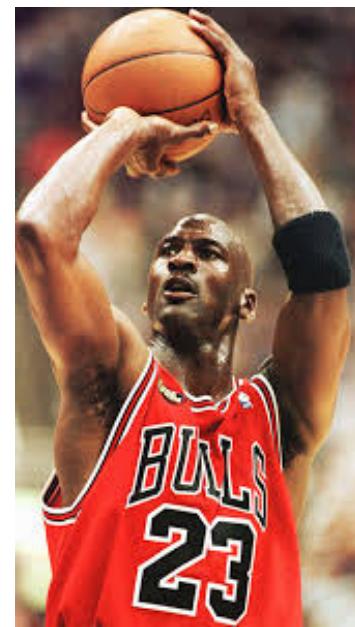


El talento individual te puede
hacer ganar partidos
pero es el trabajo de equipo lo
que te hace ganar
campeonatos



Michael Jordan

Grupo no es lo mismo que Equipo

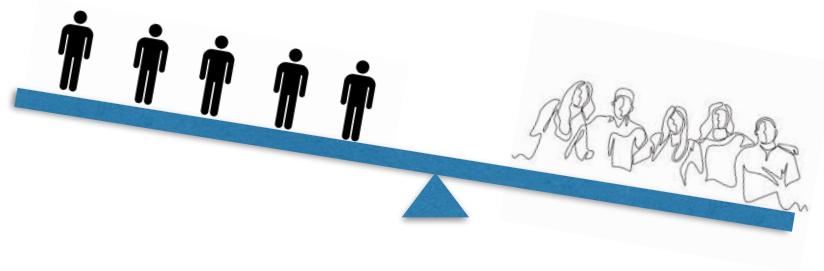
- ▶ Grupo - conjunto de individuos
- ▶ Equipo - conjunto de personas con habilidades complementarias comprometidas con un propósito, una meta y un enfoque donde todos son responsables

Por qué se requieren equipos

- ▶ Permite lograr mucho más en el tiempo disponible
- ▶ Permite contar con diferentes habilidades
- ▶ Permite contar con mayor diversidad de ideas
- ▶ Soporte moral

Qué es un equipo de alto rendimiento

- ▶ Sinergéticos - $1 \times n > n \times 1$
- ▶ Logro del equipo prevalece sobre logro individual
- ▶ Excelente comunicación entre miembros
- ▶ Roles y responsabilidades claras
- ▶ Confianza



Equipos de alto rendimiento exhiben características comunes

- visión u objetivo compartido
- sentido de identidad
- miembros competentes
- compromiso con el equipo
- confianza mutua
- interdependencia entre miembros
- comunicación efectiva
- sentido de refuerzo (apoyados desde arriba)
- tamaño del equipo pequeño (máximo 10)
- miembros disfrutan de su trabajo

Visión u objetivo compartido



- ▶ poner un hombre en la luna en esta década
- ▶ completar el modelo standard (bosón de Higgs)
<https://www.youtube.com/watch?v=akCJc7K3DUU>
- ▶ construir un computador personal insanamente grandioso (Macintosh)

Identidad



- ▶ un nombre común (Ejemplo: piratas en caso de Macintosh)
- ▶ camisetas (t-shirts)
- ▶ humor común
- ▶ blocks, mugs, etc

¿Cómo se arma un buen equipo ?

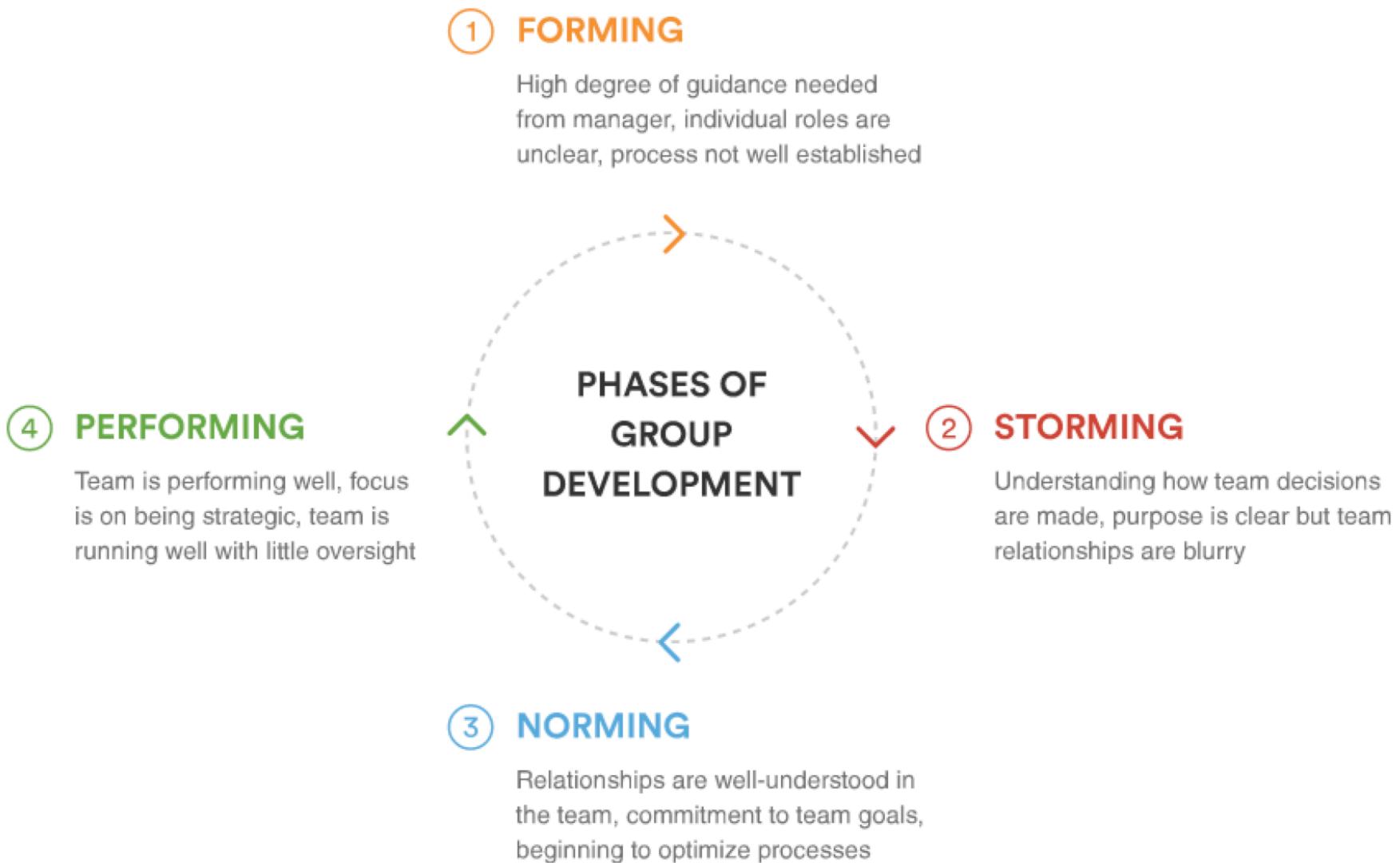
- ▶ los buenos equipos no se arman sino que se "crecen" (como un jardín)
- ▶ igual que un jardín, toma tiempo y cuidado
 - ▶ conócete a ti mismo
 - ▶ cultiva la confianza
 - ▶ cultura de estándares altos



