

# 1. Fondamenti e Notazione Visuale

## 1.1 Elementi Base - Come Disegnarli

### Nodi di Azione

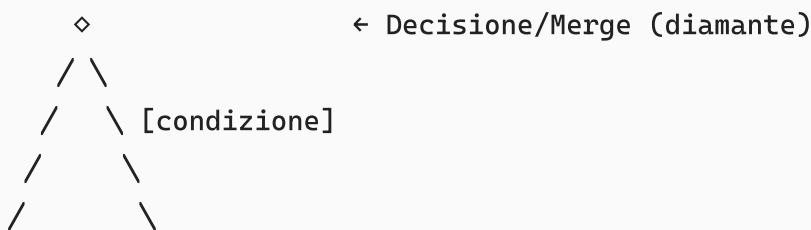


← Rettangolo con angoli arrotondati

### Nodi di Controllo

- ← Nodo iniziale (cerchio pieno nero)
- ← Nodo finale (cerchio con bordo spesso)
- ⊗ ← Fine flusso (cerchio con X)

### Decisioni e Merge



← Decisione/Merge (diamante)

### Fork e Join (Concorrenza)



← Fork/Join (barra spessa nera)

## 1.2 Come Costruire un Diagramma - Metodologia Step-by-Step

### STEP 1: Identifica il Processo

Domande da porsi:

- Qual è l'**obiettivo finale** del processo?
- Chi sono gli **attori coinvolti**?
- Quali sono le **azioni principali**?
- Ci sono **decisioni** da prendere?
- Esistono **attività parallele**?

## STEP 2: Traccia il Flusso Principale

Sempre iniziare con il "happy path"

- ```
● → [Azione 1] → [Azione 2] → [Azione 3] → ◎
```

## STEP 3: Aggiungi le Decisioni

Identifica i punti di scelta

- ```
● → [Azione 1] → ◇ → [Azione 2A] → ◎  

    ↓ [condizione B]  

    [Azione 2B] → ◎
```

## STEP 4: Introduci Swimlanes (se necessario)

Quando ci sono più attori

## 2. Esempi Progressivi con Costruzione Guidata

### Esempio 1: Login Sistema (SEMPLICE)

**Analisi Preliminare:**

- **Obiettivo:** Autenticare l'utente
- **Attori:** Utente, Sistema
- **Azioni:** Inserimento dati, controllo, risposta
- **Decisioni:** Credenziali valide/non valide

