

# 5. Seguimiento de Proyectos de Software

Recopilaron parcialmente:

Sergio Barrera Gallegos

María Elena Domínguez Domínguez

Revisó y corrigió:

Ángeles Sumano López

Modificación 2010, 2011:

Juan Manuel Fernández Peña

# SEGUIMIENTO DE CALENDARIZACIÓN

# Seguimiento de la calendarización

- No basta tener calendario y plan, debe revisarse el avance
- Se requiere coleccionar información del avance, de varias formas
- Ayudan las formas gráficas
- Existen formas cuantitativas

# Recolección de información

Características	Tiempo y forma	Comentarios
Oral, formal, regular	Cada semana o mes	Registrar minuta escrita
Oral, formal, ad-hoc	Fin de etapa, revisión	Aunque es oral, se espera entreguen documento; se registra minuta
Escrita, formal, regular	Semanal, Hojas de trabajo, reporte de avance	
Escrita, formal, ad-hoc	Excepciones, cambios	
Oral, informal, ad-hoc	Ocasional, discusión de café, interacción social	Sirve como aviso temprano; debe apoyarse con reporte posterior

# Hojas de tiempo o avance

- Semana
- Actividades por proyecto con duración
- Porcentaje de avance de cada actividad (riesgo de caer en 99% eterno) o preguntar por estimado de probabilidad de terminar en la fecha establecida
- Tiempo no dedicado a sus proyectos
- Puede ser como PSP

# Visualización del avance

# Tabla de recursos

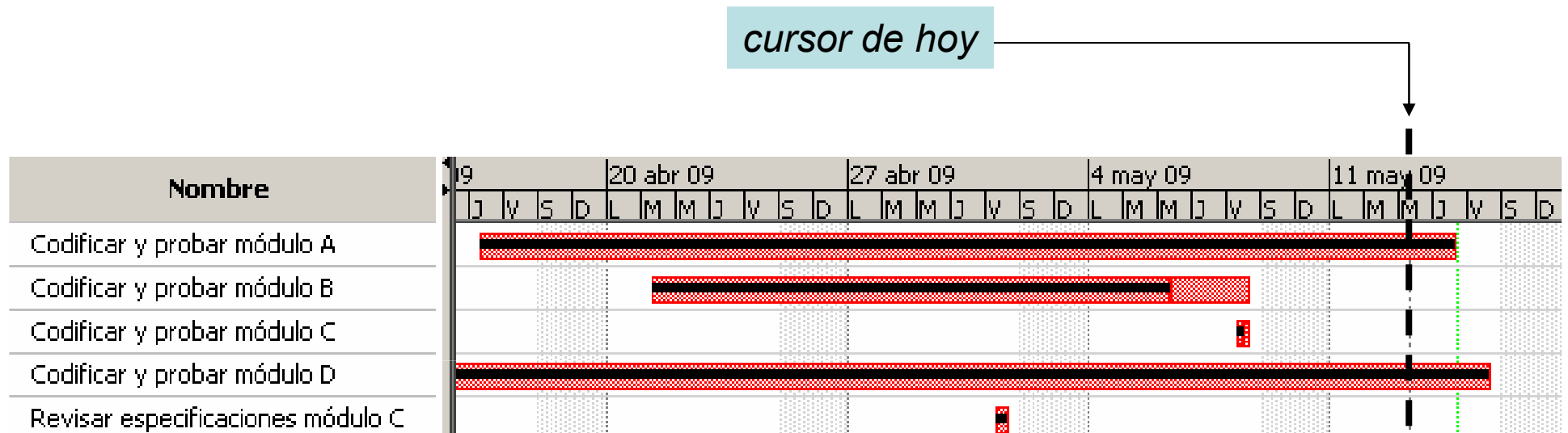
Tareas de Trabajo	Inicio Previsto	Inicio Real	Terminación prevista	Terminación real	Personas asignadas	Esfuerzo asignado	Notas
1.1.1 Identificar necesidades y beneficios							
Reunirse con los clientes	sem 1, d1	sem1, d1	sem1, d2	sem1, d2	BLS	2 pd	La determinación del ámbito requerido mas esfuerzo/tiempo
identificar necesidades y restricciones del proyecto	sem 1, d2	sem1, d2	sem1, d2	sem1, d2	JPP	1 pd	
Establecer enunciado del producto	sem1, d3	sem1, d3	sem1, d3	sem1, d3	BLS/JPP	1 pd	
Hito: enunciado del producto definido	sem1, d3	sem1, d3	sem1, d3	sem1, d3			
1.1.2 Definir salidas/control/entradas (SCE) deseadas							
Alcance de las funciones del teclado	sem1, d4	sem1, d4	sem2, d2		BLS	1.5 pd	
Alcance de las funciones de entrada de voz	sem1, d3	sem1, d3	sem2, d2		JPP	2 pd	
Alcance de los modos de interacción	sem2, d1		sem2, d3		MLL	1 pd	
Alcance del diagnóstico de documentos	sem2, d1		sem2, d2		BLS	1.5 pd	
Alcance de otras funciones del PP	sem1, d4	sem1, d4	sem2, d3		JPP	2 pd	
Documentos SCE	sem2, d1		sem2, d3		MLL	3 pd	
RTF: revisión de SCE con el cliente	sem2, d3		sem2, d3		todos	3 pd	
Modificar SCE conforme se requiera	sem2, d4		sem2, d4		todos	3 pd	
Hito: SCE definidas	sem2, d5		sem2, d5				
1.1.3 Definir la función/comportamiento							

# El gráfico de Gantt

- En las barras originales se agrega barra interior que marca avance.
- Este método informa de los avances que se registran en la gráfica
- “Cursor de hoy”
  - proporciona una indicación visual inmediata de las actividades que están por delante o detrás del programa.

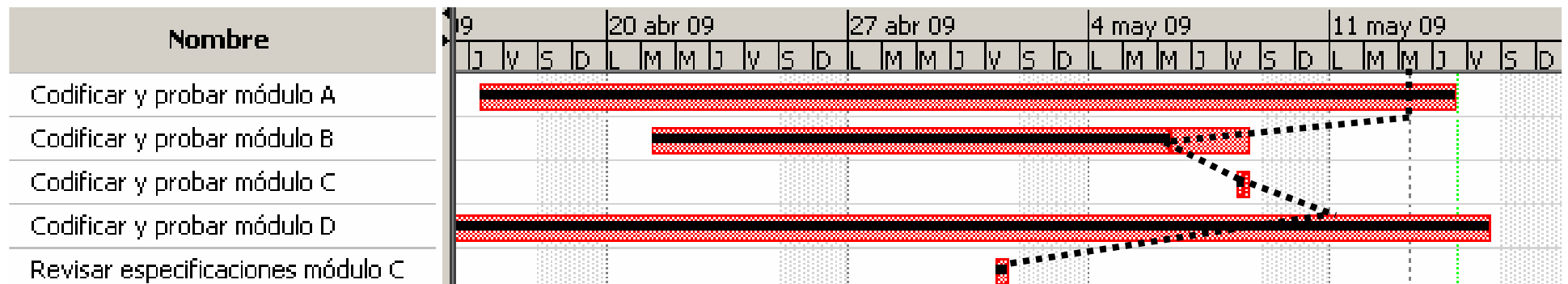


# Ejemplo: Vista de una parte de la Gráfica de Gantt



- Se muestra la parte del gráfico de Gantt, donde:
  - El código y prueba de los módulos A y D han sido terminados por delante del calendario
  - La codificación y prueba de los otros dos módulos están por detrás del programa.

# Gráfico de corrimiento (slip chart)

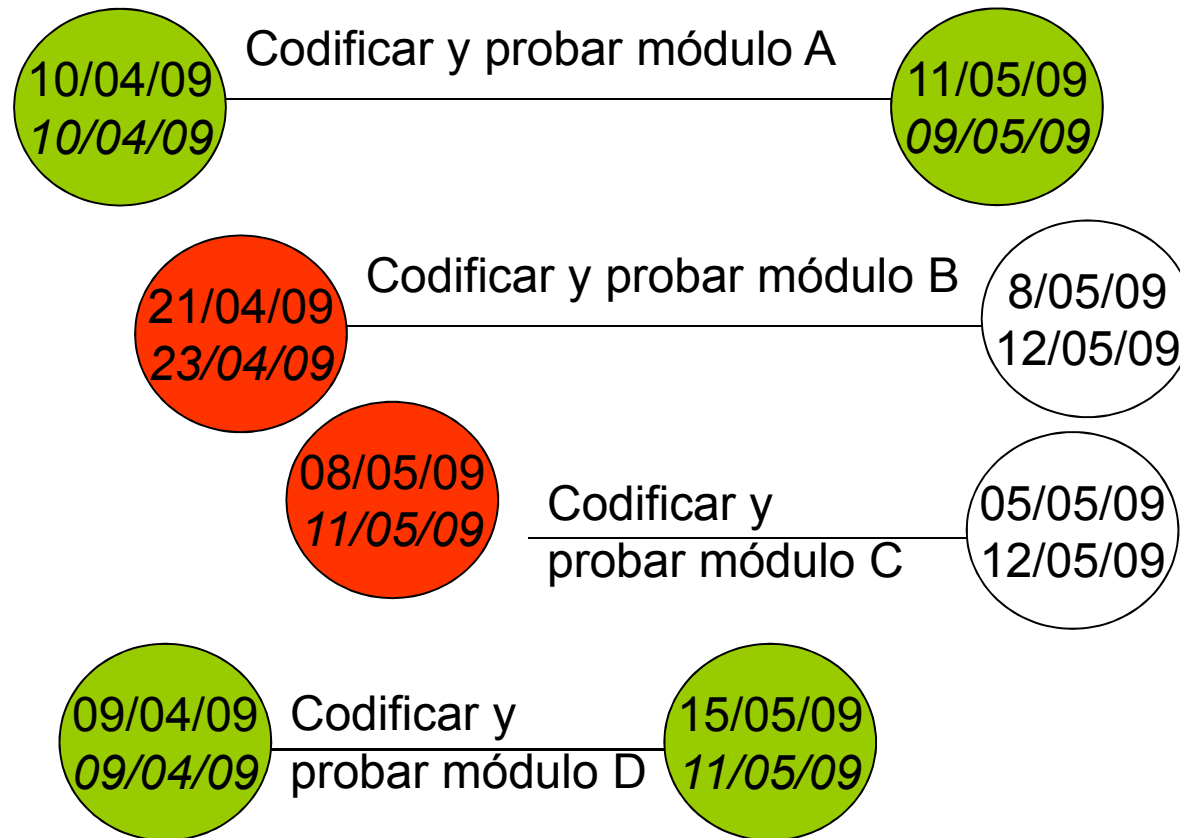


- Proporciona una indicación visual más llamativa de las actividades que no está progresando según el programa, mientras el corrimiento de la línea sea mas pronunciado, mayor es la variación del plan
- Un desliz muy irregular de la línea indica la necesidad de reprogramar