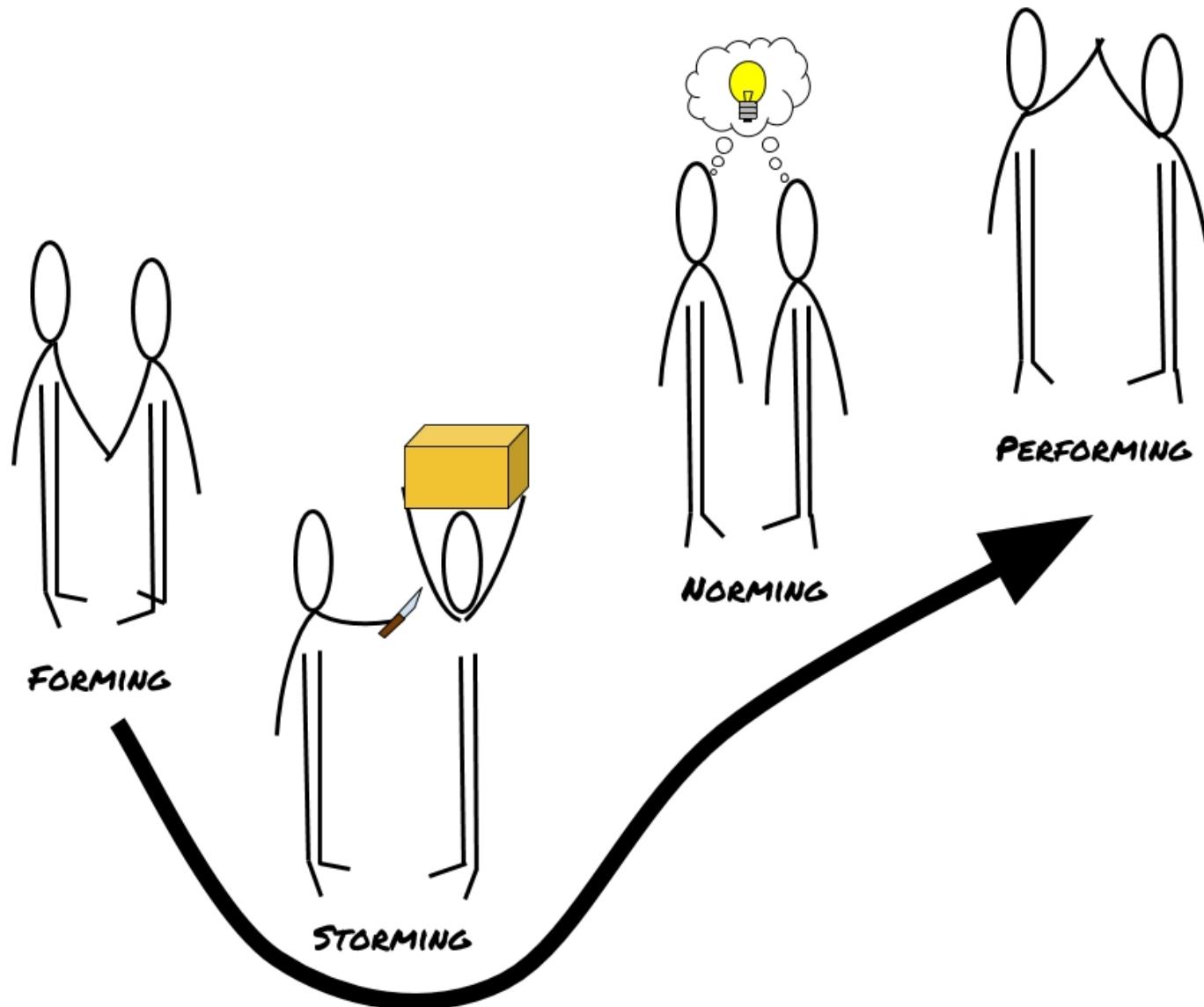
Las 4 fases de maduración



1. Forming - período exploratorio, cautela, exploración de límites
2. Storming - se definen aspectos de poder, control y liderazgo
3. Norming - se establece cohesión, se comienzan a apreciar las diferencias, nace la confianza
4. Performing - sentido de identidad, alto nivel de trabajo, funciona como equipo



