

# *Revisiones del Software*

## Introducción

# Bibliografía

---

- IEEE Std 1028-1997
  - Standard for Software Reviews
- Página: recursos para revisiones
  - [http://www.processimpact.com/pr\\_goodies.shtml](http://www.processimpact.com/pr_goodies.shtml)  
(por Karl E. Wiegiers)

## Ventajas de las revisiones de SW

---

- No requiere de código ejecutable, por lo que puede ser realizada desde el inicio
  - Por lo tanto, es menos costosa
- Se encuentran varios defectos a la vez
- Encuentra hasta un 85% de los defectos (vs. 50% que encuentra las pruebas)
- Se localiza la posición exacta del defecto
- Refuerza el uso de estándares
- Mejora la capacitación

## **Desventajas de las revisiones de SW**

---

- Requiere del tiempo de los expertos
- No se pueden verificar características no-funcionales (ej. rendimiento)
- Validan cumplimiento de lo que se especificó, en vez de lo que realmente desea el cliente
- Es difícil de implementar
  - Es vista como “improductiva” por los ingenieros

# **Pero ¿Realmente son improductivas las revisiones?**

---

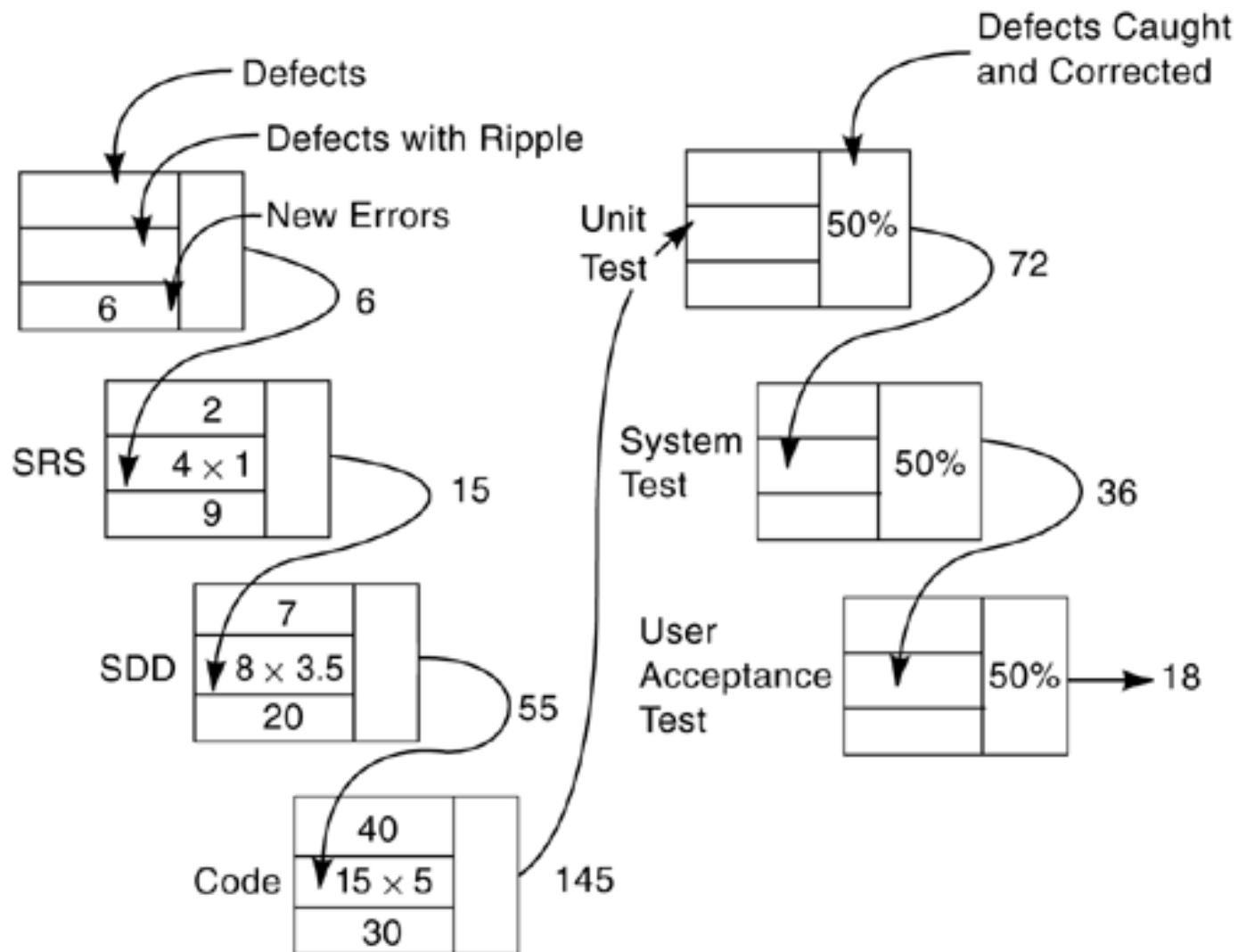
- Supongamos un sistema al que se le encuentran 200 defectos en las pruebas internas
  - De acuerdo a estadísticas mundiales, un 40% de esos defectos son por requerimientos defectuosos
  - Eso significa que 80 de ellos se cometieron en el análisis de requerimientos
- Supongamos que si se hubieran detectado en el análisis cada uno hubiera tomado 15 minutos arreglarlo
  - Según estadísticas mundiales arreglar un defecto de requerimientos en las pruebas internas cuesta 40 veces más que arreglarlo en el análisis
  - Entonces cada uno se arreglará en 10 horas, lo que da un total de 800 horas hombre arreglarlos todos

