



ESTIMACIONES DE SOFTWARE

“PREDICTION IS VERY DIFFICULT, ESPECIALLY
ABOUT THE FUTURE.”

*La predicción es muy difícil, especialmente
acerca del futuro.*

—NIELS BOHR,

Algunas consideraciones

Por definición una estimación no
es precisa.

Estimar no es planear y planear no
es estimar.

Las estimaciones son la base de los
planes, pero los planes no tienen
que ser lo mismo que lo estimado.

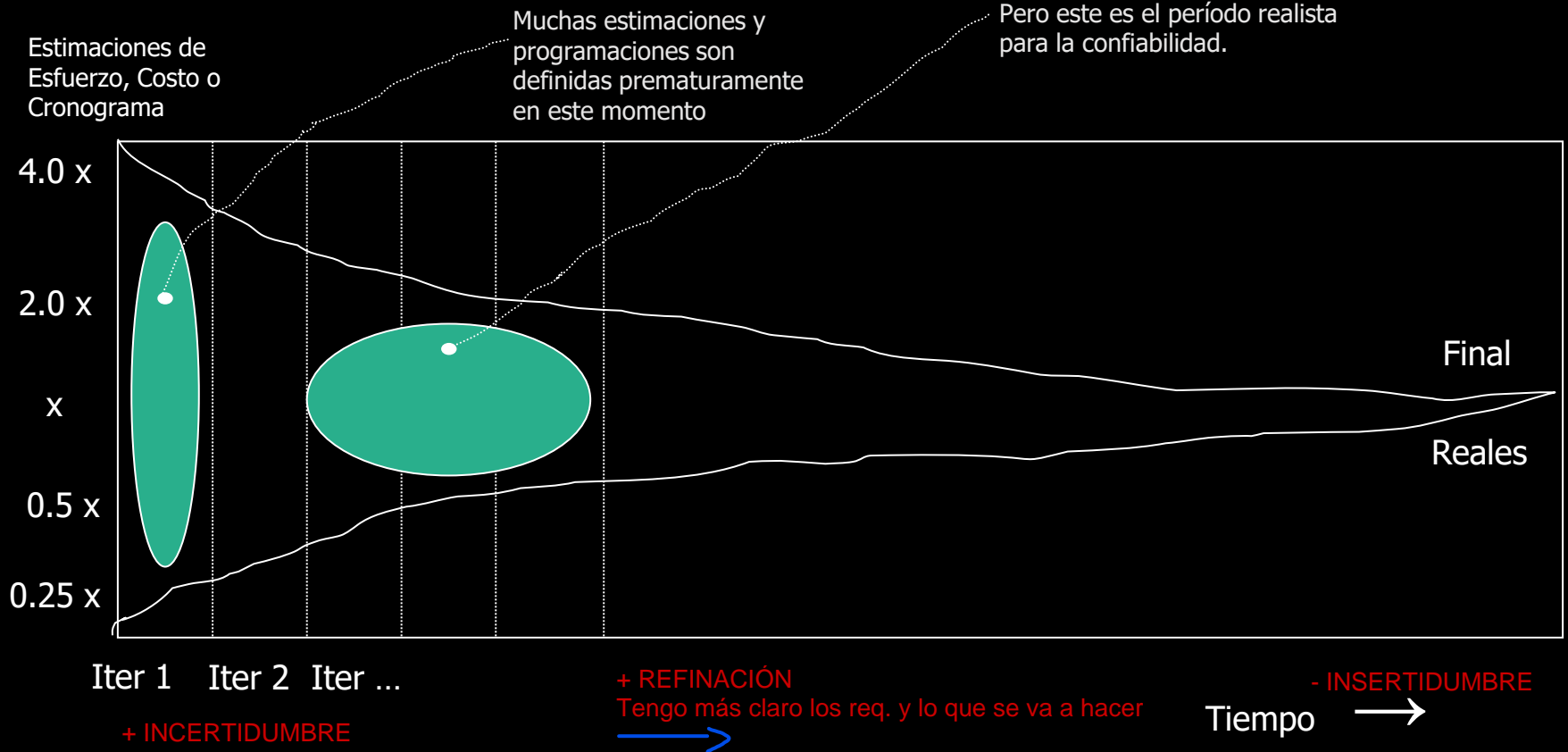
A mayor diferencia entre lo
estimado y
lo planeado mayor riesgo.