# Ball Chart (1/2)

- Si se desea hacer un poco más llamativo se puede utilizar es utilizar un gráfico con círculos
  - Especialmente útil cuando no se han cumplido los objetivos
- En esta versión de Ball chart, los círculos indican el comienzo y termino de las actividades.
- Los círculos contienen:
  - fechas originalmente previstas.
  - fechas de cuando se producen las revisiones o fecha real (con letras cursivas).
- Si la fecha de inicio o de término de una actividad resulta posterior a la fecha real el circulo se colorea de rojo
- En caso de que la fecha real sea anterior a la planeada, entonces el circulo se colorea de verde

# El gráfico de bola provee incentivos visuales



# Gráfico de la línea de tiempo

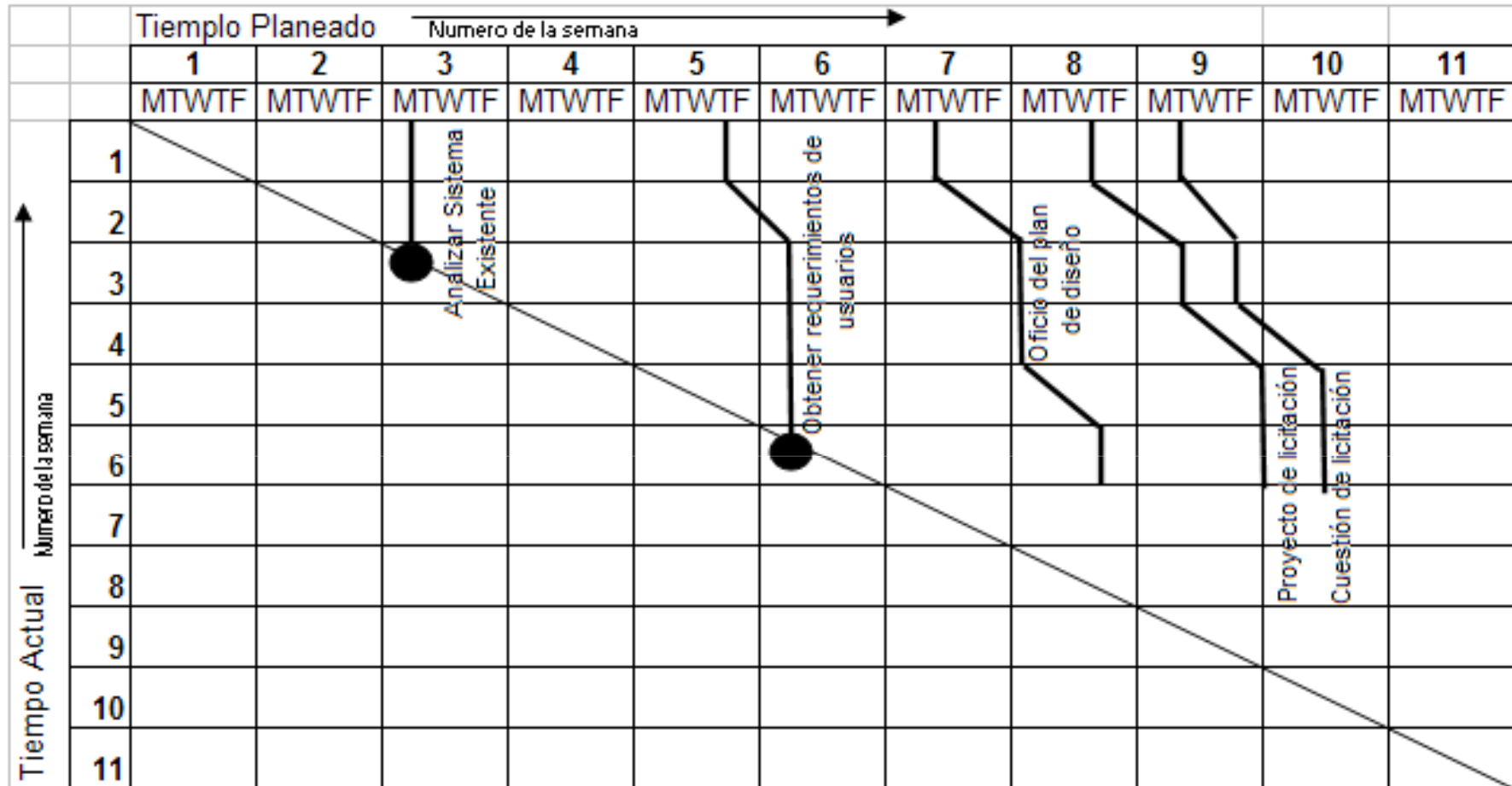
- Métodos anteriores no muestran claramente los corrimientos de fechas.
- La línea de tiempo es un método de grabación y visualización de la forma en que los objetivos han cambiado a lo largo de la duración del proyecto.



# Construcción de línea de tiempo

- Se representarán actividades críticas (ruta crítica y otras importantes)
- Cada actividad se representa como una línea que comienza el día en que debe estar concluida
- Cada semana se actualiza. Si la tarea va bien, su línea crece hacia abajo
- Si se reprograma el final de una tarea, se hace un quiebre hasta la fecha indicada
- Si se termina la tarea se pone un círculo negro sobre la diagonal

# Línea de tiempo al final de la 6ª semana



- Las líneas verticales asociadas a una actividad son rectas si se termina según lo planificado
- Si hay desviaciones de tiempo las líneas de actividad se muestran quebradas y provocan quiebres en las actividades asociadas
- El círculo relleno indica la finalización de una actividad



# Ejercicio

- Para el final de la semana 8 se ha completado el *oficio del plan de diseño*, pero se considera que el proyecto de la licitación va a tener una semana más de lo previsto originalmente.
- ¿Cómo será el gráfico de línea de tiempo al final de la octava semana?
- Si el resto del proyecto va según lo planeado, ¿Cómo será el gráfico de línea de tiempo cuando el proyecto haya sido completado?

# Valor Ganado (1/10)

- La técnica ayuda al Administrador de Proyecto (AP) a:
  - Formar conclusiones rápidas acerca de los niveles de personal y productividad
  - Entender la estructura de la división del trabajo cuando ocurre un problema
- Se compararan tres datos :
  - Valor Planeado (VPL): Cuanto trabajo se tenia planeado terminar hasta ahora .
  - Costo Real (CR): Cuanto se ha gastado realmente hasta ahora
  - Valor Ganado (VG): Cuanto del trabajo planeado se ha terminado hasta ahora
- Nota: Todos se miden en pesos o en horas

# Ejemplo

De la planeación:

Inicio							28 ago '11							04 sep '11						
	Nombre de tarea	Duración	Comienzo	Fin	Pred	No	D	L	M	X	J	V	S	D	L	M	X	J	V	
1	Inicio	7 días	lun 29/08/11	mar 06/09/11																
2	Plática inicial	0.5 días	lun 29/08/11	lun 29/08/11																
3	entrevistas us	1.5 días	lun 29/08/11	mar 30/08/11	2															
4	preparar mode	2 días	mié 31/08/11	jue 01/09/11	3															
5	establecer ver	2 días	mié 31/08/11	jue 01/09/11	3															
6	Revisar con cl	1 día	vie 02/09/11	vie 02/09/11	4,5															
7	terminar espec	2 días	lun 05/09/11	mar 06/09/11	6															
8	Fin inicio	0 días	mar 06/09/11	mar 06/09/11	7															

Suponer: es viernes 2 de septiembre por la tarde y apenas se concluyó la tarea 4; la 5 se planea terminar el lunes.

# Ejemplo

Tarea	Terminada	Valor Planeado	Costo Real	Valor Ganado
1				
2	Sí	4	4	4
3	Sí	12	12	12
4	Sí	16	24	16
5	No	16	24	0
6	No	8	0	0
7	No	16	0	0

Medido en horas-persona

Debería haberse concluido hasta tarea 6

# Ejemplo

Tarea	Terminada	Valor Planeado	Costo Real	Valor Ganado	Valor planeado acumulado	Valor ganado acumulado	Costo real acumulado
1							
2	Sí	4	4	4			
3	Sí	12	12	12			
4	Sí	16	24	16			
5	No	16	24	0	48	32	64
6	No	8	0	0			
7	No	16	0	0			
TOT PLAN		72					

# Valor Ganado (2/10)

- Los dos primeros (VPL y CR) se comparan con VG en forma de diferencias y razones ofreciendo varios indicadores:
  - a) Varianza de Planificación (Schedule):
    - $VP = VG - VPL$
  - b) Varianza de costo:
    - $VC = VG - CR$
  - c) Índice de Rendimiento de Planeación:
    - $IRP = VG / VPL$
  - d) Índice de Rendimiento del Costo:
    - $IRC = VG / CR$

# Ejemplo

<b>Indicador</b>	<b>fórmula</b>	<b>reemplazo</b>	<b>valor</b>
Varianza de planificación	$VP=VG-VPL$	$VP = 32 - 48$	$VP = -16$
Varianza de costo	$VC = VG - CR$	$VC = 32 - 64$	$VC = -32$
Índice de rendimiento de planeación	$IRP= VG/ VPL$	$IRP = 32 / 48$	$IRP = 0.66$
Índice de rendimiento del costo	$IRC= VG/CR$	$IRC = 32 / 64$	$IRC = 0.5$

# Valor Ganado (3/10)

- Las varianzas positivas son favorables y las negativas desfavorables.
  - Si  $VP > 0$  indica que se avanzó más de lo planeado.
- Si  $VC > 0$  indica que se avanzó con menos gastos de lo presupuestado.
- El punto de equilibrio de las varianzas es 0.
  - Valores cercanos a cero indican que el proyecto va de acuerdo al calendario (VPL) y al costo (VC).



