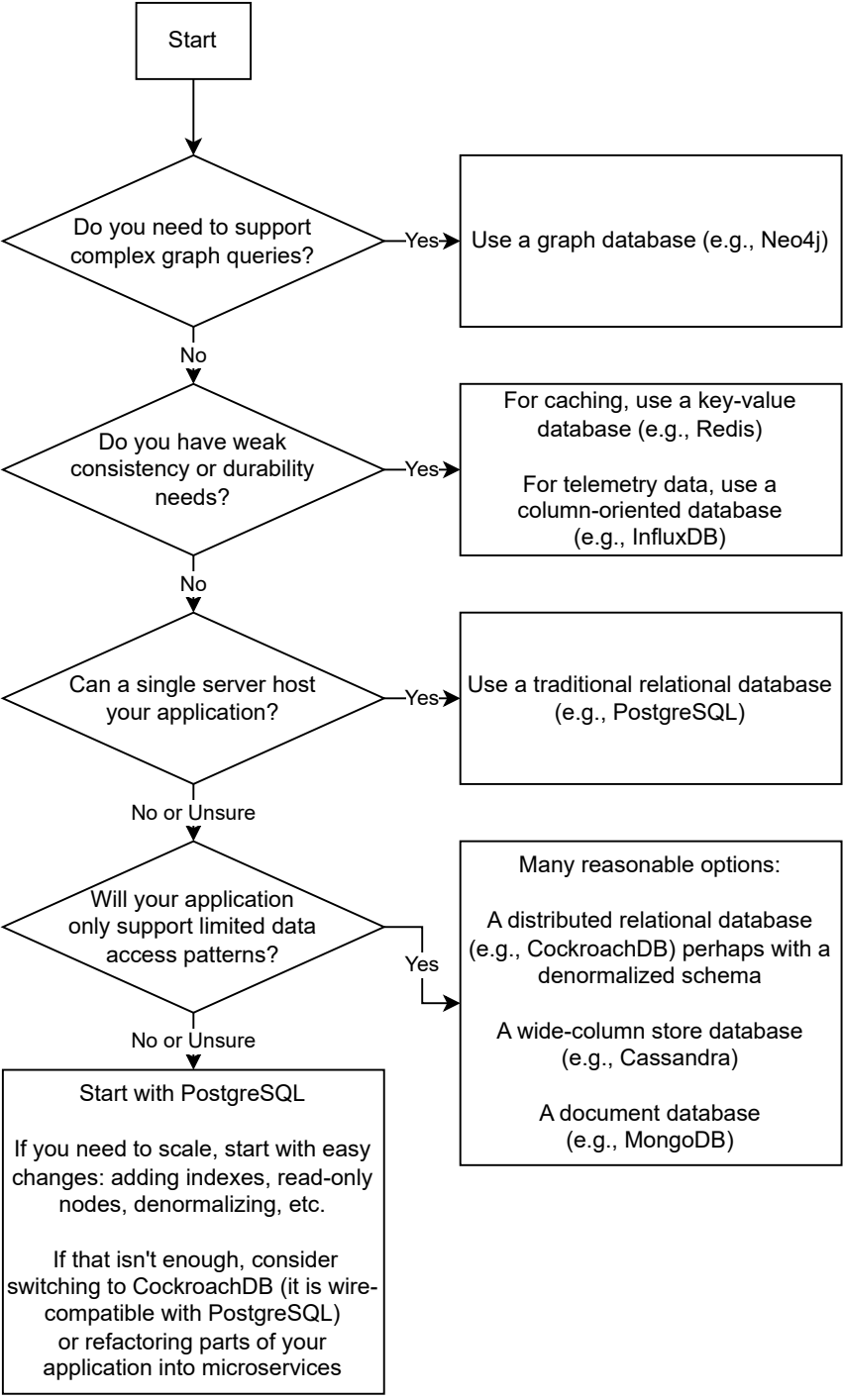
Start
Do you need to support complex graph queries?
Yes
Use a graph database (e.g., Neo4j)
No
Do you have weak consistency or durability needs?
Yes
For caching, use a key-value database (e.g., Redis)
For telemetry data, use a column-oriented database (e.g., InfluxDB)
No
Can a single server host your application?
Yes
Use a traditional relational database (e.g., PostgreSQL)
No or Unsure
Will your application only support limited data access patterns?
Yes
Many reasonable options:
A distributed relational database (e.g., CockroachDB) perhaps with a denormalized schema
A wide-column store database (e.g., Cassandra)
A document database (e.g., MongoDB)
No or Unsure
Start with PostgreSQL
If you need to scale, start with easy changes: adding indexes, read-only nodes, denormalizing, etc.
If that isn't enough, consider switching to CockroachDB (it is wire-compatible with PostgreSQL) or refactoring parts of your application into microservices