Errors and Fixes in Experiment Models

# I. Online Order Processing – Errors

**1. Non-interrupting boundary timer may cause a deadlock.**

Fix: Replace it with an interrupting boundary event to ensure flow continues properly.

**2. Exclusive gateway used where parallel execution is required.**

Fix: Replace the XOR gateway with a parallel gateway to enable simultaneous shipment preparation and inventory update.

**3. Non-interrupting timer event incorrectly leads to the final task.**

Fix: The timer should notify customer service only, without interfering with the main dispatch flow.

**4. Missing merging gateway causes lack of synchronization.**

Fix: Add a parallel merging gateway before the dispatch task to synchronize the parallel paths.

**5. Non-interrupting timer attached to the wrong task.**

Fix: Move the timer to the border of the 'Prepare Shipment' task, where the timing condition is relevant.

**6. Lack of a label on the timer boundary event.**

Fix: Add a descriptive label to the timer boundary event to clarify its function.

# II. Post-Delivery Cancellation and Quick Complaint Resolution – Errors

**1. Parallel gateway used instead of event-based gateway for awaiting input.**

Fix: Use an event-based gateway to correctly wait for either a message or timer event.

**2. Incorrect use of intermediate event-based gateway at process start.**

Fix: Replace it with a start event-based gateway, which can instantiate the process.

**3. Too many start events are defined.**

Fix: Consolidate them by using a single event-based gateway to initiate the process.

**4. Flow ends with another event-based gateway.**

Fix: Event-based gateways should only appear at the start of a decision, not at the end of a flow.

**5. Duplicate end event labels reduce clarity.**

Fix: Give each end event a unique label (e.g., 'Refund Issued', 'Replacement Sent').

**6.** **Generic names used for intermediate catch events (e.g., ‘Event 1’, ‘Event 2’).**

Fix: Rename the events to reflect their purpose, such as ‘Cancellation Request Received’ or ‘Quick Complaint Received’.

# III. Product Catalog Update – Errors

**1. Exclusive gateway used instead of inclusive gateway.**

Fix: Use an inclusive gateway to allow multiple catalog updates (add, update, remove) simultaneously.

**2. Missing end event despite presence of start event.**

Fix: Add an end event after 'Publish Updated Catalog' to explicitly show process completion.

**3. Lack of merging gateway before an exclusive gateway.**

Fix: Use an inclusive merging gateway to synchronize flows before the next decision point.

**4. Parallel paths flow into a task without synchronization.**

Fix: Add an inclusive merge before 'Publish Updated Catalog' to collect all possible updates.

**5. Boundary timer used incorrectly on a task.**

Fix: Replace it with a timer start event, which better reflects the periodic nature of the process.

**6. Missing condition labels on outgoing flows from the inclusive gateway.**

Fix: Add meaningful condition labels to all outgoing flows to indicate decision logic.

# IV. Customer Inquiry Management – Errors

**1. Broken flow continuity in private process.**

Fix: Ensure a continuous sequence flow connects all tasks within the internal process.

**2. Gateway used without a prior decision task.**

Fix: Add a 'Categorize Inquiry' task before the gateway 'What type of inquiry is it?'

**3. Poorly structured flow ('spaghetti diagram').**

Fix: Reorganize layout to improve readability of control and sequence flows.

**4. Incorrect pool and lane usage.**

Fix: Place customer interactions in a separate pool from internal teams.

**5. Parallel gateway routes inquiries to all departments.**

Fix: Replace with an exclusive gateway to route the inquiry based on its type.

**6. Missing labels for swimlanes and pools.**

Fix: Add lane and pool names (e.g., ‘Customer’, ‘Support Team’) to clarify participant responsibilities.

# V. Returns and Exchanges Handling – Errors

**1. Inspection step skipped before decision point.**

Fix: Add an 'Inspect Returned Item' task before deciding between refund or exchange.

**2. Refund and exchange paths modeled in parallel.**

Fix: Use an exclusive gateway to reflect the either/or nature of this decision.

**3. Quality inspection task placed in the wrong swimlane.**

Fix: Move 'Inspect Returned Item' to the Logistics lane, where it logically belongs.

**4. Unlabeled gateways and conditions.**

Fix: Clearly label all exclusive gateways and outgoing paths to clarify decision logic.

**5. Unconnected 'Receive Returned Item' task.**

Fix: Connect it properly to both 'Send Return Instructions' and 'Inspect Returned Item' to maintain flow logic.

**6. Disorganized flow layout with zigzagging connectors at inconsistent angles (the so called ‘slalom’ structure).**

Fix: Reorganize the process from left to right and align flows to enhance visual readability.