## **Pero ¿Realmente son improductivas las revisiones?**

---

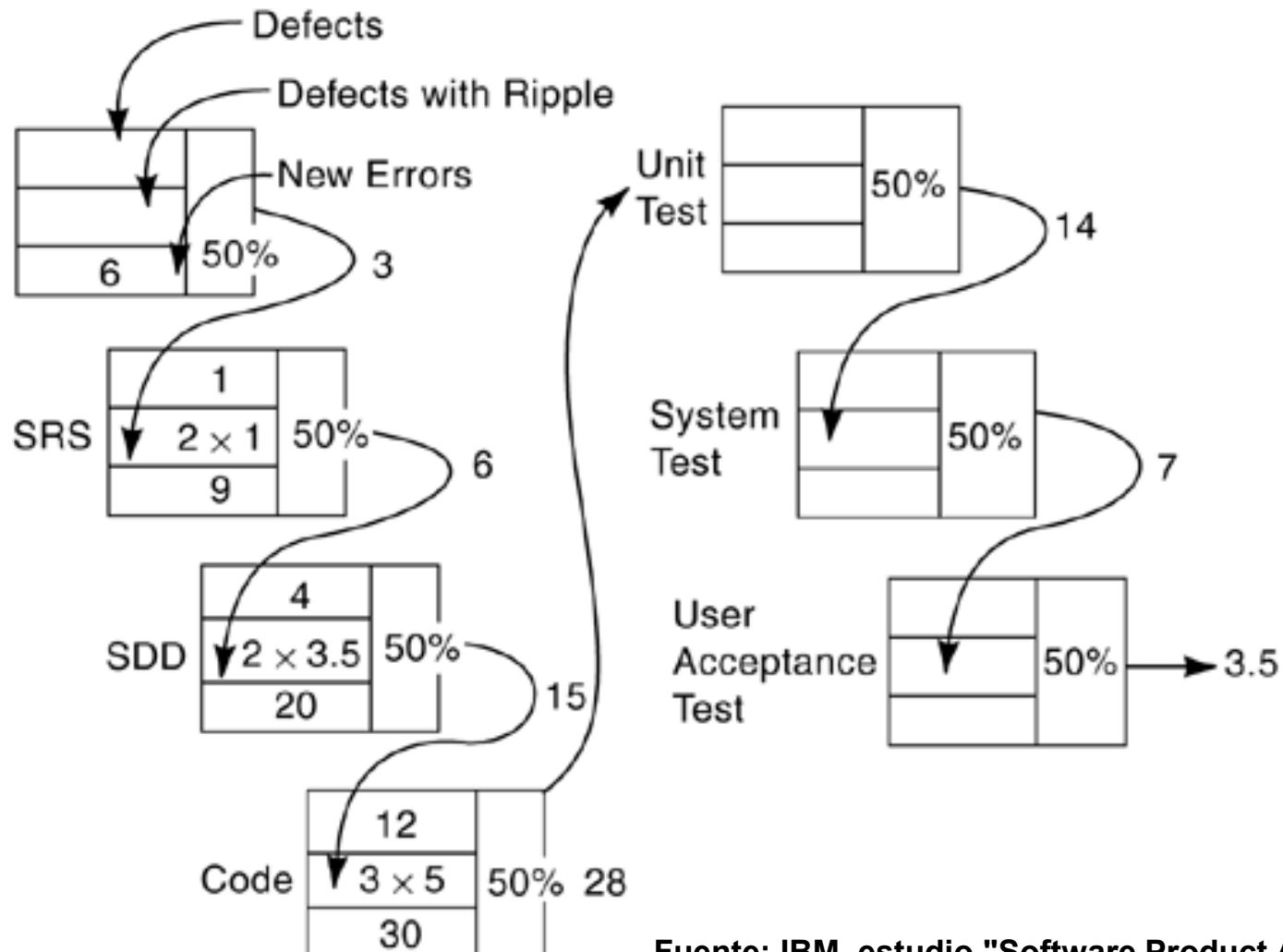
- Supongamos que se hubiera inspeccionado el documento de análisis
  - Según estadísticas mundiales es fácil detectar 70% de los defectos con una inspección (56 defectos)
- Supongamos que un equipo de 5 personas hubiera trabajado por 2 días en la revisión (80 HH)
- En total se hubieran dedicado 334 HH
  - Revisión = 80 HH
  - Corrección defectos en análisis = 14 HH ( $56 * 15 \text{ min}$ )
  - Corrección defectos en pruebas = 240 HH ( $24 * 10 \text{ hr}$ )
- Un ahorro de 466 HH ( $800 - 334 \text{ HH}$ )

