Tolerancia a Fallos

En presencia de fallos, el sistema distribuido continúa operando en **forma aceptable**.

- **Dependable Systems.**
- **Garantizar** comportamiento en distintas condiciones.
- Prevenir **de cara al usuario**.
- Nivel de tolerancia.
- **Fallo** (parcial) -> **Error** (en el estado del sistema) -> **Falla/Avería** (comportamiento incorrecto).

Clasificación de fallos

Según **frecuencia**:

- **Transientes.** Una vez y desaparecen. Repetir lo arregla.
- **Intermitentes.**
- **Permanentes.** H/ reemplazar componente defectuoso.

Según **tipo**:

- **Crash.**
- **Timing.**
- **Omisión.**
- **Respuesta.** Valor incorrecto.
- **Arbitraria o Bizantina.**
  - En tiempos y respuesta.
  - Distinta info para distintos consumidores.

Condiciones

- **Del entorno.**
  - Entorno físico del hardware.
  - Interferencia y ruido.
  - Drifts de relojes.
- **Operacionales.**
  - Especificaciones
  - Networking.
  - Protocolos.

Detección de errores

- **Fault Removal.** Removerlos antes de que pasen.
- **Fault Forecasting.** Probabilidad de que un componente falle.
- **Fault Prevention/Avoidance.** Evitar condiciones que llevan a generarlos.
- **Fault Tolerance.** Aceptar los errores y tratarlos en el sistema.

Frente a errores

- **Resiliencia:** mantener nivel aceptable en presencia de fallos y desafíos.
- **Degradación suave:** difiere del comportamiento normal pero sigue siendo aceptable.
- **Enmascarado de errores.**
  - Tolerar mediante **redundancia**.
    - Física = **replicación**.
    - De información = **valor**.
    - De tiempo = **retries**.
- **Replicación.** Evitar SPOF.
- **Recuperación** de un error y llevarlo a estado correcto.

  - Almacenamiento estable.
  - Checkpointing (periódico).
  - Message logging (repetir desde checkpoint).
  - Consenso.

Tipos de Replicación

- **Pasiva.** Una primaria y varias secundarias/backup.
- **Activa.** Múltiples máquinas hacen lo mismo. Orden total.
- **Semi-activa (Leader-Follower).** Un lider toma decisiones no determinísticas.

Confiabilidad

Dependability. Medida de confianza en el sistema.

- **Availability.**
- **Reliability.**
- **Maintainability.** Ciclo de despliegue, provee:
  - Inmutabilidad.
  - Resiliencia.
  - Desacoplamiento.
- **Safety.** El sistema debe poder ser recuperado automática o manualmente ante cualquier falla.

Coordinación y Acuerdo

Exclusión mutua distribuida

- Obtener **acceso exclusivo** a un **recurso disponible p/ la red**.
- Pasaje de mensajes.
- Requerido:
  - **Safety:** solo un proceso a la vez.
  - **Liveness:** evitar starvation, espera eterna de mensajes.
  - **Fairness:** c/ proceso misma prioridad. In-order processing.

Algoritmos

- **Servidor central.** Un coordinador de la sección crítica.
  - Se sabe identificar el recurso.
  - Requests encolados (FIFOs).
  - Acceso *time-bounded*.
- **Token Ring.**
  - El token circula por el anillo.
  - Acceso "por turnos".
- **Ricart & Agrawala.**
  - Cuando querés acceder:
    1. `Request` con **timestamp** del proceso, **ID** y **nombre** del recurso.
    2. Enviar a todos.
    3. Esperar `OK` de todos.
    4. Entrar.
  - Cuando recibis `Request`:
    1. Envía `OK` si no está interesado.
    2. Si tiene la seccion, no responde y lo encola.
    3. Si está esperando, se comparan timestamps. El del **timestamp menor gana**.
       * El perdedor envía `OK`.

    * El ganador encola request.
- Cuando terminas de usar la sección, mandás `OK` a todos los encolados.