What I Wish I Had Known Before Scaling Uber to 1000 Services

Las tecnologías están creciendo rápidamente y su uso también en 1 año se puede ver un claro crecimiento en el uso de UBER en una ciudad. Para poder hacer tracking a este crecimiento que cada vez se ve incrementado se debe usar los microservicios.

¿Por qué usar Microservicios? Al usar microservicios nos permite formar equipos que se moverán y harán actualizaciones independientemente. Todos poseen un tiempo de actividad por lo que siempre habrá actualizaciones.

Sin embargo, en nuestra época ahora usamos e sistemas distribuidos que más difícil de trabajar que un simple sistema monolítico y que pasa si el sistema falla. Todo tiene un costo por usar microservicios puede que tengas que decidir entre complejidad entre políticas. También se deben mantener estándares ya que algún developer quiera usar su lenguaje, pero su team usa otro.

Al usar microservicios nos permite usar varios lenguajes entre teams. Pero tiene un costo al pasar personas a otro team ya que usan lenguajes diferentes esto hará que se fragmente la cultura de la organización.

Todo se vuelve un RPS lo que hace que los lenguajes s e vuelvan complicadas hacen es que json/rest se complicado, json necesita un esquema. Todas estas tecnologías que se ven genial en pequeña escala. Cuando se ven en gran escala empieza a añadirse un costo.

Al uso de repositorios está el debate en que si uno es bueno o malo. Al tener solo uno es fácil de buscar, pero no se puede ver el track ya que se perderían. Tener muchos repositorios lo hace complicado en el ordenamiento, pero fácil de ver los historiales.

Al usar microservicios cada team desarrolla su software, pero pueden existir dependencias con otros teams lo que causaría un bloqueo al tener que esperar que el team termine para poder continuar por eso la comunicación es importante. Entender que el sistema funciona como un todo y no independientemente.

Los teams pueden querer hacer dashboard para hacer seguimiento a sus problemas, pero cuando se comparan con los otros team pueden llegar a ser diferentes por eso al desarrollar programas es importante estandarizar.

El performance es importante en estos días donde comprar computadoras se hace más barato mientras que los ingenieros más caros es fácil decir que solo compras más equipo para mejorar el performance, pero va a llegar un punto donde se aceptara en la organización y tengas un problema de espacio. Por eso también es necesario poner un SLA para poner un estándar en cuanta optimización.

Surge el problema de fanout cuando hay mucha gente accediendo al servicio y siempre habrá una persona que le demorara más al aumentar la cantidad de persona habrá mas gente que le tarde el servicio para esto sale tracing de los servicios para saber los problemas e intentar solucionarlos.

Se deben hacer testeos de errores, a los usuarios no les gustara ya que pueden perturbar sus operaciones, pero se deben hacer de todas maneras.

En conclusión los microservicios no siempre son algo bueno todo viene con un costo y se tiene que aprender a hacer las decisiones en el momento para decidir la mejor solución y aprender de estos tips que mencionaron para facilitar el uso y la implementación de los microservicios.