# Sin revisiones: más errores en cascada



Fuente: IBM, estudio "Software Product Assurance."

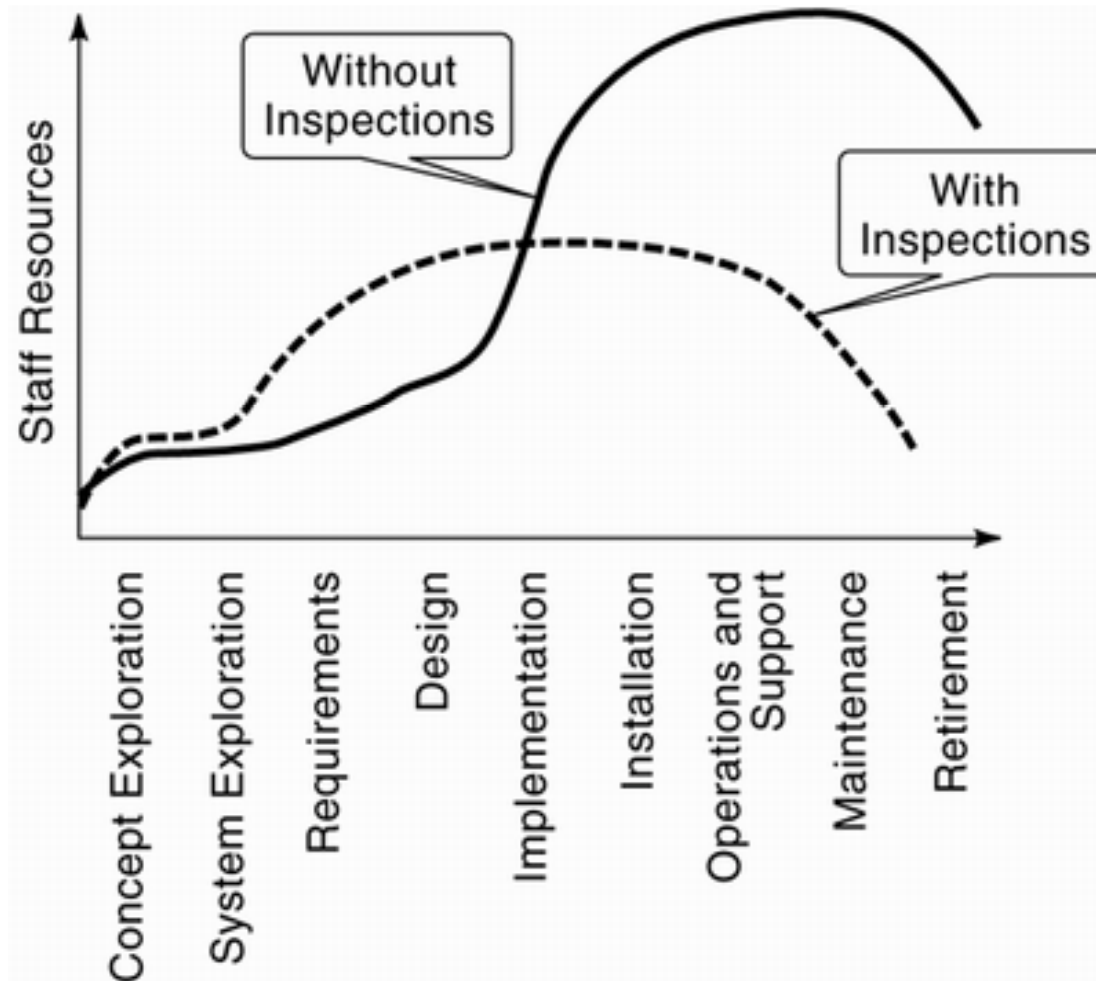
# Con revisiones: menos errores en cascada