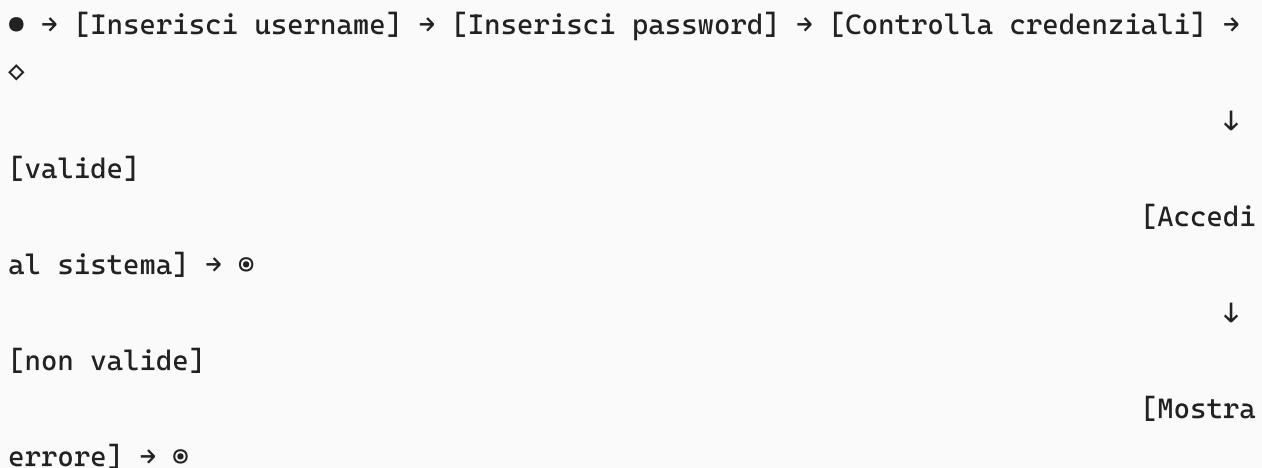
**Costruzione Step-by-Step:**

**STEP 1 - Flusso base:**

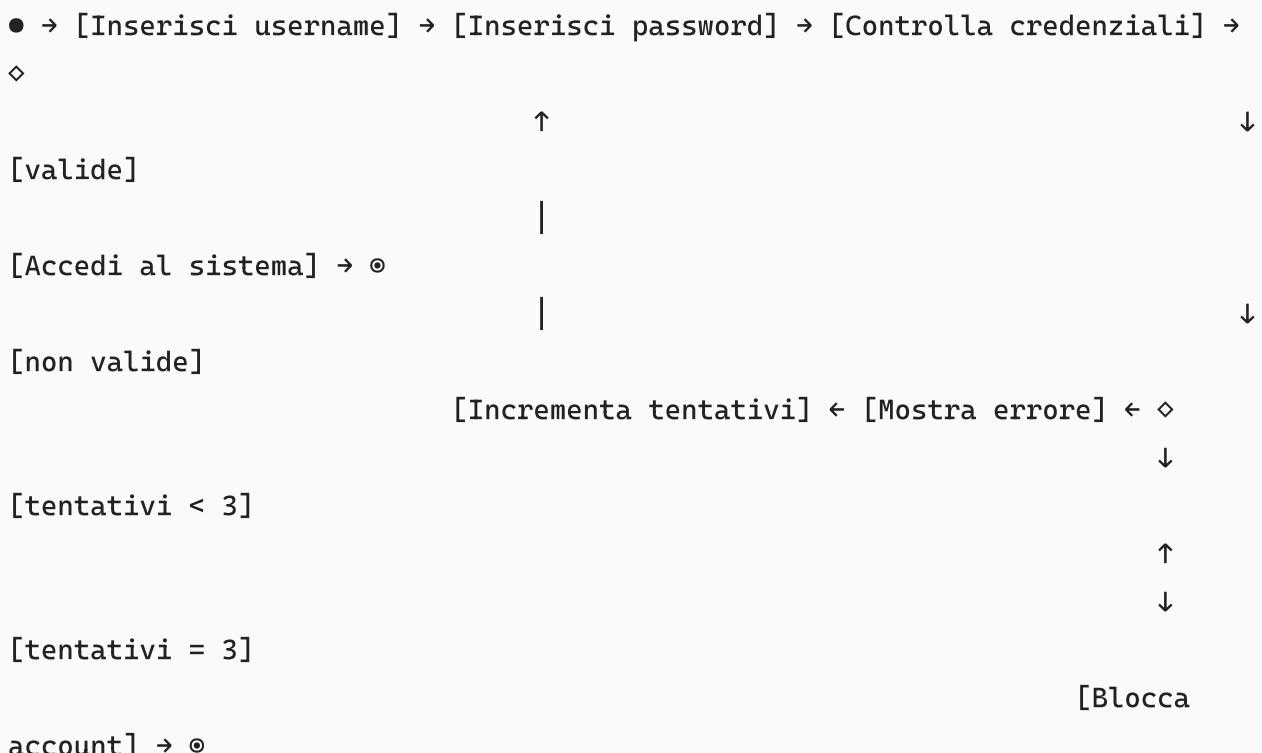
- ```
● → [Inserisci username] → [Inserisci password] → [Controlla credenziali] →  

    ◎
```

**STEP 2 - Aggiungi decisione:**



### STEP 3 - Gestisci tentativi multipli:



## Esempio 2: Ordine E-commerce (INTERMEDIO)

### Analisi:

- **Obiettivo:** Completare un acquisto
- **Attori:** Cliente, Sistema, Magazzino, Pagamento
- **Decisioni:** Stock disponibile, pagamento riuscito
- **Concorrenza:** Verifica stock + preparazione spedizione