# Valor Ganado (4/10)

- Aunque las varianzas sean positivas, si se desvían mucho del punto de equilibrio indica que para futuros proyectos debe mejorar la planificación.
  - Tal vez fuimos demasiado pesimistas.
- Si es negativa, definitivamente debe revisarse la planificación y, tal vez, la estimación del proyecto.
  - Fue demasiado optimista

# Valor Ganado (5/10)

- Los índices mayores a 1 son positivos y los menores a 1 son negativos.
  - IRP  $>1$  indica desarrollo de software mayor al planeado.
  - IRC  $>1$  indica que el desarrollo cuesta menos de lo planeado.
- Los índices tienen su equilibrio en 1.
  - si tienen valores cercanos a 1, indican que el proyecto marcha de acuerdo a lo estimado.
- Si los valores están muy alejados del equilibrio se debe revisar la estimación y planeación.

# Valor Ganado (6/10)

- IRC es un indicador de productividad
- IRP un indicador de avance (progreso)
- Si la productividad es buena y bajo el avance, el proyecto requiere mas gente.
- Si la productividad es baja, entonces pasa una de dos cosas:
  - Hay mucho trabajo no planeado
  - Se estimó mal y el proyecto tiene más trabajo necesario del que se pensó.

# Valor Ganado (7/10)

## Nomenclatura de Valor Ganado

En este trabajo	En inglés	Pressman 5ª edición	Pressman 6ª edición
Valor Planeado (VPL)	Planned Value (PV) or Budgeted Cost of Work Scheduled (BCWS)	Costo Presupuestado del Trabajo Planeado (CPTP)	Costo Presupuestado para Trabajo Calendarizado (CPTC)
Costo Real (CR)	Actual Cost (AC) or Actual Cost of Work Performed (ACWP)	Costo Real del Trabajo Realizado (CRTR)	Costo Real del Trabajo Realizado (CRTR)
Valor Ganado o devengado (VG)	Earned Value (EV) or Budgeted Cost of Work Performed (BCWP)	Costo Presupuestado del Trabajo Desarrollado (CPTD)	Costo Presupuestado del Trabajo Realizado (CPTR)

# Valor Ganado (8/10)

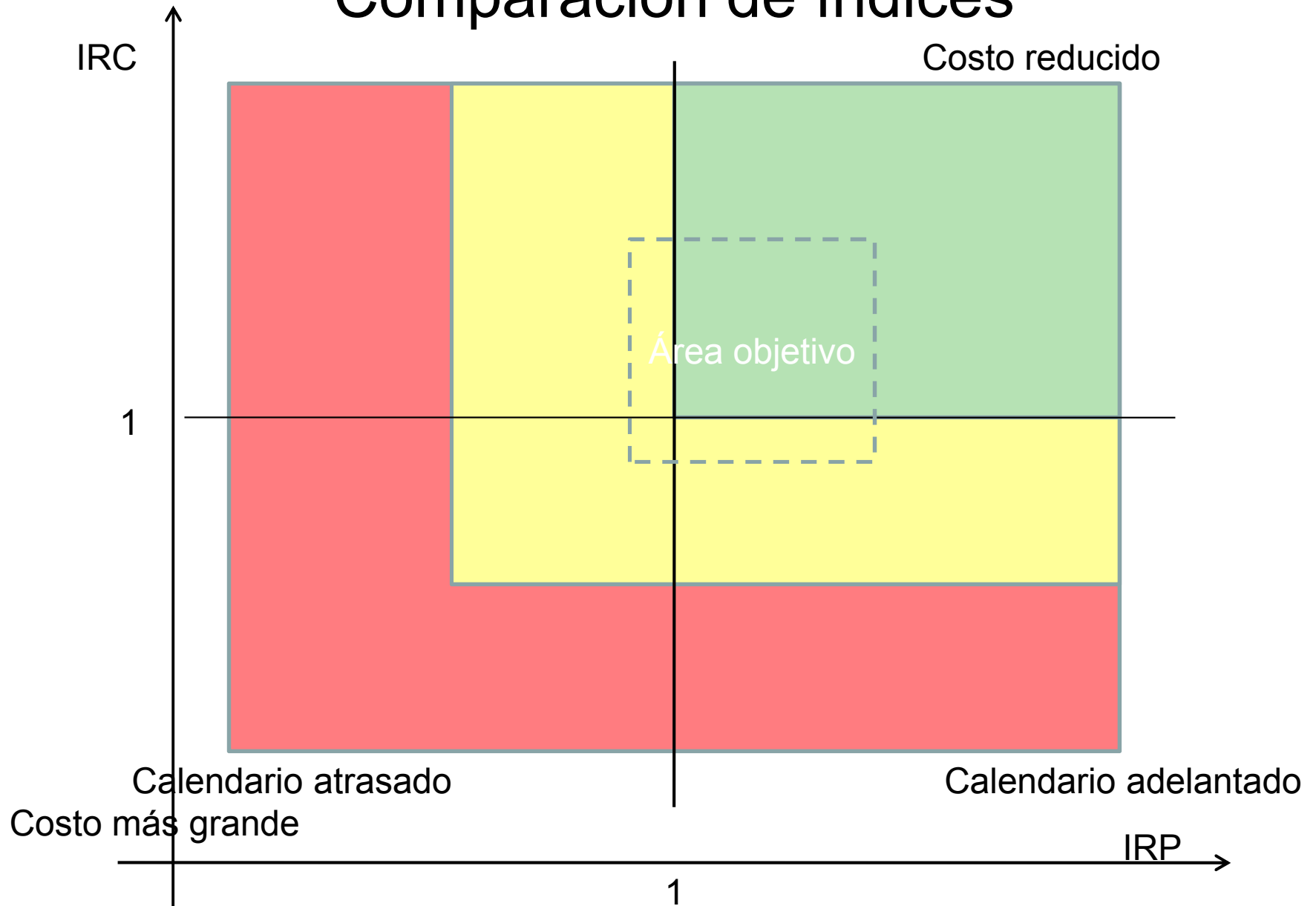
## Nomenclatura de Valor Ganado

En este trabajo	En inglés	Pressman 5ª edición	Pressman 6ª edición
Varianza en la Planificación (VP)	Schedule Variance (SV)	Varianza de la Planificación (VP)	Varianza en la Calendarización (VC)
Varianza en el Costo (VC)	Cost Variance (CV)	Varianza en el Coste (VC)	Varianza del Costo (Vco)
Índice de rendimiento en la Planeación (IRP)	Schedule Performance Index (SPI)	Índice de Desarrollo de Planificación (IDP)	Índice de Desempeño de la Calendarización (IDC)
Índice de rendimiento del Costo (IRC)	Cost Performance Index (CPI)	Índice de Desarrollo de Coste (IDC)	Índice de Desempeño del Costo (IDCo)

# Propuesta del modelo CMMI sobre Valor Ganado

- Objetivo:
  - Monitorear la realización de un proyecto a partir del plan original de ejecución, tomado la línea base, y aplicando la técnica administrativa de valor ganado.
  - La aplicación dará como resultado los indicadores: índice de rendimiento del costo, CPI (Cost Performance Index) e índice de rendimiento de planeación, SPI (Schedule performance index)
- Preguntas que resuelve:
  - ¿Están los indicadores SPI y CPI en el área objetivo?
  - ¿Cuál es la desviación en cuanto a costo?
  - ¿Cuál es la desviación en referencia al cronograma?

# Comparación de índices



# Comparación de índices

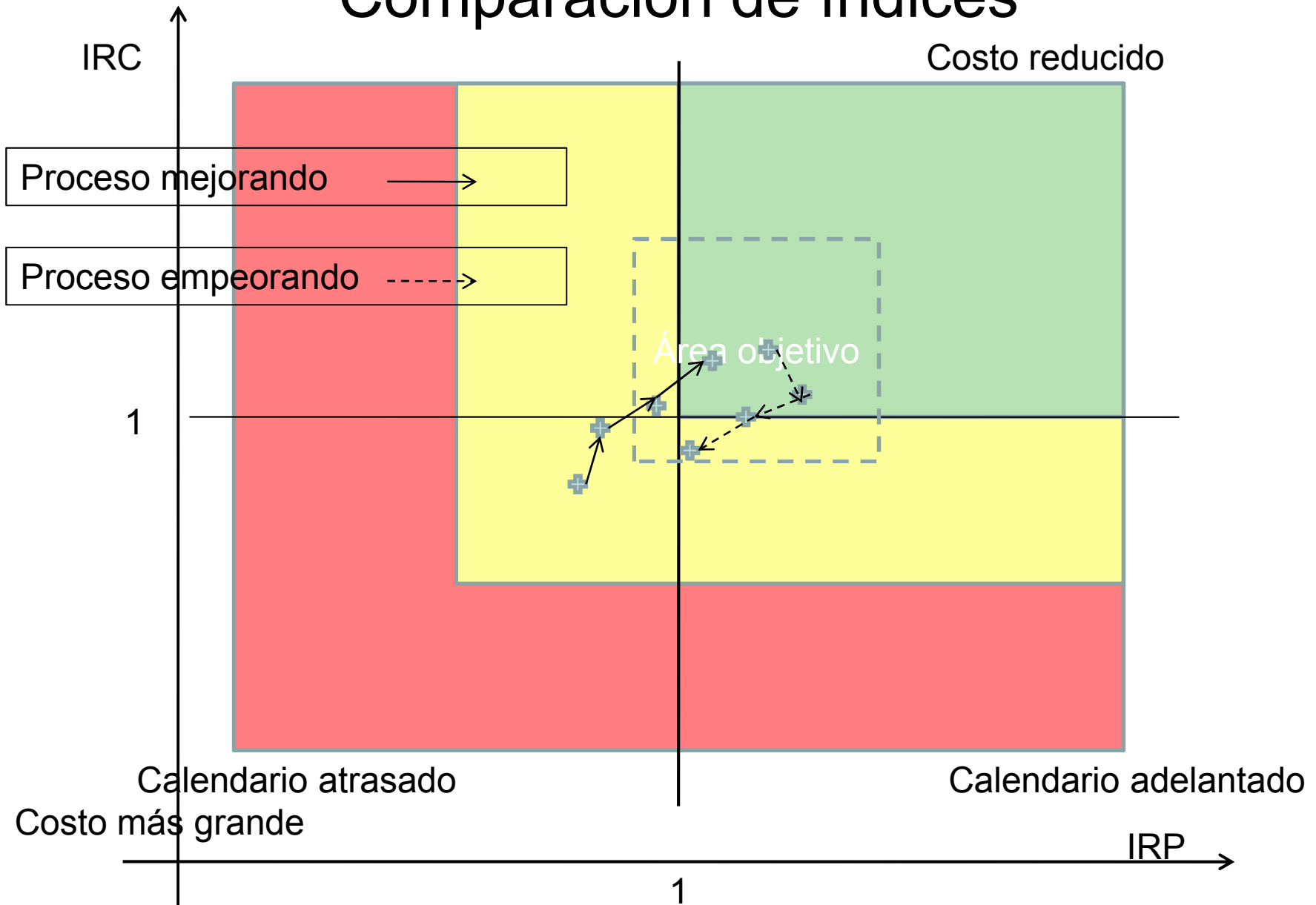
- El área objetivo se determina por cada empresa, indica tanta tolerancia que se considera normal en los índices; note que, usualmente, es menor en la parte negativa
- El área verde fuera del área objetivo, representa errores de planeación que no afectan por el momento, pero que deben corregirse a futuro, para ganar competitividad.
- El área amarilla representa zona riesgosa, pero no crítica.
- El área roja requiere acción inmediata, ya que puede dañar el proyecto de manera definitiva