Las 4 fases



Se puede acelerar el proceso

- ▶ Forming
 - ▶ claridad en metas y objetivos
 - ▶ claridad en responsabilidades
 - ▶ identificar mecanismos de comunicación
 - ▶ validar habilidades que trae cada uno
- ▶ Storming
 - ▶ asegurar que todos participen
 - ▶ establecer normas de discusión y resolución de conflictos
- ▶ Norming
 - ▶ dejar que miembros vayan tomando mayores responsabilidades
 - ▶ graficar el progreso y recompensar los éxitos
 - ▶ reducir tiempo de reuniones

Algunos Modelos de Equipos

- ▶ Según tarea predominante
 - ▶ resolución de problemas
 - ▶ creatividad
 - ▶ ejecución táctica (SWAT)
- ▶ Modelos de equipo atlético o grupo de teatro
- ▶ El modelo de "chief programmer team"

Tarea Predominante

- ▶ Resolución de Problemas
 - ▶ problema complejo poco definido
 - ▶ Ej: desarrollo de solución ad-hoc para un problema específico
- ▶ Creatividad
 - ▶ exploración de posibilidades y alternativas
 - ▶ liberarse de restricciones
 - ▶ pensamiento libre
 - ▶ Ej: idea para un nuevo procesador de texto revolucionario
- ▶ Ejecución Táctica
 - ▶ ejecución de un plan bien definido en la forma más eficiente posible
 - ▶ Ej: desarrollo de software perfectamente especificado



Modelo Equipo Atlético

- ▶ los jugadores (programadores) son las estrellas
- ▶ papel de líder es sacar obstáculos y crear ambiente para que los desarrolladores trabajen en forma eficiente (no juega)
- ▶ jugadores pueden especializarse
 - ▶ arquitectura
 - ▶ reusabilidad
 - ▶ plataformas específicas



Modelo Grupo de Teatro



- ▶ dirección más fuerte
- ▶ director mantiene visión del producto y asigna responsabilidades
- ▶ negociación de papeles
- ▶ participantes tienen cierto margen de acción (latitud) pero menor que en el caso del equipo atlético

El "Chief Programming" team

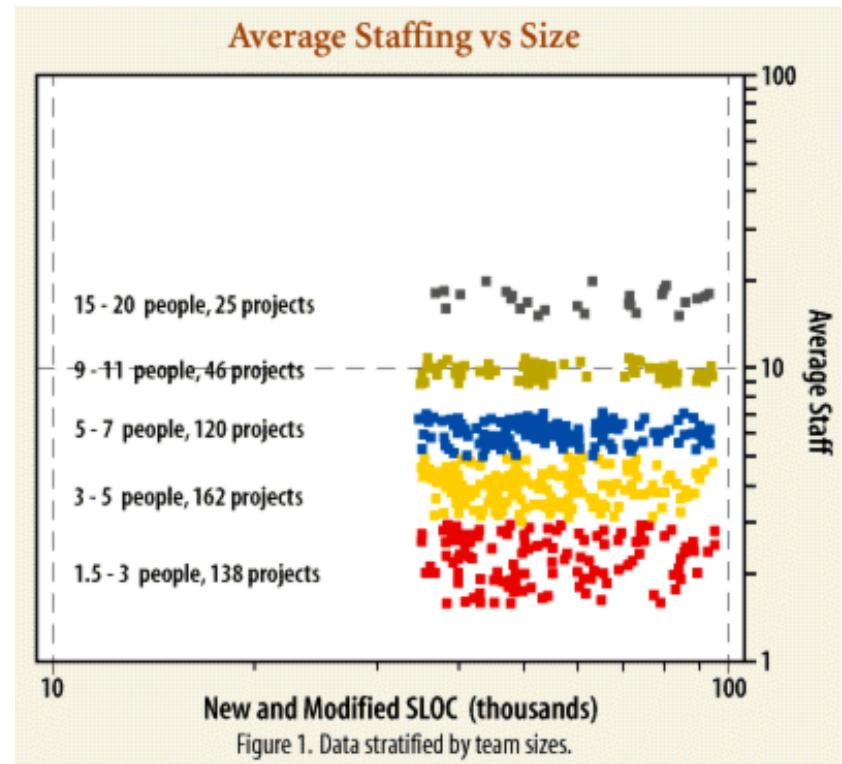


- ▶ aprovecha idea de programadores altamente productivos (10/1)
- ▶ programador jefe hace el diseño y escribe la mayor parte del código
- ▶ resto del equipo cumple funciones de apoyo
 - ▶ administrador (asuntos administrativos)
 - ▶ especialista en herramientas
 - ▶ especialista en el lenguaje de programación
 - ▶ programadores secundarios
- ▶ tremendamente efectivo si uno cuenta con un programador estrella



¿Tamaño del Grupo ?