### Costruzione con Swimlanes:

CLIENTE: ● → [Seleziona prodotto] → [Aggiungi al carrello] → [Vai al checkout] → [Inserisci dati pagamento] → [Attendi conferma]

SISTEMA: ↓ ↓ ↓

↓  
[Mostra dettagli] → [Aggiorna carrello] → [Calcola totale] →  
[Valida dati] → ◊

↓ [dati ok]

[Elabora pagamento] → ◊

↓ [successo]

MAGAZZINO:

[Verifica stock] → ◊

↓ [disponibile]

[Prepara spedizione] → [Spedisci] → ◎

↓ [non disponibile]

[Notifica mancanza] → ◎

↓ [fallimento]

PAGAMENTO:

[Gestisci errore] → ◎

## Esempio 3: Sistema di Backup (AVANZATO CON CONCORRENZA)

### Analisi:

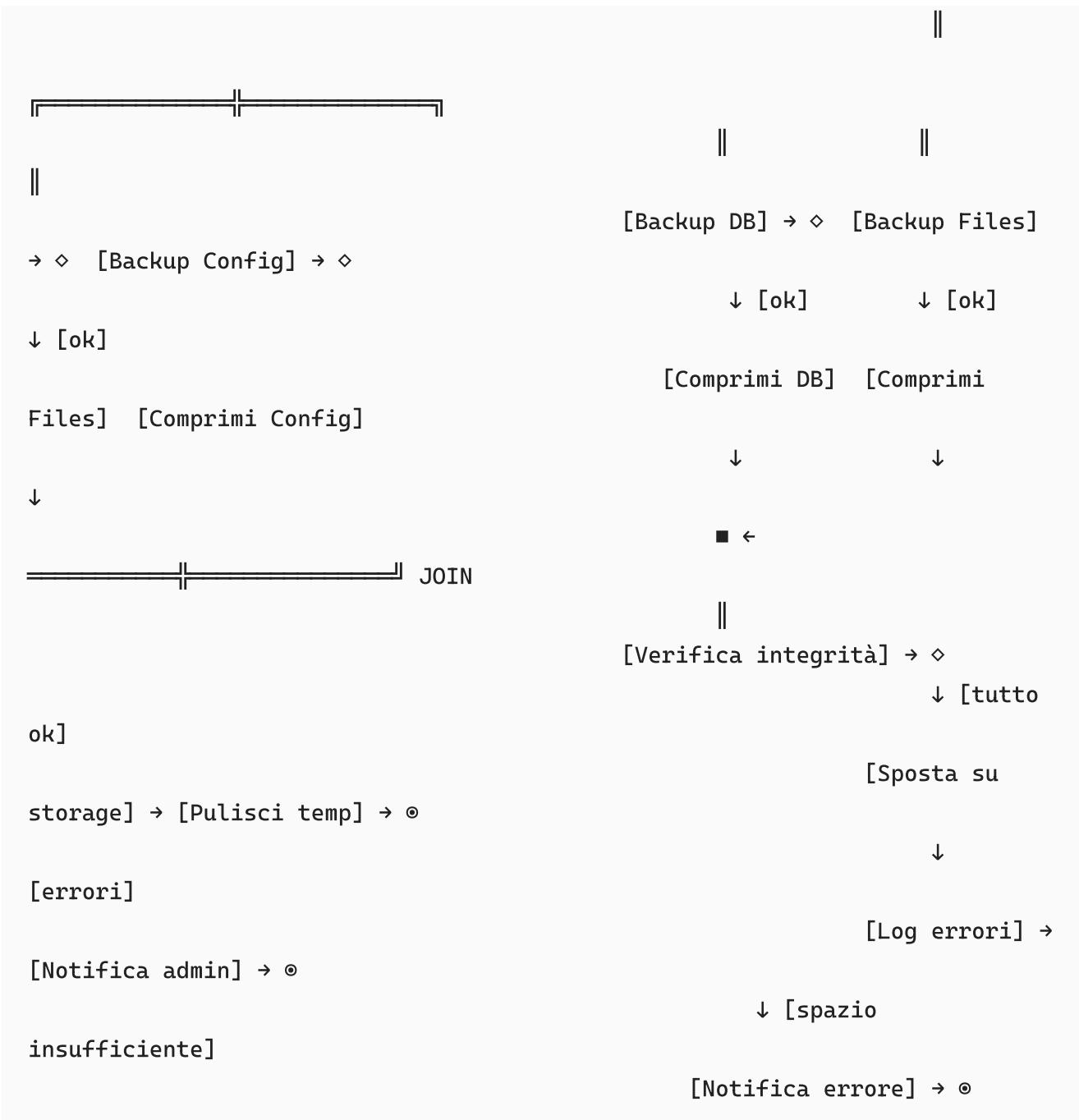
- **Obiettivo:** Backup completo sistema
- **Concorrenza:** Backup database + backup file + backup configurazioni
- **Sincronizzazione:** Tutti i backup devono completarsi