# Comparación de índices

- El cuadro mostrado es estático.
- Es muy importante considerar tendencias, que pueden ir mejorando o empeorando.
- La tendencia sirve como aviso temprano de problemas futuros.
- En el cuadro siguiente se muestran:
  - un curso que va mejorando y
  - uno empeorando (línea punteada)

# Comparación de índices



# Propuesta del modelo CMMI

- Interpretación y análisis
  - Los puntos dentro del área objetivo indican que el proyecto esta controlado
  - Los puntos fuera de esta área deben considerarse como alertas, debe analizarse la situación y tomarse acciones correctivas
- Consideraciones
  - El equipo debe definir en consenso el área objetivo
  - valores de SPI y CPI para las áreas verde, amarilla y roja
  - las acciones a realizar cuando el proyecto se ubica fuera del área objetivo, ejemplo: revisiones técnicas
  - Definir acciones especiales cuando se está en el área roja

Presiones sobre el avance

# Seguimiento de la calendarización

## Presiones sobre calendario

- *Time-boxing. Técnica útil cuando*
  - se enfrentan severas presiones por la fecha límite, los gestores de proyecto experimentados pueden optar por la técnica de control
  - Se utiliza un modelo incremental o ágil con iteraciones reducidas
- Como se debe entregar todo o un incremento en fecha fija, en vez de posponer la entrega, se reduce entregable

# Seguimiento de la calendarización

## Presiones sobre calendario

- *Time-boxing* es una estrategia que:
  - Sugiere aceptar el límite de tiempo, pero entregar sólo lo que se pueda
  - Evita querer entregar todas las “grandes” funcionalidades en primera versión
  - Encajona en el tiempo disponible las tareas asociadas con cada incremento.
  - Se coloca una “caja” alrededor de cada tarea, significa que la calendarización para cada tarea se ajusta al trabajar hacia atrás desde la fecha de entrega para cada incremento.
  - Cuando una tarea se acerca al límite de su caja de tiempo, el trabajo se detiene, se evalúa y comienza la siguiente tarea.

# Seguimiento de la calendarización

## Presiones sobre calendario

- Su justificación es que cuando se llegue al límite de la caja de tiempo, es probable que se haya completado el 90 por ciento de la tarea, el restante 10 por ciento, aunque importante puede:
  1. demorarse hasta el siguiente incremento o
  2. completarse más tarde si se requiere.

# Seguimiento de la calendarización

## Presiones sobre calendario

- La reacción inicial al enfoque del encajonamiento de tiempo usualmente es negativa, se piensa que van a ir quedando pendientes
- Asegura terminar y entregar productos útiles a tiempo, más que completar detalles menores retrasados
- Si se ve el proyecto completo, formado por varias iteraciones, se verá que los faltantes se van cumpliendo después, pero no se dejan; ocurre una selección por prioridad o por minimización de riesgos
- Es un punto importante del Proceso Unificado (RUP)



# Seguimiento de la calendarización

## Presiones sobre calendario

- Ayuda a ser conciente del tiempo; a hacer lo que más importa
- A veces hacemos cosas que *urgen*, pero no importan y posponemos las más importantes.
- Ayuda a verificar cuánto tiempo requiere una tarea realmente

# CONTROL ESTADÍSTICO DE PROCESOS

# Variación de la gestión: control estadístico de procesos

- Tanto el proceso de desarrollo de software como el producto son influenciados por muchos parámetros.
  - la métrica elegida para un proyecto o producto no será la misma que para otro proyecto.
- *¿Cómo se puede decir si unos valores de métricas mejoradas que ocurren como consecuencia de actividades de mejora están de hecho teniendo un impacto cuantitativo real?*
- *¿Cómo saber si el proceso está empeorando?*

# Gráfico de control

- Desarrollada por Walter Shewart en 1920
- Técnica gráfica para determinar si los cambios y la variación en los datos de la métrica son significativos
  - permite que las personas interesadas en la mejora de procesos de software determinen si la dispersión y la localización de métricas de procesos son estables o inestables

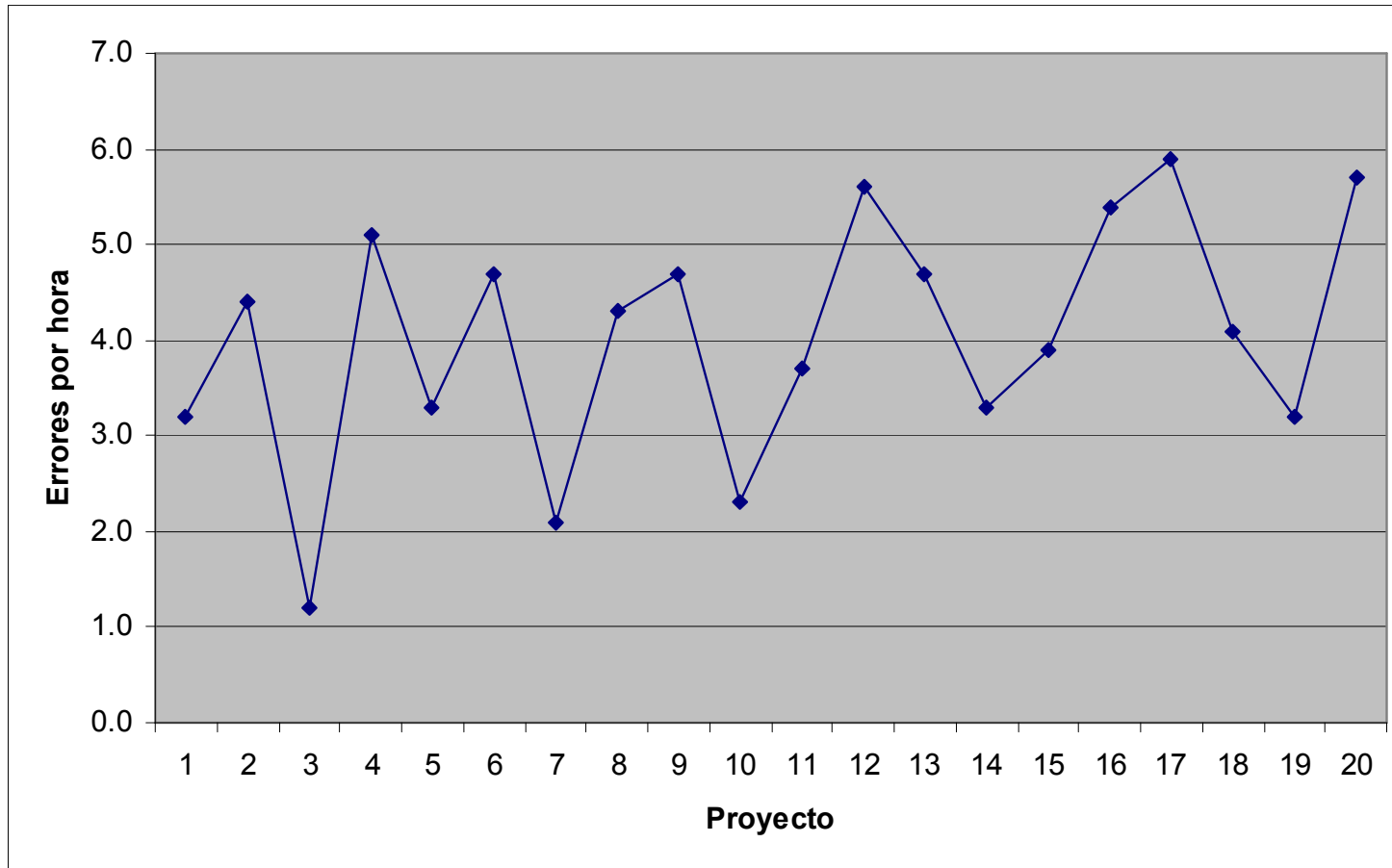
# Ejemplo de gráfico de control (1)

- Se considera una organización de software que registre en la métrica del proceso los errores descubiertos por hora de revisión,  $Er$ .
  - Durante los pasados 15 meses, la organización ha registrado el  $Er$  para 20 pequeños proyectos.
  - Suponga los siguientes datos obtenidos por proyecto revisado

Proy. #	Errores
1	3.2
2	4.4
3	1.2
4	5.1
5	3.3
6	4.7
7	2.1
8	4.3
9	4.7
10	2.3

Proy. #	Errores
11	3.7
12	5.6
13	4.7
14	3.3
15	3.9
16	5.4
17	5.9
18	4.1
19	3.2
20	5.7

# Ejemplo de gráfico de control (2)



Datos acumulados para 20 proyectos. Errores identificados por hora de revisión.