- ▶ Estudio sobre 491 proyectos medianos (35 a 95 KLOC)
- ▶ Equipos de desarrollo de distintos tamaños
- ▶ Se estudió
 - ▶ productividad
 - ▶ tiempo de desarrollo
 - ▶ esfuerzo



Average Productivity Index

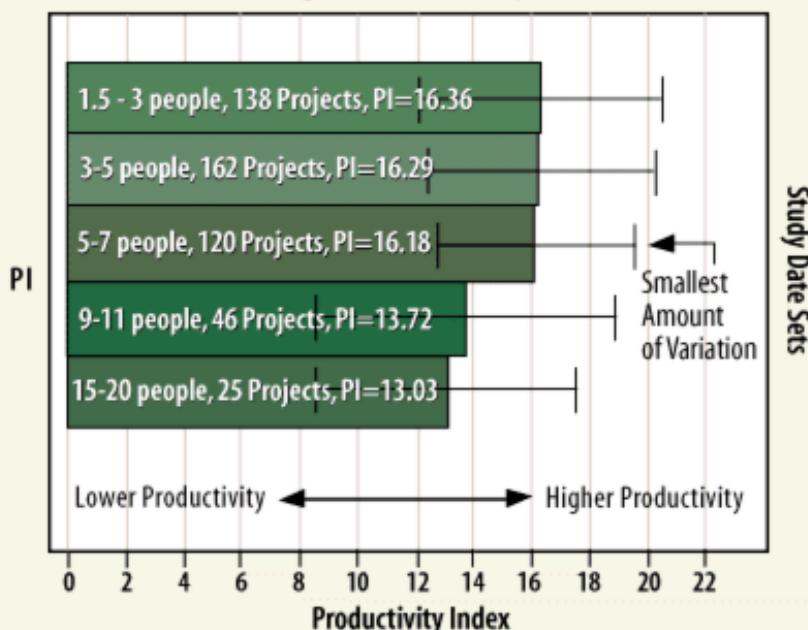


Figure 2. Average Productivity Index for each project staffing group with variation bounds.

Average Schedule Months

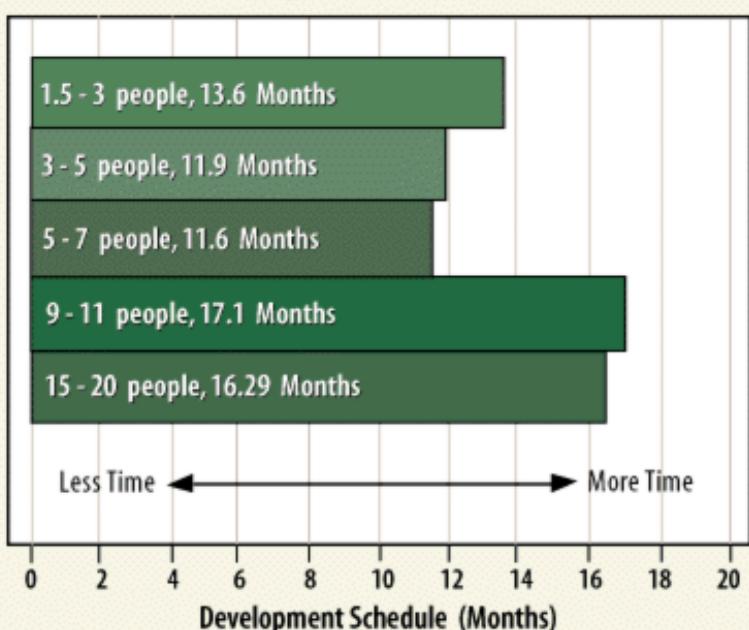


Figure 3. Average Schedule for each project staffing group

Development Effort

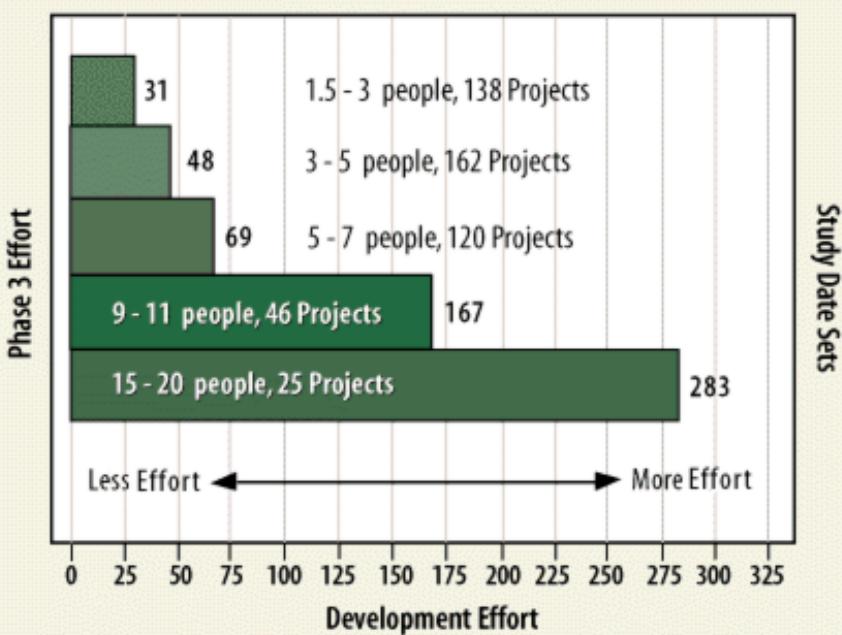
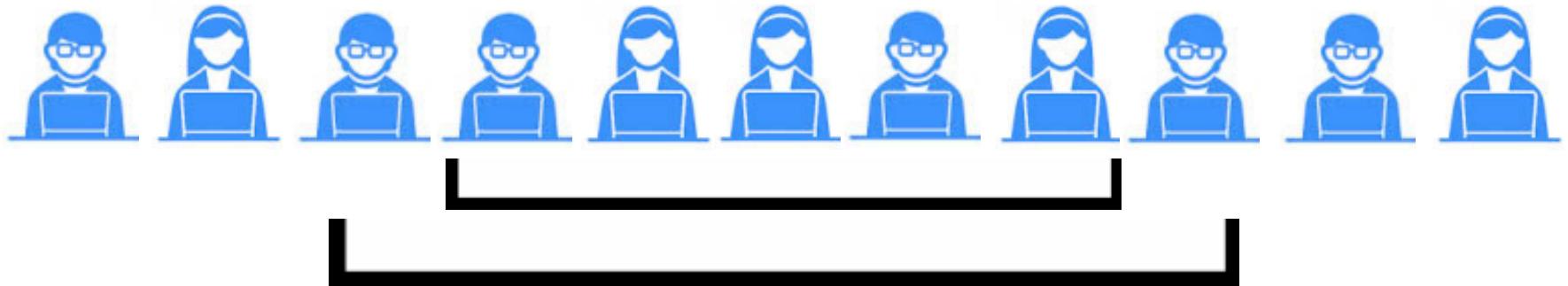


Figure 4. Average effort for each project staffing group.