### Diagramma con Fork/Join:

● → [Inizializza backup] → [Controlla spazio disco] → ◊

↓ [spazio ok]

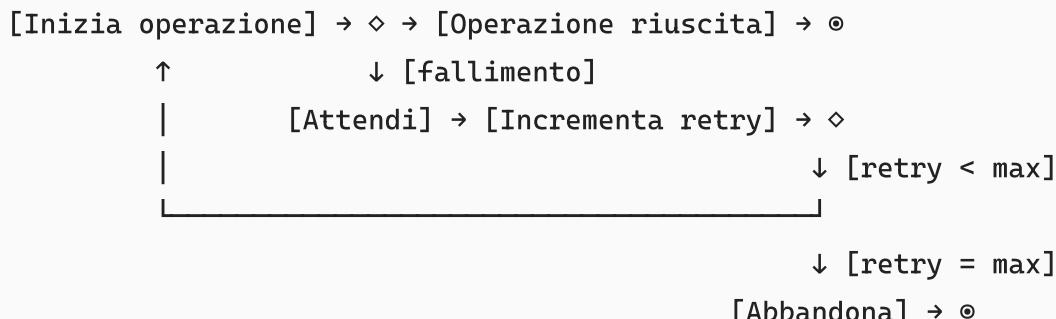
[Avvia backup] → ■ ← FORK



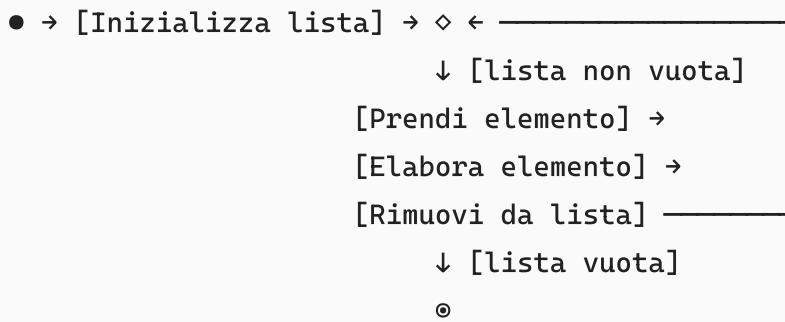
### 3. Come Gestire Situazioni Specifiche