# Gráficos de control

- Existen dos tipos diferentes de gráficos de control que se usan en la evaluación de los datos métricos:
  1. El gráfico de control de rango móvil (Rm)
    - Determinar la estabilidad del proceso.
  2. El gráfico de control individual
    - Indica si un proceso está bajo control o fuera de control

# Gráfico de control de rango móvil (Rm)

- Procedimiento para desarrollar un gráfico de control de rango móvil (Rm) :
  1. Calcular los rangos móviles: el valor absoluto de las diferencias sucesivas entre cada pareja de puntos de datos.... Dibujar estos rangos móviles sobre el gráfico.
  2. Calcular la media de los rangos móviles... dibujando ésta (barra Rm)
    - Línea central del propio gráfico.
  3. Multiplicar la media por 3.268. Dibujar esta línea como el límite de control superior (LCS)
    - Esta línea supone tres veces el valor de la desviación estándar por encima de la media.
- El valor mínimo usualmente es cero, ya que no hay valores negativos



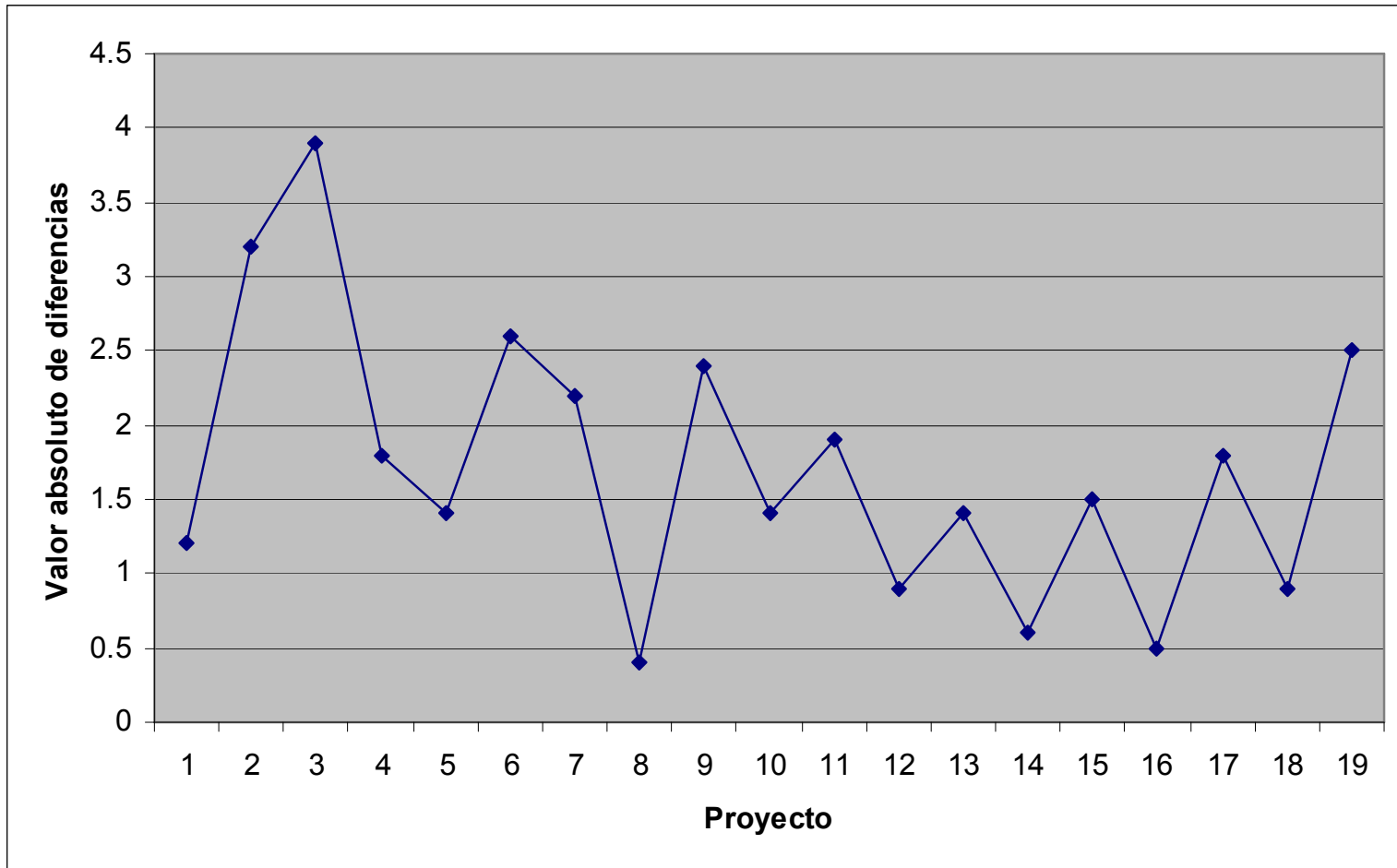
# Procedimiento para Grafo de Rango móvil

1. Calcular los rangos móviles: el valor absoluto de las diferencias sucesivas entre cada pareja de puntos de datos

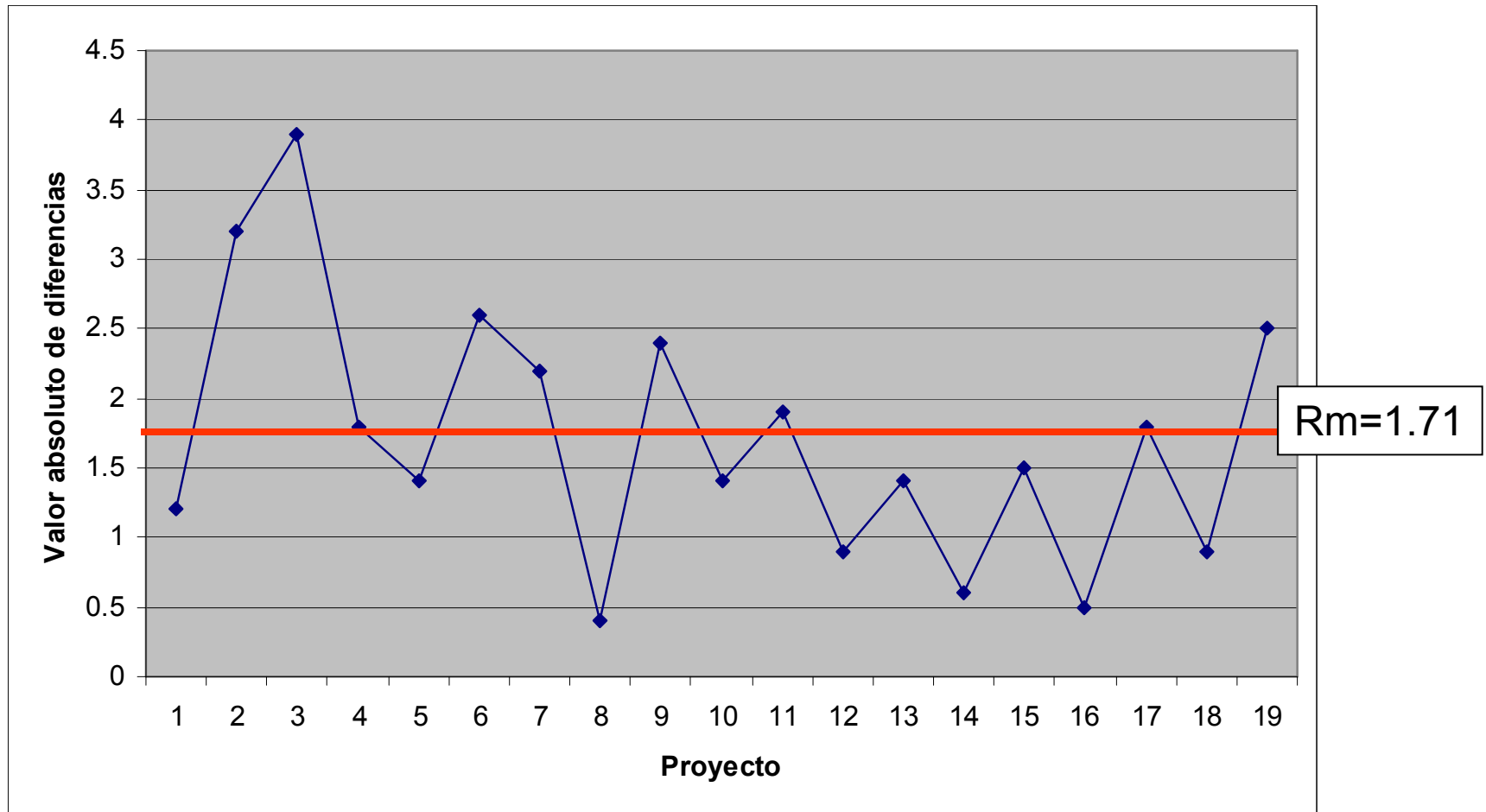
Pr	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Er	3. 2	4. 4	1. 2	5. 1	3. 3	4. 7	2. 1	4. 3	4. 7	2. 3	3. 7	5. 6	4. 7	3. 3	3. 9	5. 4	5. 9	4. 1	3. 2	5. 7
M o		1. 2	3. 2	3. 9	1. 8	1. 4	2. 6	2. 2	0. 4	2. 4	1. 4	1. 9	0. 9	1. 4	0. 6	1. 5	0. 5	1. 8	0. 9	2. 5

.... Dibujar estos rangos móviles sobre el gráfico.