Fuente: IBM, estudio "Software Product Assurance."



# Menos esfuerzo con revisiones



Fuente: Robert T. Futrell, "Quality Software Project Management"

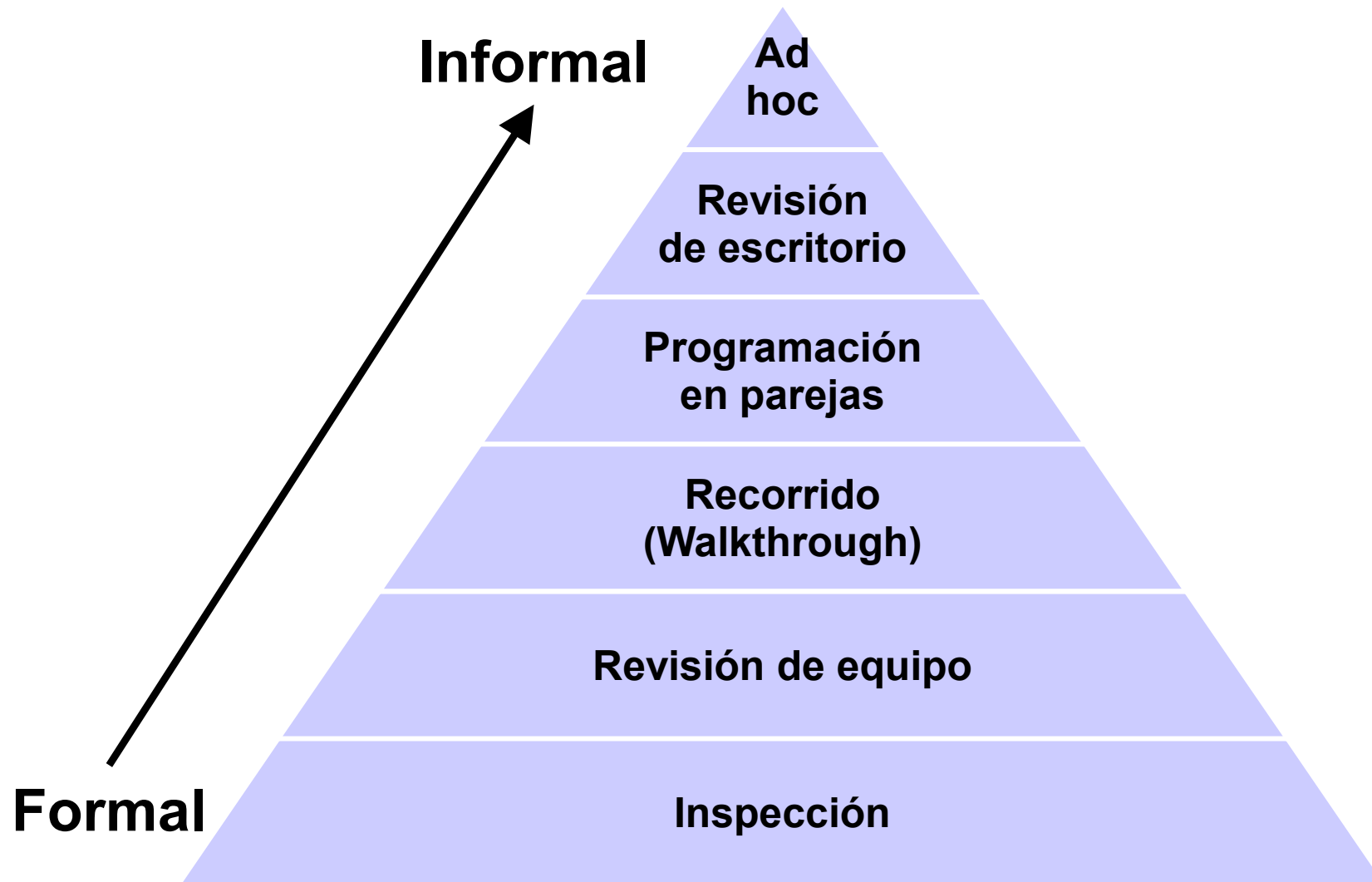
# Revisiones informales y formales

---

- Informales
  - No hay proceso definido
  - No existen roles
  - Usualmente no planeadas
- Formales
  - Objetivos definidos
  - Proceso documentado
  - Roles definidos y personas entrenados en ellos
  - Check-lists, reglas y métodos para encontrar defectos
  - Reporte del resultado
  - Recolección de datos para el control del proceso

Fuente: Karl Wiegers

# Tipos de revisiones de SW



Fuente: Karl Wieggers

# Productos que pueden ser revisados

1. Proposal
2. Contract
3. Schedule
4. Budget
5. Software project management plan (SPMP)
6. Feasibility statement
7. Software quality assurance plan (SQAP)
8. Software requirements specification (SRS)
9. Software configuration management plan (SCMP)
10. Project test plan
11. Logical model—DCD, DFD, ERD, class model, object model, PSPEC, CSPEC, AFD
12. Activity diagram, use case
13. Data dictionary
14. Traceability matrix
15. Software design document
16. Structure chart
17. Chapin (Nassi-Schneiderman) chart
18. State transition diagram, use case scenario, interaction diagram
19. Pseudocode, decision table, decision tree
20. Integration test plan
21. Conversion plan
22. System test plan
23. Software baseline
24. Acquisition plan
25. Transition plan
26. User's guide/manual
27. Operating documentation
28. Test report
29. Training plan
30. Preacceptance checklist
31. Installation plan
32. Acceptance test plan
33. Operational system
34. Acceptance test report
35. Maintenance plan

Fuente: Robert T. Futrell, "Quality Software Project Management"

# *Inspección vs. Recorrido (walkthrough)*

Tomado de:

[http://www.processimpact.com/reviews\\_book/peer\\_review\\_process.doc](http://www.processimpact.com/reviews_book/peer_review_process.doc)

# Una definición de inspección

---

- Mecanismo formal por medio del cuál
  - un grupo de personas
  - ajenas a un producto de trabajo
  - ayuda a detectar defectos en él
- No son para:
  - Revisar el progreso de un producto
  - Evaluar al autor del producto

# Características de la inspección

---

- Se involucra el mayor número de personas ajenas al autor (mínimo 2)
  - Deben ser personas técnicamente competentes
  - No se permite la presencia de los jefes del autor
- Existe un compromiso formal entre los involucrados
- Está planeado
- Existe un registro y una corrección de defectos