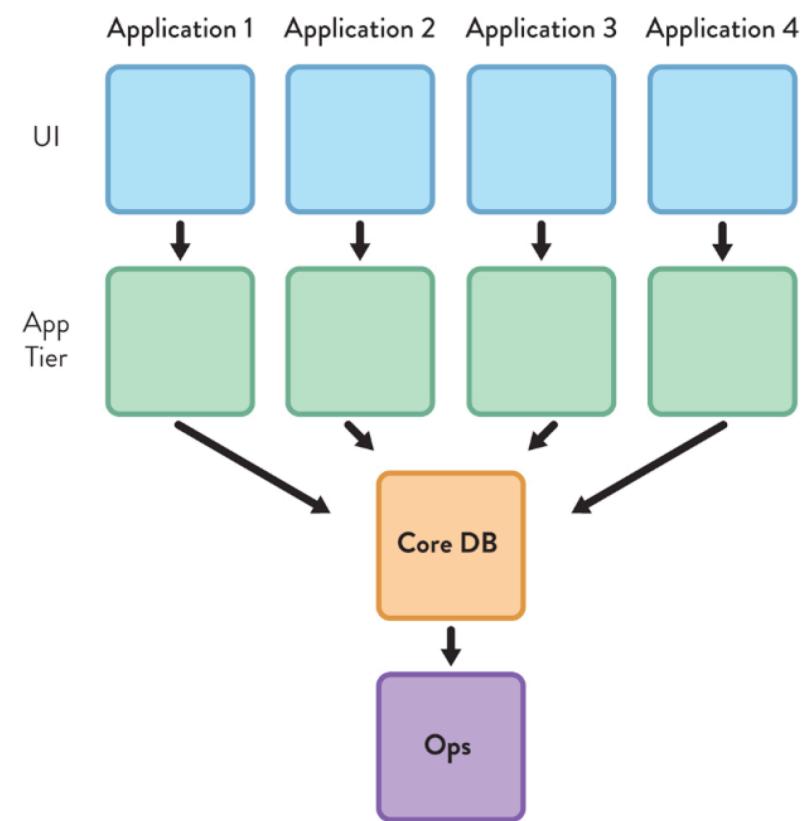
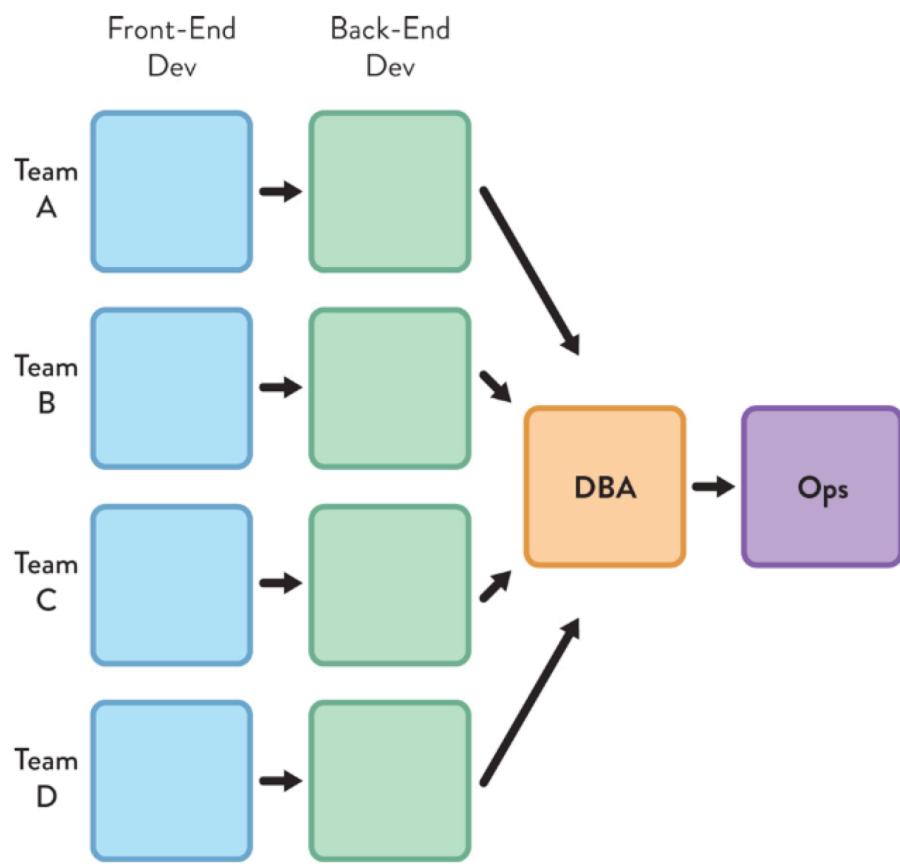
- ▶ Para proyectos medianos equipos de 3 a 7 personas son los mejores (generalmente 3 a 5 es incluso mejor)
 - ▶ equipo provee algo de protección a pérdida de un miembro
 - ▶ más fácil construir cohesión y motivación
 - ▶ menos comunicaciones
 - ▶ menos management