# Gráfico de control de rango móvil (Rm)

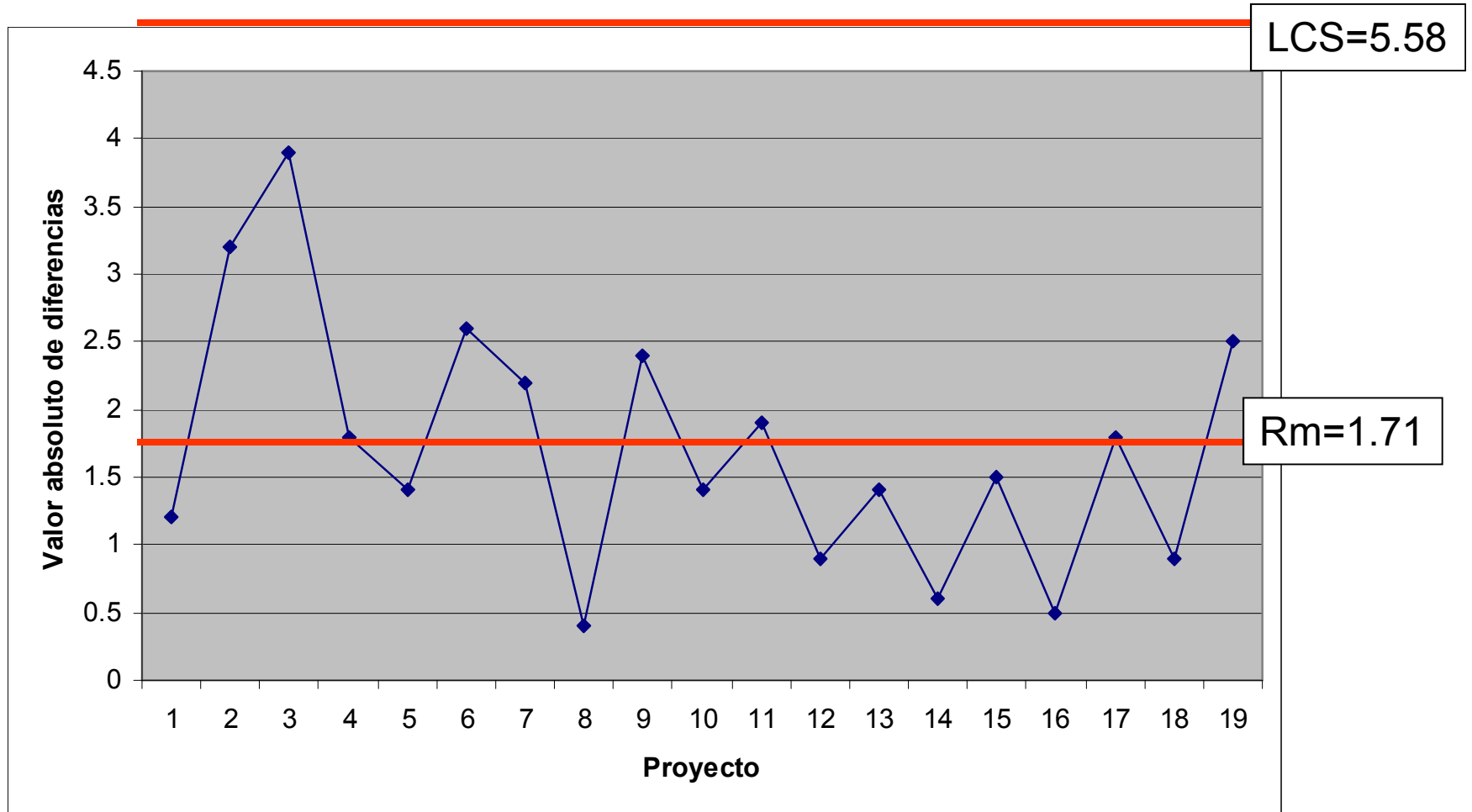


# Gráfico de control de rango móvil (Rm)



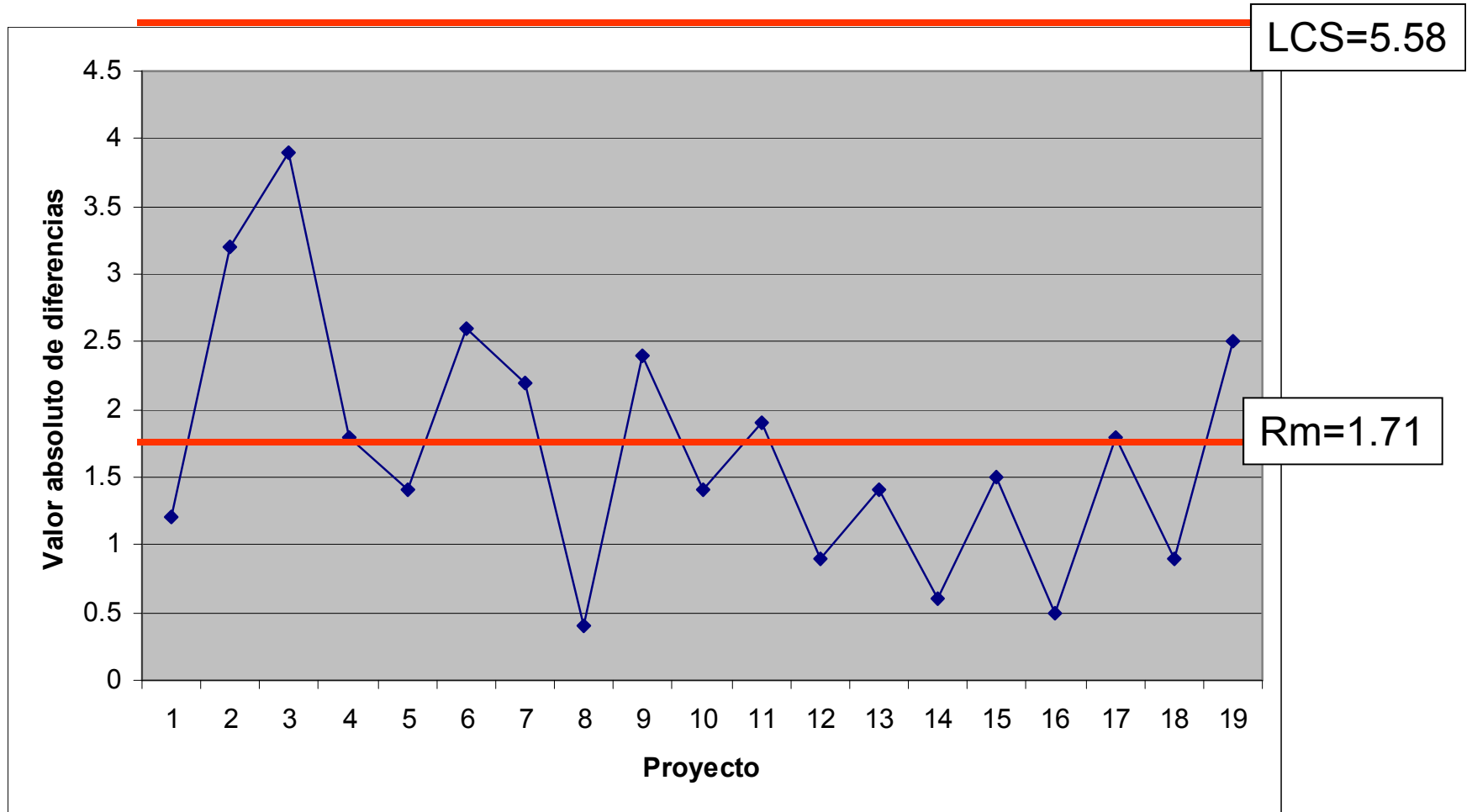
2. Calcular la media de los rangos móviles...  
dibujando ésta (barra  $R_m$ )
  - Línea central del propio gráfico.

# Gráfico de control de rango móvil (Rm)



3. Multiplicar la media por 3.268. Dibujar esta línea como el límite de control superior (LCS)
  - Esta línea supone tres veces el valor de la desviación estándar por encima de la media.

# Gráfico de control de rango móvil (Rm)



Para determinar si la dispersión de las métricas del proceso es estable puede preguntarse una cuestión muy sencilla: ¿Están los valores de rango móvil dentro del LCS? En este ejemplo la respuesta es sí,  $\therefore$  la dispersión de la métrica es estable

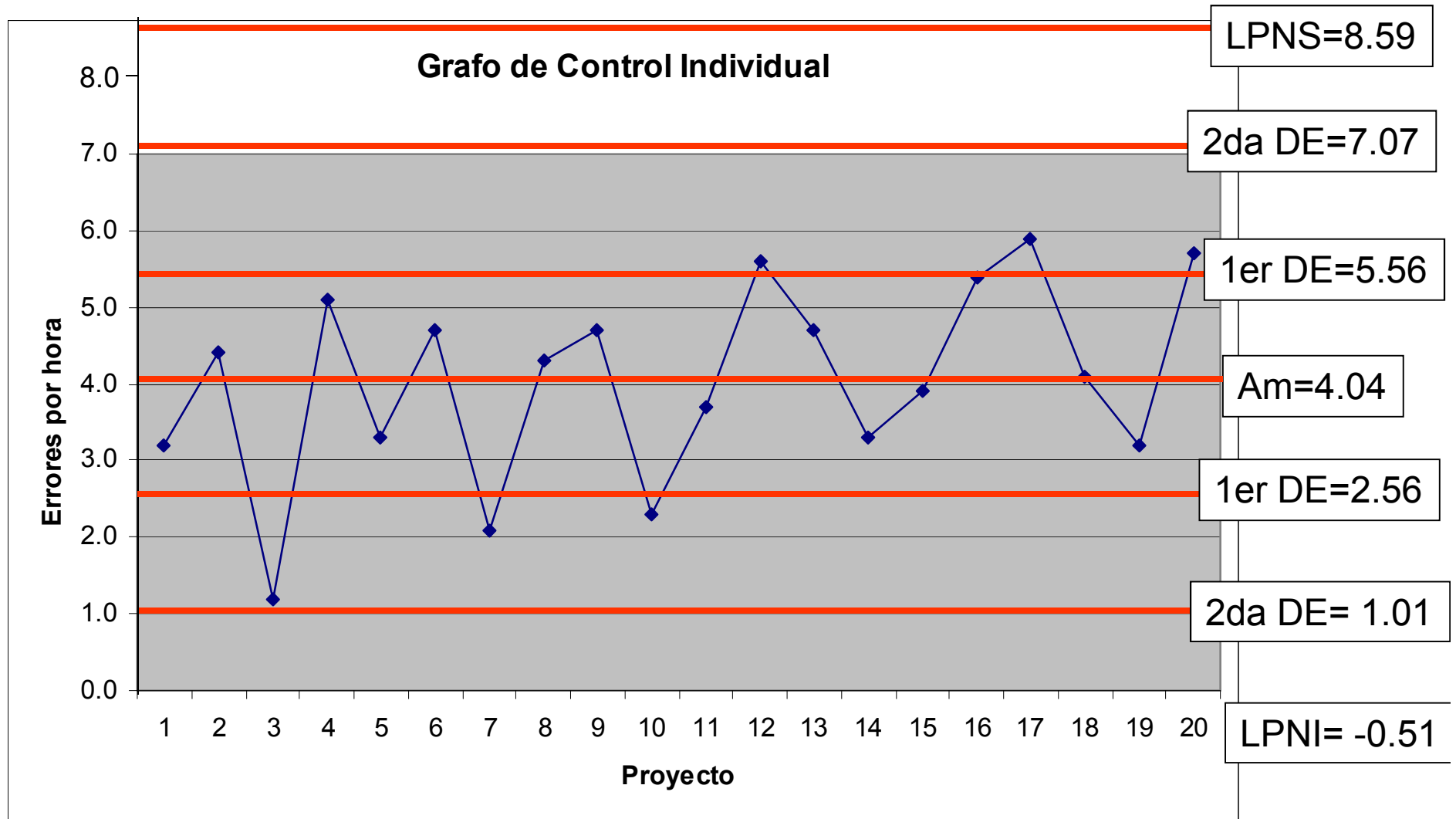
# Gráfico de control individual (1/4)

- Se desarrolla de la siguiente manera:
  1. Dibujar los valores de la métrica individual
  2. Calcular el valor promedio  $A_m$ , para los valores de la métrica.
  3. Multiplicar la media de los valores  $R_m$  (la barra  $R_m$ ) por 2.660 y añadir el valor de  $A_m$  calculado en el paso 2, se denomina límite de proceso natural superior (LPNS). Dibujar el LPNS.

