软件开发教父Martin Fowler对微服务的定义 https://martinfowler.com/articles/microservices.html

一个完整的微服务系统，应该包含哪些功能？ http://www.infoq.com/cn/articles/what-complete-micro-service-system-should-include

微服务架构技术栈选型手册 http://www.infoq.com/cn/articles/micro-service-technology-stack

A good series of articles talking about Micro Service:

https://www.nginx.com/blog/building-microservices-inter-process-communication/

Spring Cloud: <https://github.com/dyc87112/SpringCloud-Learning/tree/master/2-Dalston%E7%89%88%E6%95%99%E7%A8%8B%E7%A4%BA%E4%BE%8B>

https://m.imooc.com/article/23679

Http2.0 : https://www.zhihu.com/question/34074946

Netty : <https://www.jianshu.com/p/b9f3f6a16911> <https://www.zhihu.com/question/24322387>

Protocbuff website : <https://developers.google.com/protocol-buffers>

<https://github.com/google/protobuf/>

Protocolbuffer learning : <https://www.jianshu.com/p/1538bf85dad1>

gRPC: <https://grpc.io/docs/>

<https://github.com/grpc>

<http://dockone.io/article/2836>

Code generator cmd: protoc -I=. --python\_out=./ ./grpc-helloworld.proto

Netty: <https://www.jianshu.com/p/b9f3f6a16911>

gRPC Spring-boot Starter

<https://github.com/yidongnan/grpc-spring-boot-starter/blob/master/README-zh.md>

<https://github.com/yidongnan/grpc-spring-boot-starter/tree/master/examples>

kafka : <http://ifeve.com/kafka-1/>

CAP: <https://www.zhihu.com/question/54105974?from=profile_question_card>

Zookeeper: <https://yq.aliyun.com/articles/599997>

Eureka: <https://www.iyunv.com/thread-421988-1-1.html>

Eureka register: <https://www.jianshu.com/p/36eeb724de74>

The spread and synchronization of Eureka Server is directional

multi-eureka:https://www.jianshu.com/p/1d683e6404af

zookeeper distribute lock : <https://github.com/code4wt/distributed_lock>

<https://www.jianshu.com/p/36eeb724de74>

扩展不好

运行时支撑服务

* 服务注册中心

Eureka, Consul, Dubbo

* 服务网关

Zuul, Kong

* 配置中心

Config, Apollo

服务监控

* 日志监控

ELK

* 调用链

CAT，ZipKin, Pinpoint

* Metrics监控

OpenTSDB, InfluxDB, Prometheus

* 服务容错

Hystrix

后台服务

* 消息系统

Kafka, hermes, RocketMQ, RabbitMQ

* 缓存治理

Redis

* 分布式数据访问

Shardingjdbc, MyCAT

* 任务调度系统

xxl-job, elastic-job

服务安全

* 协议

OAuth, OpenID connect

* 开源授权服务器

Apereo CAS, keycloak, spring cloud security, WSO2

服务部署平台

* 集群资源调度

Kubernetes, mesos, swarm

* 镜像治理

Docker Registry (Vmware harbor)

* 发布平台

Spinnaker

* IAM

Apereo CAS, keycloak

服务设计

* 单一职责原则
* 松耦合
* 领域模型驱动
* 康威定律

服务说明

* 自动化、集中管理的API文档中心