Topologías de Equipos de Trabajo¹

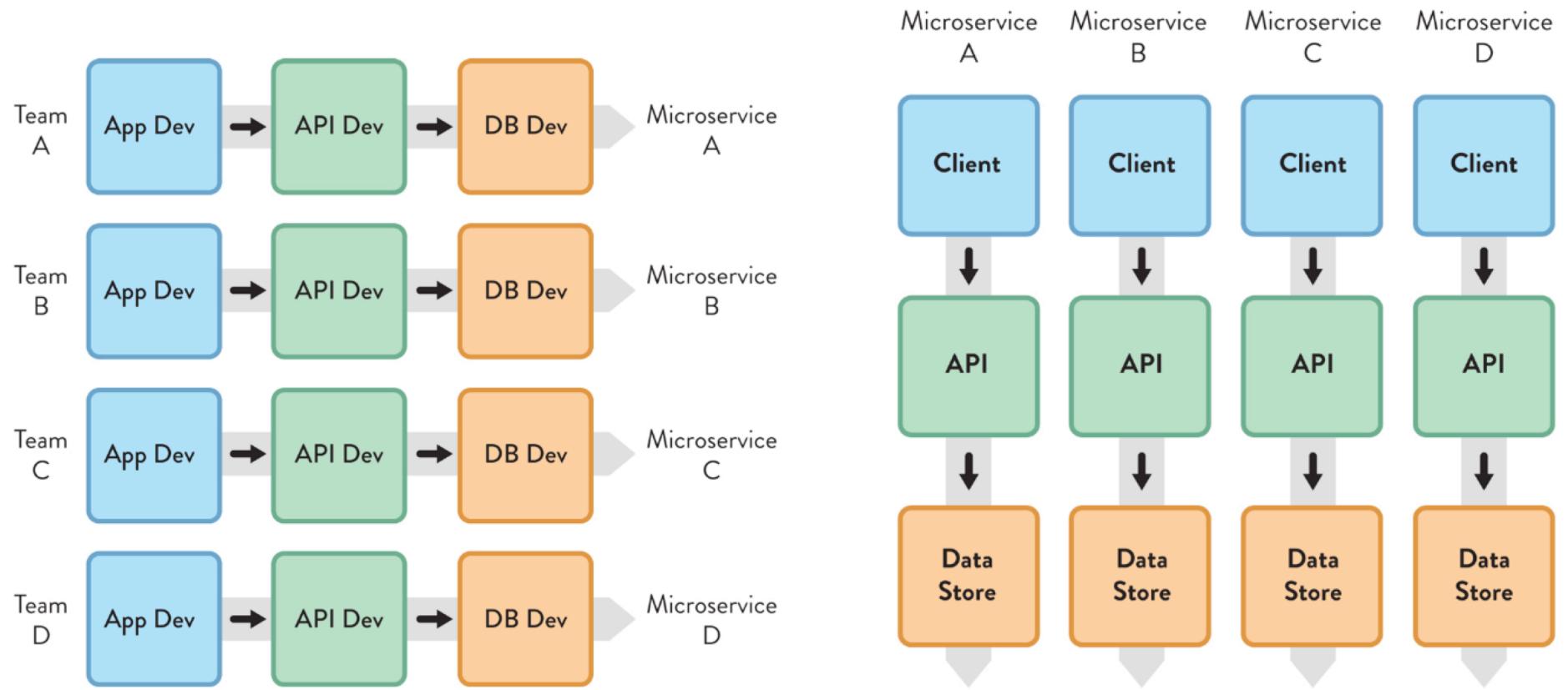
- ▶ La topología de los equipos de trabajo está intimamente relacionada con la arquitectura del software
- ▶ Conway's law "Las organizaciones que diseñan sistemas tienden a producir diseños que son copias de las estructuras de comunicación de ellas"