Las estimaciones no son
compromisos.

¿Para qué estimamos?

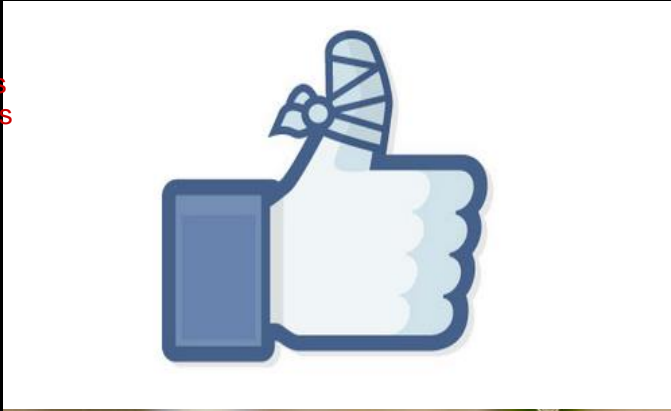
Previsibilidad de lo que va a ser nuestro proyecto

4



¿De dónde vienen los errores de estimación?

Errores en los métodos
de recolección de datos



Demasiado caos en
el proyecto



Falencias en las
personas que hacen
las estimaciones
(no tienen capacitación
necesaria)



Mismos errores que
se hacen el proyecto






Técnicas fundamentales de estimación - Contar

Se realizan en todo momento del proyecto.
En las estimaciones iniciales se cuentan los
requerimientos, casos de prueba, cantidad de
componentes, etc. (todo para estimar cuánto nos
va a costar, el tamaño del proyecto, el esfuerzo,
etc.)

Métodos utilizados

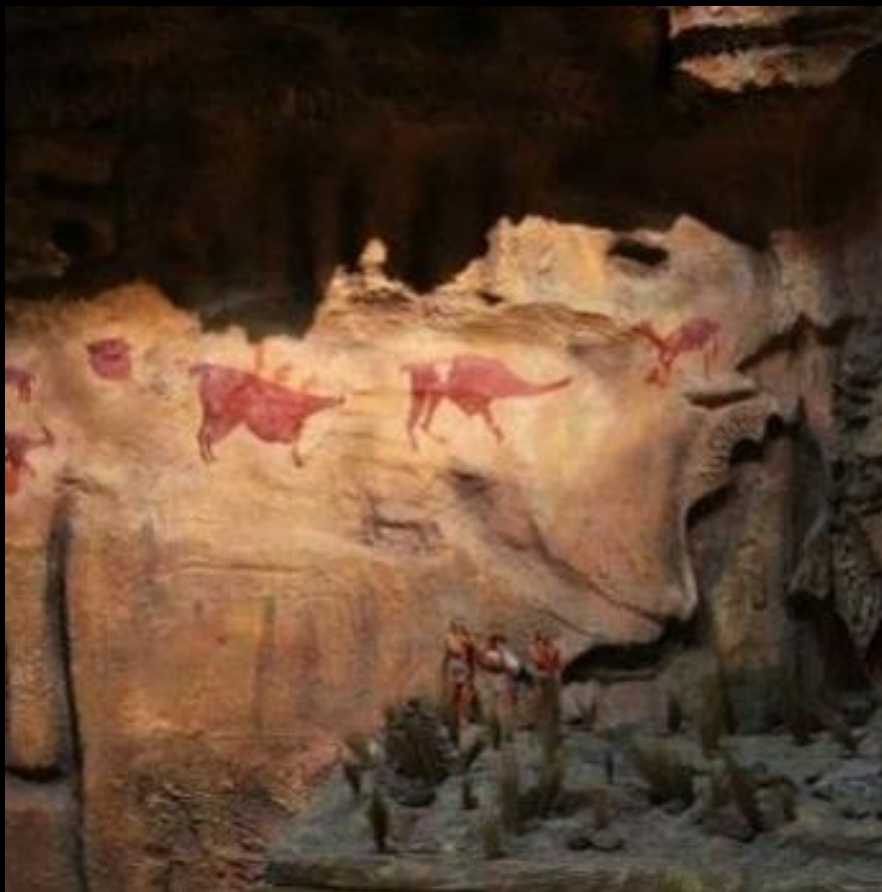
- Basados en la experiencia.
- Basados exclusivamente en los recursos.
- Método basado exclusivamente en el mercado.
- Basados en los componentes del producto o en el proceso de desarrollo.
- Métodos algorítmicos