#### 3.1 Loop e Iterazioni - Come Modellarli

**Situazione: "Riprova fino al successo"**

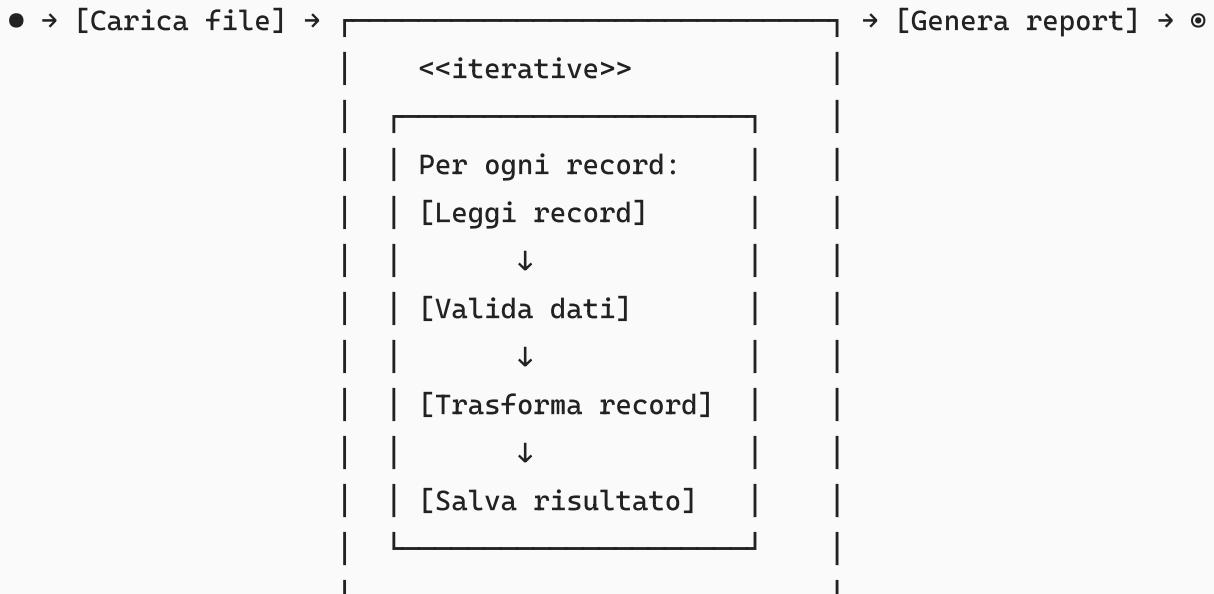


## Situazione: "Elabora tutti gli elementi"



## 3.2 Regioni di Espansione - Quando e Come Usarle

### Situazione: Processamento Batch



## 4. Esercizi Guidati con Soluzione Step-by-Step

### Esercizio 1: ATM Bancomat

#### Traccia:

Modellare il processo di prelievo da ATM:

- Inserimento carta e PIN
- Massimo 3 tentativi per PIN
- Selezione operazione (prelievo/saldo)
- Se prelievo: controllo saldo sufficiente
- Erogazione denaro
- Restituzione carta

## Soluzione Guidata:

## **STEP 1 - Identifica attori e azioni:**

- **Attori:** Cliente, ATM, Banca
  - **Azioni principali:** Inserisci carta, inserisci PIN, seleziona operazione, preleva
  - **Decisioni:** PIN corretto, saldo sufficiente
  - **Loop:** Tentativi PIN

## **STEP 2 - Flusso base:**

- → [Inserisci carta] → [Inserisci PIN] → [Seleziona operazione] → [Preleva denaro] → [Restituisci carta] → ☺

### **STEP 3 - Aggiungi controlli:**

- → [Inserisci carta] → [Inserisci PIN] → [Verifica PIN] → ◊
    - ↓ [PIN corretto]
    - [Seleziona operazione]
  - ◊

↓ [prelievo]

[Inserisci importo] → [Verifica saldo] → ◊

↓ [saldo ok]

[Eroga denaro] → [Restituisci carta] → ◉

↓ [saldo insufficiente]

[Mostra errore] → [Restituisci carta] → ⚡

↓ [saldo]

[Mostra saldo] → [Restituisci carta] → ⓧ

↓ [PIN errato]

[Incrementa tentativi]

→ ◊

↓ [tentativi < 3]

↑ (ritorna a inserisci PIN)

↓ [tentativi = 3]

[Trattieni carta] → ◎

#### STEP 4 - Aggiungi swimlanes:

CLIENTE: ● → [Inserisci carta] → [Inserisci PIN] → [Seleziona operazione] → [Inserisci importo] → [Prendi denaro] → [Prendi carta] → ◎

ATM: ↓ ↓ ↓ ↓  
↓ ↓ ↓ ↓  
[Leggi carta] → [Verifica PIN] → ◇ → [Mostra menu] → [Verifica saldo] → ◇ → [Eroga denaro] → [Restituisci carta]  
↓ [PIN errato]  
↓ [saldo ok]  
[Incrementa tentativi] → ◇  
↑  
↓ [< 3  
tentativi] ↑  
↑  
↓ [saldo insufficiente]  
↓ [= 3  
tentativi] [Mostra errore] → [Restituisci carta] → ◎  
[Trattieni  
carta] → ◎