## Gráfico de control individual (2/4)

4. Multiplicar la media de los valores  $R_m$  (barra  $R_m$ ) por 2.660 y restar este valor del  $A_m$  calculado en el paso 2, límite de proceso natural inferior (LPNI). Dibujar el LPNI.
  - Si el LPNI es menor que 0.0, no necesita ser dibujado a menos que la métrica que está siendo evaluada tome valores que sean menores que 0.0.
5. Calcular la desviación estándar según la fórmula  $(LPNS - A_m)/3$ . Dibujar las líneas de la desviación estándar una y dos por encima y por debajo de  $A_m$ .
  - Si cualquiera de las líneas de desviación estándar es menor que 0.0, no necesita ser dibujada a menos que la métrica que está siendo evaluada tome valores que sean menores que 0.0

# Gráfico de control individual (3/4)



Se omite LPNI por ser negativo y no existe número negativo de defectos



# Gráfico de control individual (4/4)

- Zultner revisa cuatro criterios, denominados *reglas de zona*, que pueden usarse para evaluar si los cambios representados por la métrica indican que un proceso está bajo control o fuera de control.
  - Si cualquiera de las siguientes condiciones es verdadera, los datos de la métrica indican un proceso que está fuera de control:
    1. Un valor de la métrica individual aparece fuera del LPNS.
    2. Dos de cada tres valores de métricas sucesivas aparecen más de dos desviaciones estándar fuera del valor  $A_m$
    3. Cuatro de cada cinco valores de métricas sucesivas aparecen alejados más de una desviación estándar del valor  $A_m$
    4. Ocho valores consecutivos de métrica aparecen todos situados a un lado del valor  $A_m$ .

# Uso del grafo y las reglas

- Las reglas anteriores sirven para alertar al administrador cuando un proceso se sale de control.
- Como en los indicadores de valor ganado, la observación periódica marca tendencias que avisan antes de llegar a aplicar una de las reglas mencionadas
- Si se observa una tendencia a aumentar valores alejados de la media o a caer siempre del mismo lado, debe revisarse el proceso
- Al decir revisar el proceso, no significa que sea un problema, puede indicar una mejora, por introducción de nuevos métodos o herramientas, capacitación, mejora de condiciones, etc.

# Seguimiento: Defectos y su reducción

Garantía de Calidad Estadística

# Para qué de la prevención

- Se busca una mejora continua del proceso de desarrollo
- Los defectos indican puntos débiles del proceso
- El análisis ayuda a comprender y corregir esos puntos débiles

# Pasos

1. Analizar defectos o errores para hallar causas básicas
2. Sugerir acciones preventivas para eliminar causas básicas
3. Implementar las acciones preventivas

# Elementos importantes

- Reuniones de análisis:
  - dos horas, al final de etapa, se buscan causas y remoción, se analizan tendencias, se registra. Sin administradores o autoridades.
- Equipo de acción
  - Se revisan y priorizan los problemas hallados
  - Se implementan acciones sugeridas

# Análisis

- Clasificación de defectos (elegir con cuidado). Por ejemplo Clasificación Ortogonal.
- Preparación de diagrama de Pareto
- Búsqueda de causas (árbol de falla o diagrama de espina de pescado)

# Clasificación tradicional

- Taxonomía de Beizer

Código	Tipo de problema
1XXX	Functional bugs: requirements and features
12XX	Logic
3XXX	Structural bugs

- Taxonomía de Kaner

USER INTERFACE ERRORS

Functionality

Excesive functionality

Doesn't do what the user expect

Communication

Wrong, misleading, or confusing information



# Clasificación ortogonal

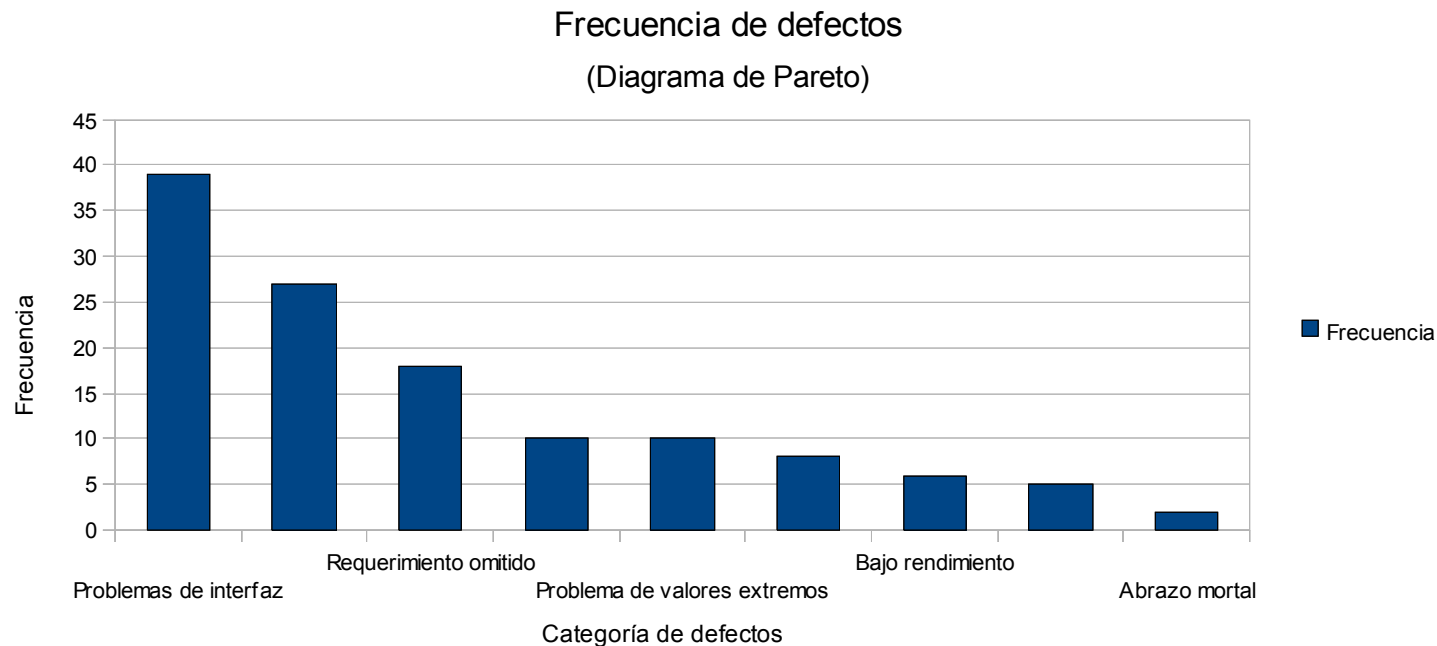
- Forma tradicional genera ambigüedades e inconsistencias
- Chillarege (1992) propuso método con ejes ortogonales en dos momentos:
  - Apertura: ocurrencia del problema (“falla”)
  - Cierre: identificación de causa (“defecto”) y cambio necesario para corregirlo

# Clasificación ortogonal

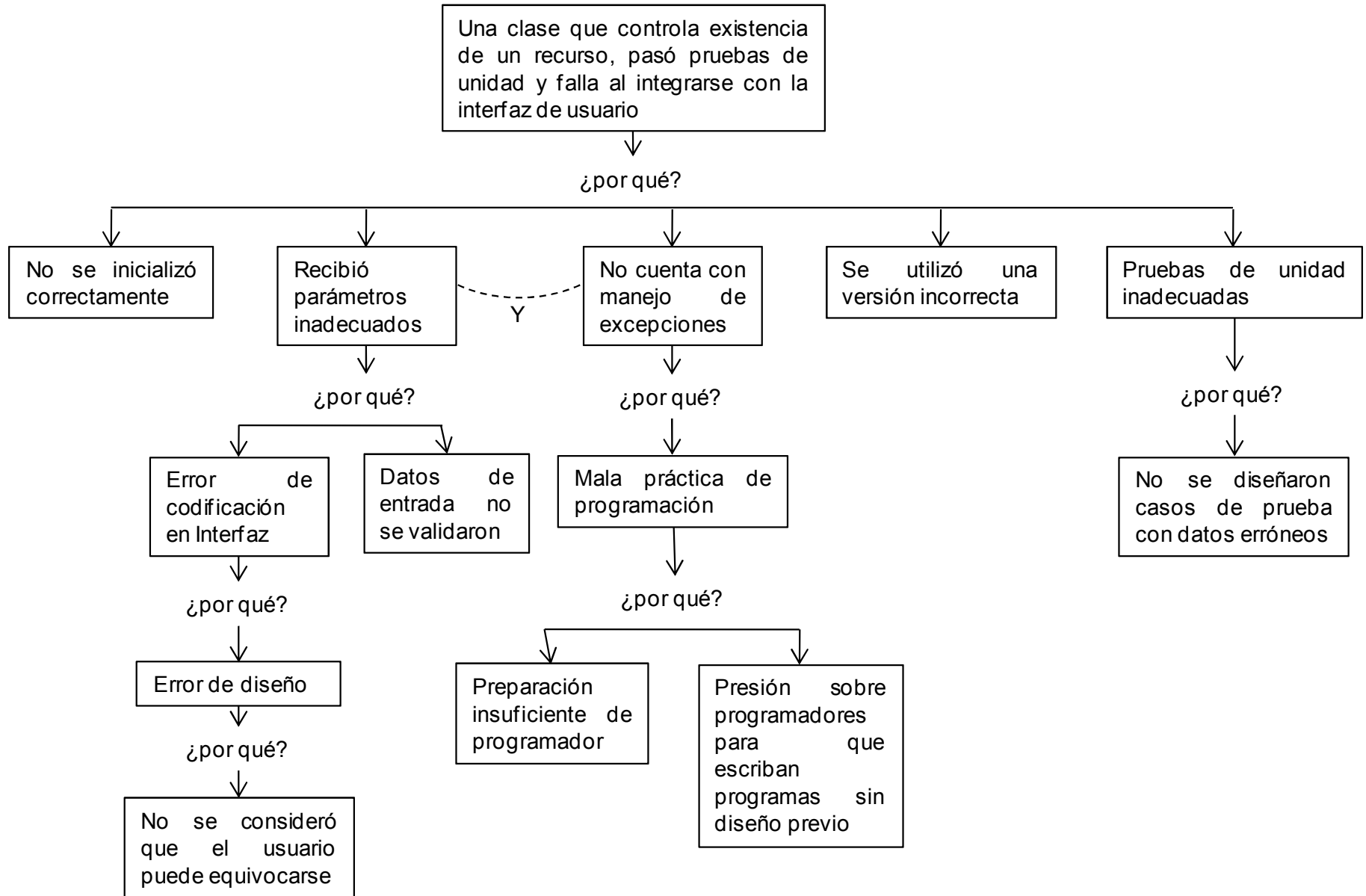
- **Apertura**
  - Etapa en que ocurre
  - Disparador (situación que lo hace aparecer)
  - Impacto
- **Cierre**
  - Sujeto del cambio (qué se cambia)
  - Tipo (asignación/inicialización, algoritmo, etc.)
  - Calificador (faltante, incorrecto, extraño)
  - Edad (código nuevo, reutilizado, etc.)
  - Fuente (origen: local, adquirido, etc.)