Métodos
basados en la
experiencia:

- Datos Históricos
- Juicio experto
 - Puro,
 - Delphi
- Analogía

Tener info de proyectos de
mismas características
para tomar como base
para la estimación



Datos históricos

Algunos de los problemas:

- Cómo almacenar estos datos para encontrar los datos que me sirven para estimar
- El ayer no es igual al hoy
- Diferentes tecnologías
- Diferentes trabajadores
- Diferentes tamaños
- Diferentes variables



¿Qué datos históricos necesito?

Para determinar esto, se necesita saber qué - vamos a estimar:

- Tamaño
- Esfuerzo (en horas lineales)
- Meter esas horas de esfuerzo en un calendario, asignar recursos, etc.
- Recursos

Defectos:
- Dependemos del experto

Juicio experto: Puro

- Un experto estudia las especificaciones y hace su estimación.
- Se basa fundamentalmente en los conocimientos del experto.
- Si desaparece el experto, la empresa deja de estimar





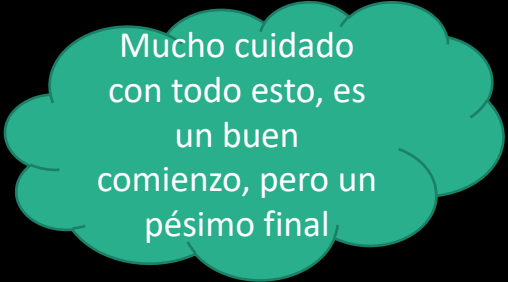
Juicio de Experto

- Es el enfoque de estimaciones más utilizado en la práctica.
- Acerca del 75% de organizaciones de software usan principalmente "juicio de experto"
- Experto en qué?

Estructurando el Juicio de Experto

- Tenga tareas una granularidad aceptable. No hace una sola estimación absoluta
- Use el método de “optimista, pesimista y habitual” y su formula = $(o + 4h + p)/6$ = horas de esfuerzo promedio

Es una guía, no hay que tomarlo como algo que va a pasar si o si



Mucho cuidado
con todo esto, es
un buen
comienzo, pero un
pésimo final

- Use un checklist y un criterio definido para asegurar cobertura.

Juicio experto: Wideband Delphi

No es una técnica democrática, es un consenso

- Un grupo de personas son informadas y tratan de adivinar lo que costará el desarrollo tanto en esfuerzo, como en duración.
- Las estimaciones en grupo suelen ser mejores que las individuales.



Wideband Delphi

Se dan las especificaciones a un grupo de expertos.

Se les reúne para que discutan tanto el producto como la estimación.

Remiten sus estimaciones individuales al coordinador.