# Roles en la inspección

---

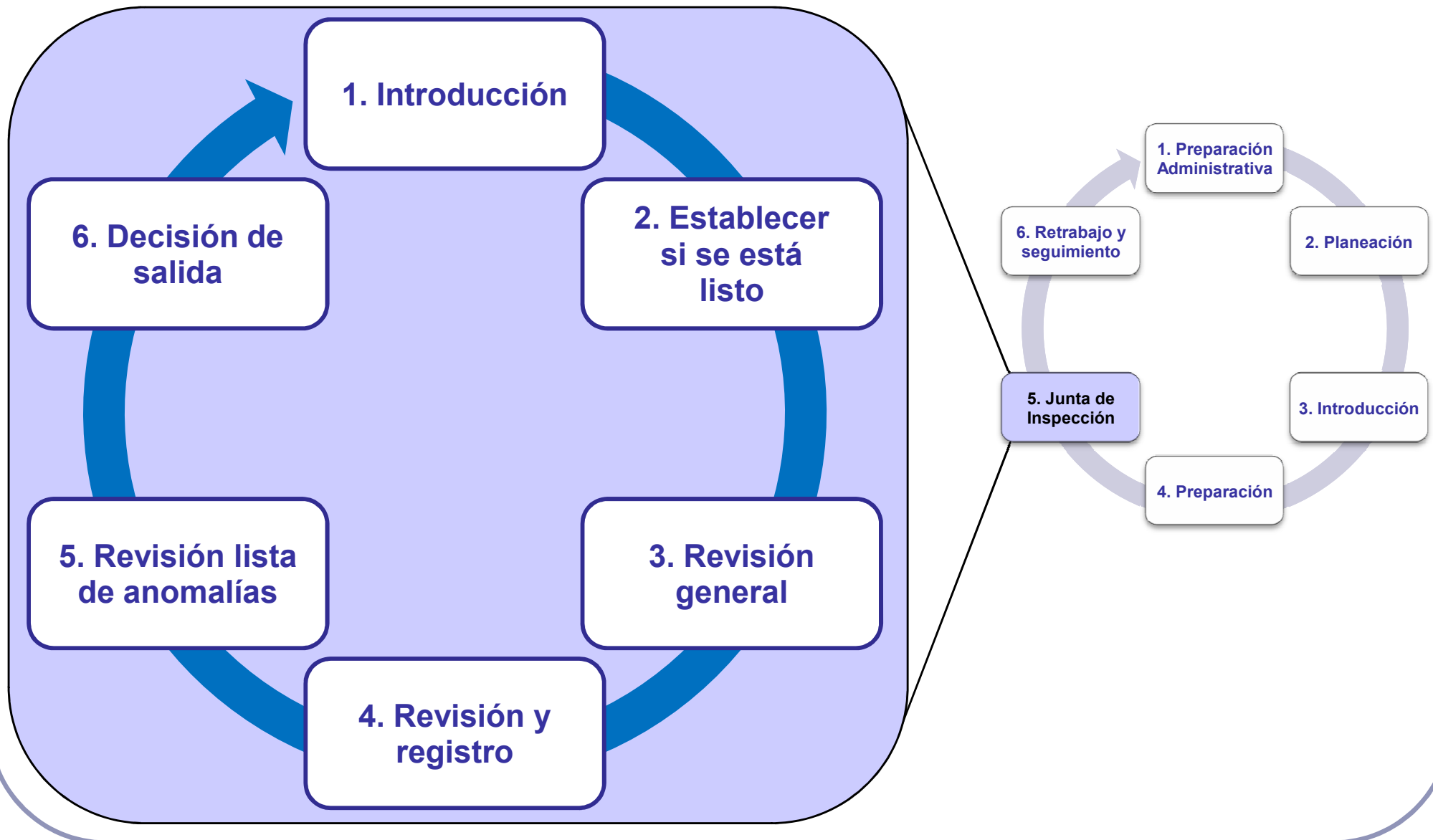
- Líder (moderador)
- Escriba
- Lector
- Autor
- Inspector
  
- NOTAS:
  - Todos son inspectores
  - El autor no puede ser ni líder, ni escriba, ni lector
  - Los otros roles se pueden compartir



# Proceso de una inspección



# Junta de inspección (paso 5)



# **Diferencias del recorrido vs. la inspección (1)**

---

- Propósito del recorrido
  - Menos formal
  - Puede ser educativo
- Participantes en el recorrido
  - Pueden ser sólo dos (autor + revisor)
- Roles en el recorrido
  - Normalmente el autor la dirige y es el lector

# **Diferencias del recorrido vs. la inspección (2)**

---

- Entradas en el recorrido
  - No se requieren procedimientos documentados, formas de reporte, checklists, etc.
- Salidas del recorrido
  - Normalmente no se colecta información (duración, tamaño del producto, tiempo de preparación, etc.)

# **Diferencias del recorrido vs. la inspección (3)**

---

- Planeación del recorrido
  - No se asignan roles
  - No hay cronograma
- Paso de “introducción” en el recorrido
  - Es más corto e informal
- “Junta de revisión” en el recorrido
  - No se realizan los siguientes pasos
    - Establecer si se está listo
    - Revisión general
    - Revisión de la lista de anomalías
    - Decisión de salida