1. M. Skelton and M. Pais "Team Topologies", IT Revolution, 2019

Ejemplo: Organización y Arquitectura

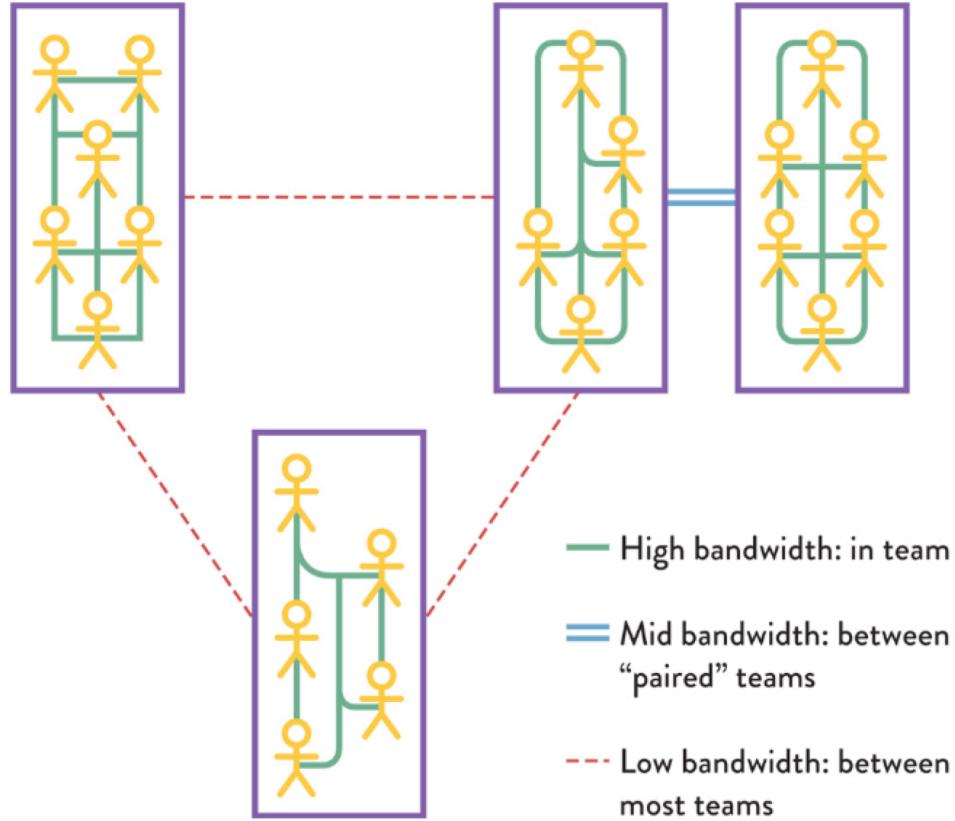


Otra posibilidad ...



Comunicación entre Equipos

- ▶ El tema de la comunicación ha sido indentificado como un problema clave en el desarrollo de software
- ▶ Se supone que se mantiene un equipo pequeño para facilitar las comunicaciones pero ...
- ▶ A veces es necesario comunicarse con otros equipos

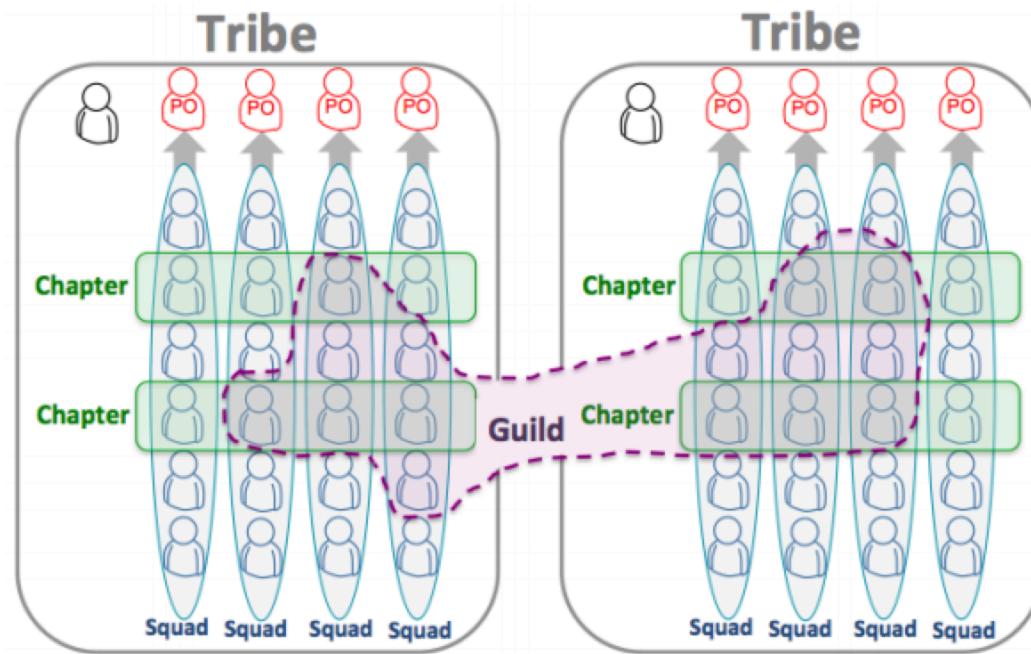


- ▶ Comunicación al interior del equipo de alto ancho de banda
- ▶ Comunicación entre equipos pareados de ancho de banda intermedio
- ▶ Entre equipos cualquiera de bajo ancho de banda
- ▶ Mucho es esto se puede manejar con ubicación física y herramientas de colaboración

Los equipos requieren construir confianza

- ▶ ... y eso requiere limitar el tamaño del equipo en la mayoría de los casos a 5-8 personas
- ▶ es deseable tener equipos diversos (géneros, backgrounds, etc)
- ▶ pueden haber familias o tribus de equipos de no mas de 50 personas
- ▶ pueden haber divisiones con no mas de 150 personas

El modelo de Spotify



- squads : equipos pequeños autónomos (5 a 9)
- varios squads que trabajan en áreas similares forman una tribe (por afinidad)
- los squads en una tribe estan familiarizados con el trabajo de otras squads en ella
- ingenieros en una tribe con habilidades similares (por ejemplo testers) forman chapters
- los guilds son voluntarios (por ejemplo Apple fans, o Latex fans)
- leer todos los detalles en el artículo (sitio del curso)