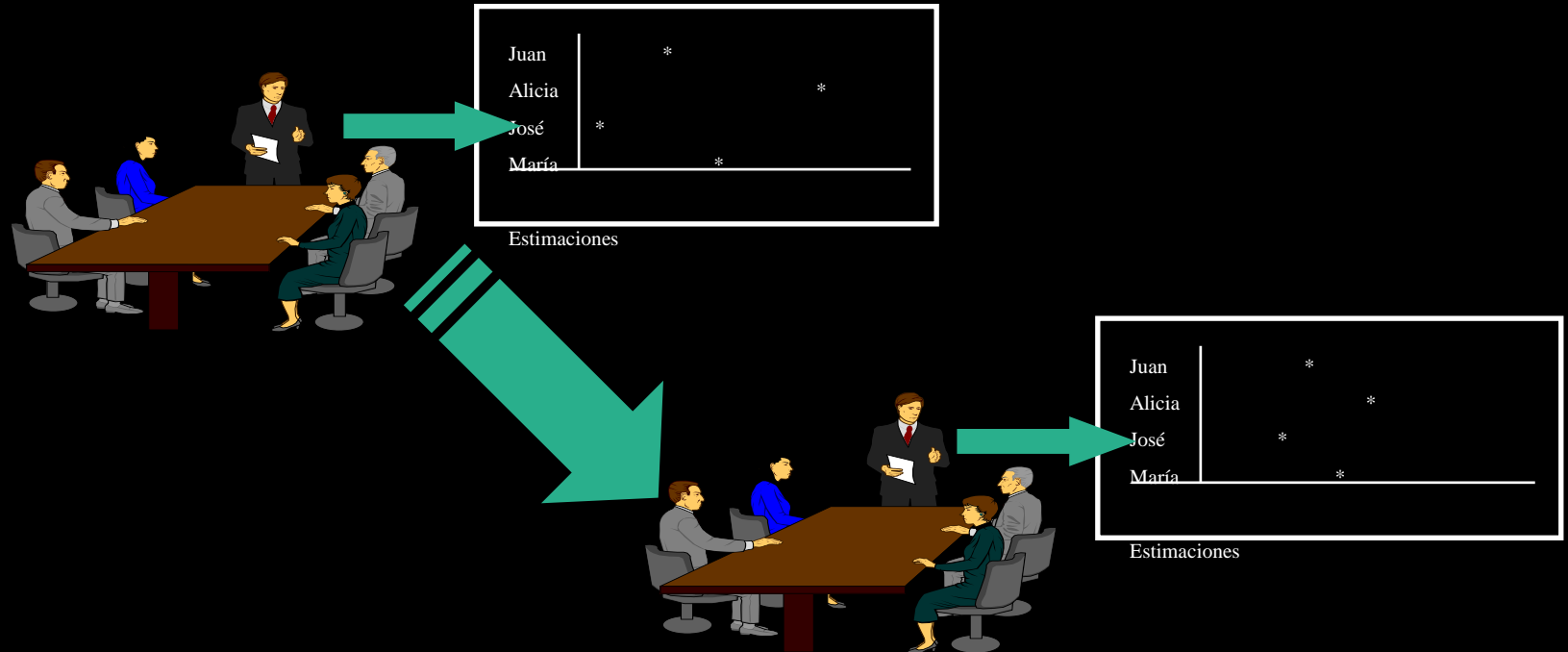
Cada estimador recibe información sobre su estimación, y las ajenas pero de forma anónima.

Se reúnen de nuevo para discutir las estimaciones.

Cada uno revisa su propia estimación y la envía al coordinador.

Se repite el proceso hasta que la estimación converge de forma razonable.

Método de trabajo del Wideband Delphi



Actividades Omitidas

- ❑ Una de la fuentes de error mas común en las estimaciones es omitir actividades necesarias para las estimación del proyecto.
 - ❑ Requerimientos faltantes.
 - ❑ Actividades de desarrollo faltantes (documentación técnica, participación en revisiones, creación de datos para el testing, mantenimiento de producto en previas versiones)
 - ❑ Actividades generales. (días por enfermedad, licencias, cursos, reuniones de compañía).
- ❑ Uso de buffers sirven para compensar los posibles errores de estimación

“Nunca tenga temor que estimaciones creadas por desarrolladores sean demasiado pesimistas, dado que los desarrolladores siempre generan cronogramas demasiado optimistas”.

Chris Peters, Microsoft VP

Bibliografía

- Software Estimation:
Demystifying the Black Art
by Steve McConnell
Microsoft Press 2006
ISBN:0735605351