BANCA: ↓  
↓

[Autentica] → [Risposta PIN]  
[Controlla conto] → [Risposta saldo]

## Esercizio 2: Processo di Recruitment

### Traccia:

Sistema di selezione del personale:

- Candidato invia CV
- HR fa screening iniziale
- Se passa: colloquio tecnico con team lead
- Se passa: colloquio finale con manager
- Decisione finale e comunicazione

- Se accettato: processo onboarding

## Soluzione Step-by-Step:

## **Analisi preliminare:**

- **Attori:** Candidato, HR, Team Lead, Manager
  - **Flusso seriale** con decisioni multiple
  - **Punti di uscita** a ogni fase

CANDIDATO: • → [Invia CV] → [Attendi risposta] → [Colloquio tecnico] → [Colloquio finale] → [Attendi decisione] → [Processo onboarding] → ◎

↓ ↓ ↓ ↓

HR: [Ricevi CV] → [Screening] → ◊ → [Organizza colloquio] →

[Coordina processo] → [Comunica decisione] → [Avvia onboarding]

↓ [non idoneo]

[Invia rifiuto] → ⓧ

**TEAM LEAD:**

## → [Colloquio tecnico] →

[Valutazione] → ◊

↓ [positivo]

[Approva candidato]

↓ [negativo]

[Respingi candidato] → ◎

**MANAGER:**

→ [Colloquio finale] → [Decisione] → ◊

↓ [assumi]

[Approva assunzione]

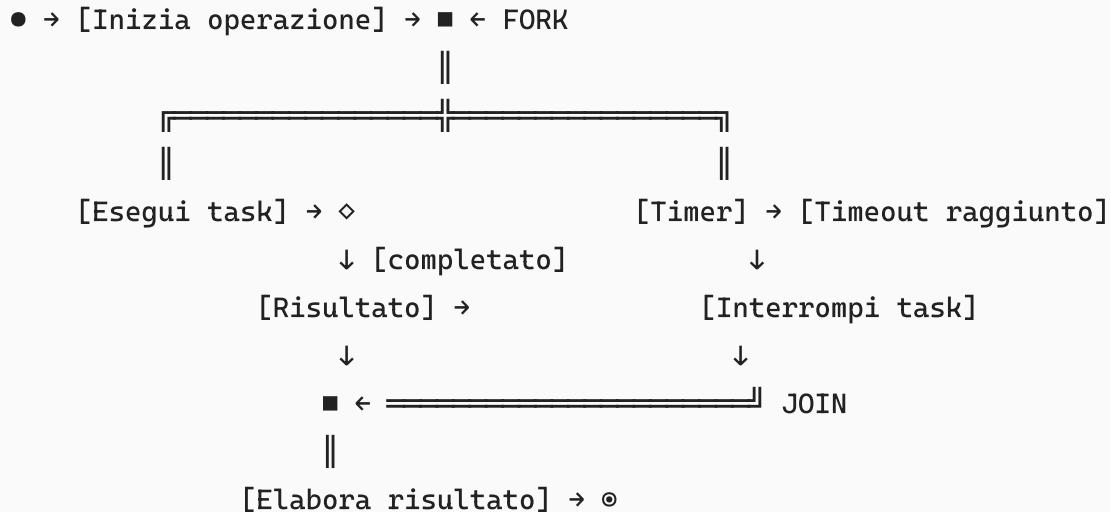
↓ [rifiuta]