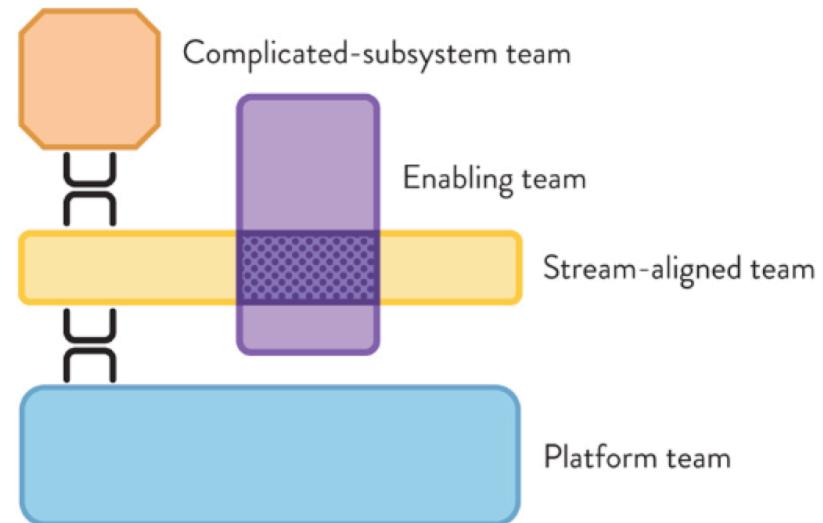
Las 4 topologías fundamentales

- ▶ stream-aligned team

- ▶ enabling team

- ▶ platform team

- ▶ complicated-subsystem team



Stream-aligned teams

- ▶ stream-aligned teams
 - ▶ el principal tipo de equipo
 - ▶ este equipo es responsable de llevar un flujo de trabajo de valor para el negocio desde el inicio hasta el cliente
 - ▶ puede ser un solo producto o servicio, un conjunto, un viaje del cliente, etc.
 - ▶ equipo es empoderado para entregar valor rápido en forma lo más independiente posible
 - ▶ cercanos al cliente y rápidos en incorporar el feedback recibido

Enabling Teams

- ▶ stream-aligned teams trabajan bajo presión de entregar valor (no pueden investigar, aprender nuevas tecnologías, etc)
- ▶ enabling-team compuesto de especialistas en un dominio técnico (o un producto)
- ▶ capacitados para probar opciones y hacer sugerencias sobre herramientas, frameworks, etc
- ▶ fuertemente colaborativos (deben estar alertas y ser proactivos a entender lo que está pasando en los streams)
- ▶ mensajeros de las buenas y malas noticias (hay una nueva herramienta que facilita, framework X deja de ser mantenido)

Complicated-Subsystem team

- ▶ responsable de construir y mantener una parte del sistema que depende fuertemente de conocimientos de especialistas en un área
 - ▶ ejemplos: modelo matemático, algoritmo de reconciliación de tiempo real, motor de reconocimiento facial
- ▶ propósito es reducir la carga de equipos stream-aligned que deben incluir el uso de este subsistema

Platform teams

- ▶ el propósito es habilitar a los streams la entrega de trabajo al proporcionar servicios esenciales
- ▶ esto puede hacerse disponible en forma de capacidades de autoservicio, APIs programables, portal, etc
- ▶ su valor debe ser medido de acuerdo por el valor de los servicios que entrega a los otros equipos

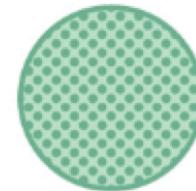
Modos de Colaboración entre Equipos



Collaboration



X-as-a-Service

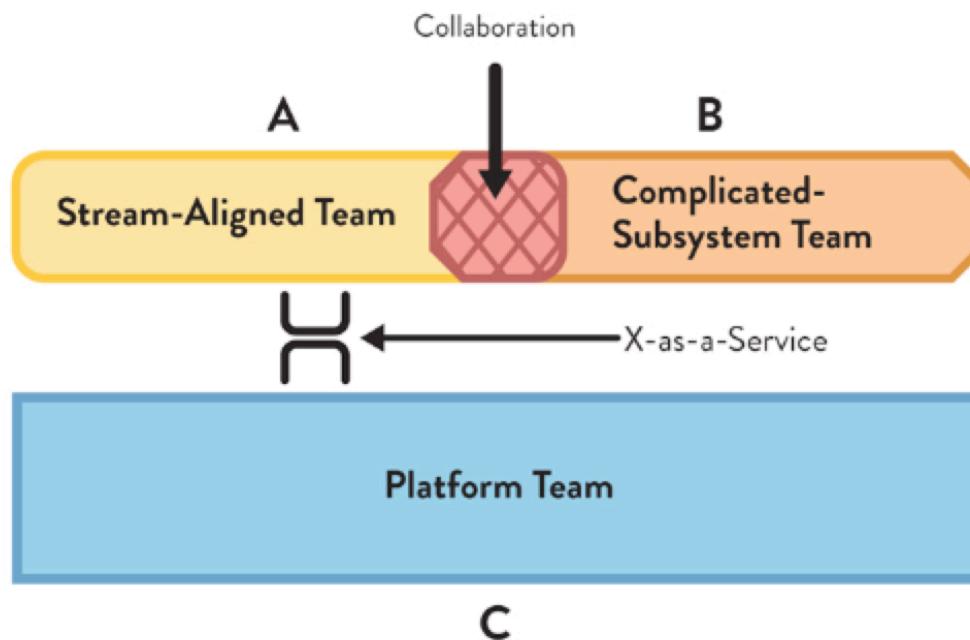


Facilitating

- ▶ Collaboration - trabajo muy cercano con el otro equipo (máximo 1 equipo)
- ▶ X-as-a-Service - se consume o provee un servicio con un mínimo de colaboración
- ▶ Facilitating - se ayuda o se es ayudado por otro equipo

Ejemplo

- ▶ El stream-aligned team A colabora con el complicated-subsystem team B y consume servicios de la plataforma provista por el equipo C



Ventajas y Desventajas de cada modo

- ▶ Collaboration (Alta interacción y respeto mutuo)
 - innovación y descubrimiento rápido
 - poca necesidad de transferir
 - se amplia la responsabilidad de cada equipo
 - se requiere aprender mas cosas
 - posible menor output durante la colaboración
- ▶ X-as-a-Service (Enfatizar la experiencia de usuario)
 - claridad en responsabilidades
 - menos necesidad de conocer detalles de otra cosa
 - menor innovación en torno a la API
 - las APIS pueden no ser muy rápidas o efectivas
- ▶ Facilitation (Ayudame a ayudarte)
 - permite desbloqueo y por lo tanto aumento del flujo
 - permite detectar gaps o desalineación de cosas en componentes y plataformas
 - se requiere staff con experiencia que no va a estar "produciendo"
 - la interacción entre los equipos puede sentirse "rara"

Ejemplo de Interacción Complejo

