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

Es realmente difícil de obtener el exacto número de servicios que están en producción en un determinado momento debido a que cambian tan rápido y hay tantos equipos distintos construyendo todas estas cosas diferentes que cada semana se están produciendo servicios nuevos. Es algo muy bueno romper todos estos monolitos en cosas más pequeñas. Podemos agregar más personas a los equipos, esto permite que los equipos se formen rápidamente y se muevan independientemente

El momento en que las cosas son más propensas a romperse es cuando las cambias, así que uber es más confiable los fines de semana, cuando nuestros ingenieros no están haciendo cambios. Hay un punto donde una parte del software que ha sido desplegado en un micro servicio nunca lo toquemos. Es más probable que cuando un micro servicio se rompa, se construya un nuevo micro servicio en vez de arreglar el antiguo micro servicio

En las épocas prehistóricas de uber había un grupo tercerizado de desarrollo y no lo veíamos como un problema de tecnología, pero en un momento el grupo tercerizado se volvió parte del equipo in-house. Inicialmente nosotros escribimos el código en Python, el equipo que llego usaba Python y java; teníamos diferentes lenguajes y es bueno que los micro-servicios nos permitieran tener equipos escribiendo código en diferentes tipos de lenguaje. Sin Embargo había un problema, si querías re organizar a las personas de un equipo tenías construido un conocimiento en la cabeza de esas personas en una plataforma que no necesariamente se traslada directamente a otra plataforma, hay un costo en mover a las personas de los equipos.

Tener un modelo de micro servicios en todos lados fragmenta la cultura de los diferentes equipos de trabajo