[Respingi candidato] → ◎

## 5. Trucchi e Tecniche Avanzate

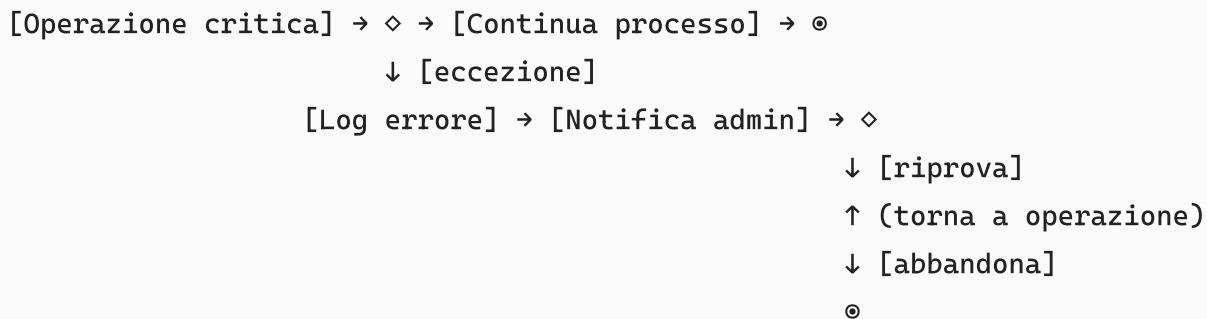
## 5.1 Come Gestire Timeout Elegantemente

### Problema: Azioni con scadenza temporale



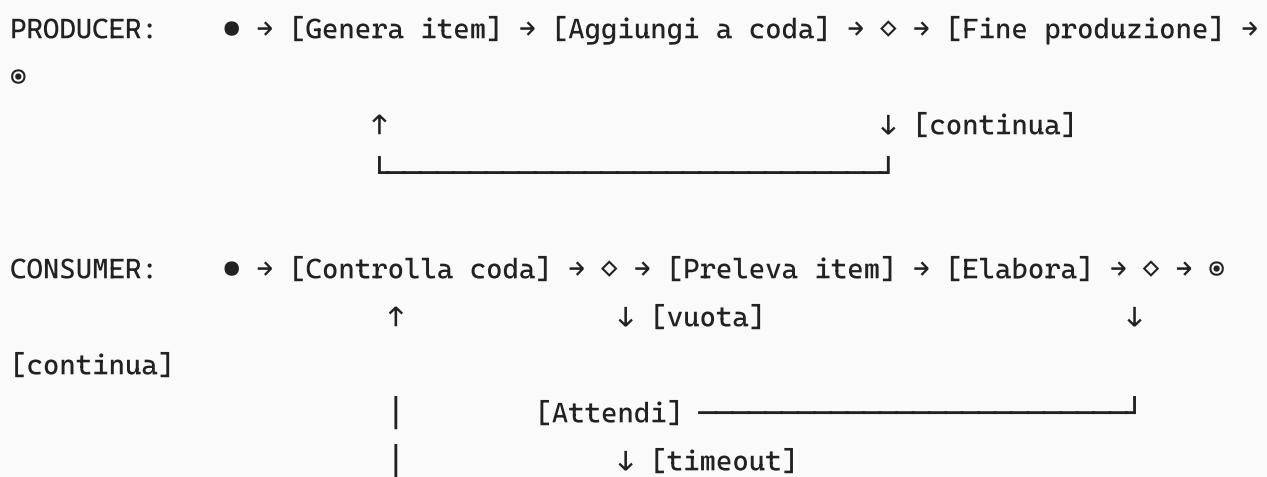
## 5.2 Gestione Eccezioni Sistematiche

### Pattern: Exception Handling



## 5.3 Pattern Producer-Consumer

### Situazione: Elaborazione asincrona



## 6. Checklist per Diagrammi Perfetti

### Struttura Base

- Un solo nodo iniziale •**
- Almeno un nodo finale ◎**
- Tutti i nodi connessi** (nessun elemento isolato)
- Fork/Join bilanciati** (stesso numero di rami)

### Flusso Logico

- Ogni decisione ha tutte le alternative coperte**
- Guard conditions mutuamente esclusive e complete**
- Loop hanno condizioni di uscita chiare**
- Timeout gestiti appropriatamente**

### Swimlanes

- Una corsia per attore principale**
- Azioni nella corsia corretta**
- Flussi tra corsie giustificati**
- Responsabilità chiare** per ogni azione

### Leggibilità

- Nomi azioni chiari e specifici**
- Layout ordinato** (da sinistra a destra, top-down)
- Frecce chiare** e non intrecciate
- Dimensioni appropriate** per la complessità