# Diagrama de Pareto

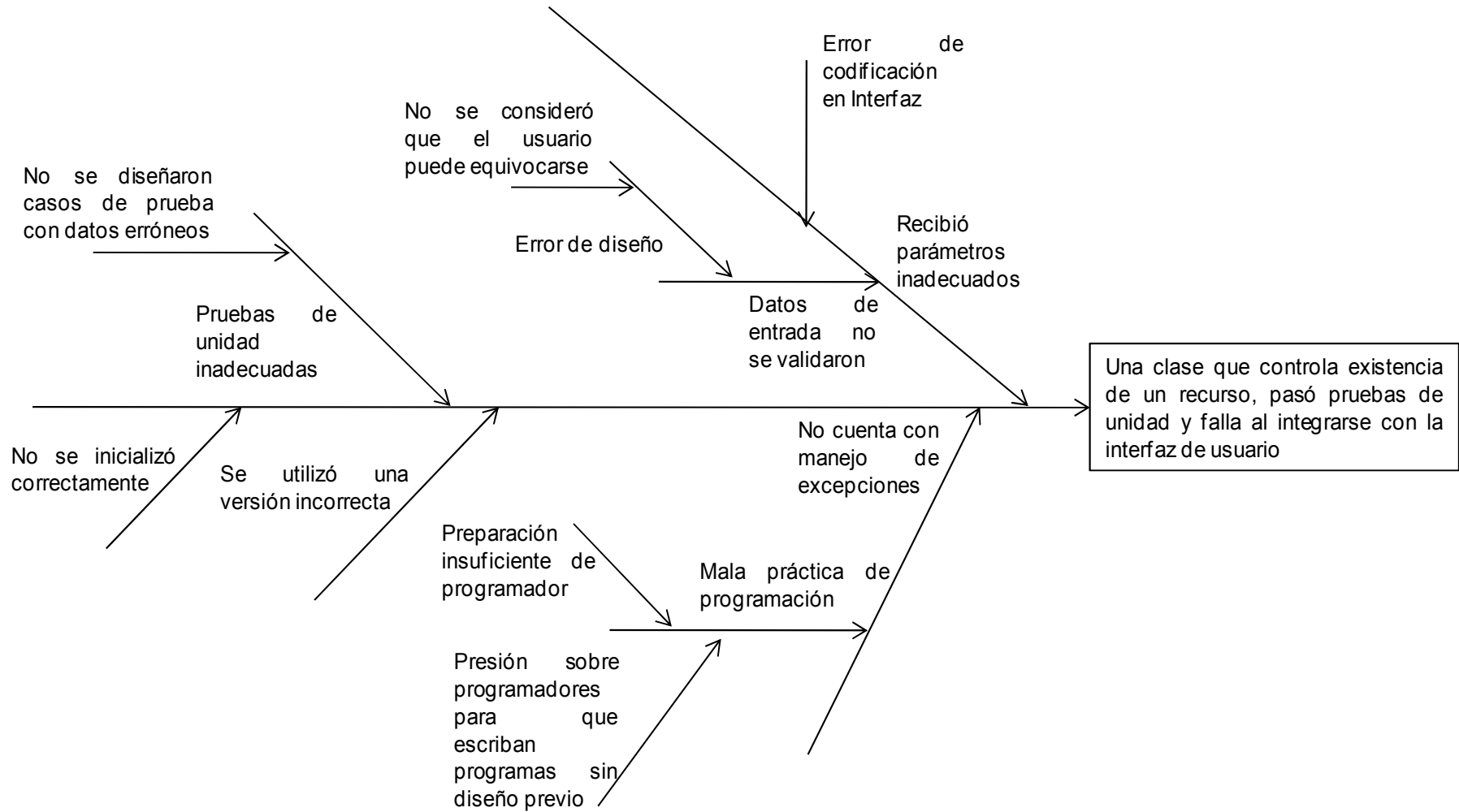
- Representa frecuencia de problemas en forma decreciente
- Interpretación ligera: 80 % de problemas por el 20 % de causas (las más graves)



# Análisis de causas: árbol por qué



# Análisis de causas: espina de pescado



# Acciones preventivas

- Problema: “No se consideró que el usuario puede equivocarse”
- Acción: Incluir siempre validación de datos de entrada

# Acciones preventivas

- Problema: mala práctica de programación por falta de capacitación y presión sobre programadores
- Acciones: capacitar adecuadamente, insistir en diseño y dar tiempo (se requiere compromiso de administración)

# Se concreta en

- Reglas de la empresa (sólo, no muy práctico a veces)
  - Asegúrese de validar cada entrada de datos del usuario
  - Asegúrese de que el código corresponda con el diseño
- Listas de verificación para Revisiones
  - ¿se valida cada entrada de usuario?
  - ¿existe diseño para el módulo programado?
  - ¿el módulo corresponde a su diseño?



# Métricas asociadas más importantes

- Número de defectos
  - Abiertos, nuevos, corregidos, cerrados
- Densidad de defectos: defectos/tamaño
  - Generalmente defectos por millar de líneas de código

# Métricas asociadas menos importantes

- Tiempo para corregir defecto
- Defectos eliminados en desarrollo (% del total hallado en la vida del producto)
- Defectos pendientes por unidad de tiempo

# Un ejemplo genérico (1/5)

- Para ilustrar la aplicación de los métodos estadísticos en el trabajo de ingeniería del software, supóngase que:
  - Una organización de ingeniería del software recopila información acerca de defectos durante un año. Algunos de los defectos se descubren cuando el software está en desarrollo; otros, después de que se ha liberado entre sus usuarios finales. Aunque se descubren cientos de diferentes defectos, todos tienen una (o más) de las causas siguientes:

# Un ejemplo genérico (2/5)

TIPO DE DEFECTO	CLAVE
Especificaciones incompletas o erróneas	EIE
Mala interpretación de la comunicación del cliente	MCC
Desviación intencional de las especificaciones	DIE
Violación de los estándares de programación	VEP
Errores en la representación de los datos	ERD
Interfaz de componente inconsistente	ICI
Error en la lógica del diseño	ELD
Prueba incompleta o errónea	PIE
Documentación imprecisa o incompleta	DII
Error en la traducción del diseño al lenguaje de programación	TLP
Interfaz hombre-computadora ambigua o inconsistente	IHC
Misceláneo	MIS

# Un ejemplo genérico (3/5)

## Tabla de recolección de datos sobre defectos

	<b>Total</b>		<b>Serios</b>		<b>Moderados</b>		<b>Menores</b>	
<b>Error</b>	<b>Núm</b>	<b>%</b>	<b>Núm</b>	<b>%</b>	<b>Núm</b>	<b>%</b>	<b>Núm</b>	<b>%</b>
EIE	205	22%	34	27%	68	18%	103	24%
MCC	156	17%	12	9%	68	18%	76	17%
DIE	48	5%	1	1%	24	6%	23	5%
VEP	25	3%	0	0%	15	4%	10	2%
ERD	130	14%	26	20%	68	18%	36	8%
ICI	58	6%	9	7%	18	5%	31	7%
ELD	45	5%	14	11%	12	3%	19	4%
PIE	95	10%	12	9%	35	9%	48	11%
DII	36	4%	2	2%	20	5%	14	3%
TLP	60	6%	15	12%	19	5%	26	6%
IHC	28	3%	3	2%	17	4%	8	2%
MIS	56	6%	0	0%	15	4%	41	9%
<b>Total</b>	<b>942</b>	<b>100%</b>	<b>128</b>	<b>100%</b>	<b>379</b>	<b>100%</b>	<b>435</b>	<b>100%</b>

- EIE, MCC y ERD son las causas vitales que explican el 53% de todos los defectos
- EIE, ERD, TLP y ELD son las causa de los defectos más serios, en ellos se debe aplicar la acción correctiva

# Un ejemplo genérico (4)

- La acción correctiva se enfoca principalmente en las vitales.
  - Conforme éstas se corrigen, nuevas candidatas ocupan la parte superior de la clasificación.
- La aplicación del principio de Pareto se puede resumir en una sola oración:
  - *Emplee su tiempo enfocándose en las cosas que realmente importan, ¡pero primero asegúrese de entender qué es lo que realmente importa!*

# Seis sigma para ingeniería del software (1)

- Seis sigma es la estrategia más ampliamente empleada en la actualidad para el aseguramiento de la calidad estadístico en la industria.
- Popularizada por Motorola en el decenio de 1980
- Metodología rigurosa y disciplinada que utiliza análisis de datos y estadístico para medir y mejorar el desempeño operativo de una compañía al identificar y eliminar los 'defectos' en la fabricación y los procesos relacionados con el servicio

# Seis sigma para ingeniería del software (2)

- La metodología seis sigma define tres pasos centrales:
  1. Definir los requisitos del cliente, entregables y metas del proyecto por medio de métodos bien definidos de comunicación con el cliente.
  2. Medir el proceso existente y su salida para determinar el desempeño de calidad actual (recopilación de métricas de defecto).
  3. Analizar las métricas de defecto y determinar las causa poco vitales.



# Seis sigma para ingeniería del software (3)

- Si un proceso de software existente está en marcha, pero se requiere mejoría, seis sigma sugiere dos pasos adicionales:
  4. Mejorar el proceso eliminando las causas originales de los defectos.
  5. Controlar el proceso para garantizar que el trabajo futuro no vuelva a introducir las causas de defectos.
- Estos pasos centrales y adicionales a veces se conocen como el método DMAMC (definir, medir, analizar, mejorar y controlar).

# Seis sigma para ingeniería del software (4/5)

- Si una organización está desarrollando un proceso de software, los pasos centrales se aumentan de la siguiente manera:
  - Diseñar el proceso para:
    1. evitar las causas originales de los defectos y
    2. satisfacer los requisitos del cliente.
  - Verificar que el modelo de proceso, de hecho, evitará los defectos y satisfará los requisitos del cliente.
- A esta variación a veces se le llama método DMADV (definir, medir, analizar diseñar y verificar)