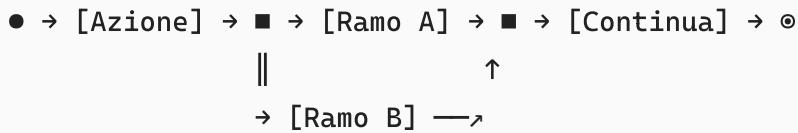
## 7. Errori Tipici e Come Evitarli

### Errore 1: Fork Senza Join

Sbagliato:



**Corretto:**



## ✗ Errore 2: Decisioni Incomplete

**Sbagliato:**

```
[Verifica] → ◊ → [Successo] → ○  
(manca il ramo "fallimento")
```

**Corretto:**

```
[Verifica] → ◊ → [Successo] → ○  
      ↓ [fallimento]  
      [Gestisci errore] → ○
```

## ✗ Errore 3: Swimplanes Confuse

**Sbagliato:**

```
UTENTE:      [Login] → [Valida credenziali] → [Mostra dashboard]  
SISTEMA:          (validazione è del sistema, non utente)
```

**Corretto:**

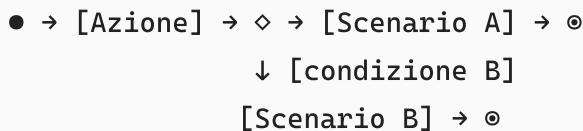
```
UTENTE:      [Inserisci credenziali] → [Visualizza dashboard]  
SISTEMA:          → [Valida credenziali] → [Autorizza  
accesso]
```

## 8. Template di Riferimento Rapido

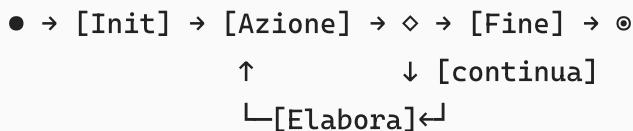
### Template 1: Processo Lineare Semplice

```
● → [Azione 1] → [Azione 2] → [Azione 3] → ○
```

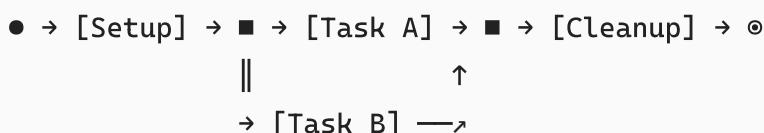
### Template 2: Con Decisione



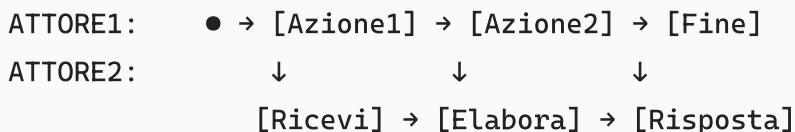
## Template 3: Con Loop



## Template 4: Con Concorrenza



## Template 5: Multi-Attore



## 9. Consigli per l'Esame

### Strategia Temporale (30 minuti)

1. **Minuti 1-5:** Leggi e analizza la traccia
2. **Minuti 6-10:** Identifica attori e flusso principale
3. **Minuti 11-20:** Disegna il diagramma base
4. **Minuti 21-25:** Aggiungi decisioni e casi speciali
5. **Minuti 26-30:** Verifica con checklist

### Cosa Impressiona i Correttori

- **Swimlanes ben organizzate** con responsabilità chiare
- **Gestione completa delle eccezioni** (non solo happy path)
- **Fork/Join utilizzati correttamente** per concorrenza reale
- **Guard conditions precise** e mutuamente esclusive
- **Layout pulito e leggibile**

### Cosa Penalizza

- **Diagrammi illeggibili** con frecce intrecciate
- **Decisioni incomplete** (mancano rami)
- **Fork senza Join** corrispondenti
- **Azioni nelle swimlanes sbagliate**
- **Mancanza di nodo iniziale o finale**

Ricorda: **Un diagramma delle attività deve raccontare una storia chiara del processo - se non riesci a spiegarlo a parole, probabilmente il diagramma non è corretto!**