RISK AND RESILIENCE BOOTCAMP

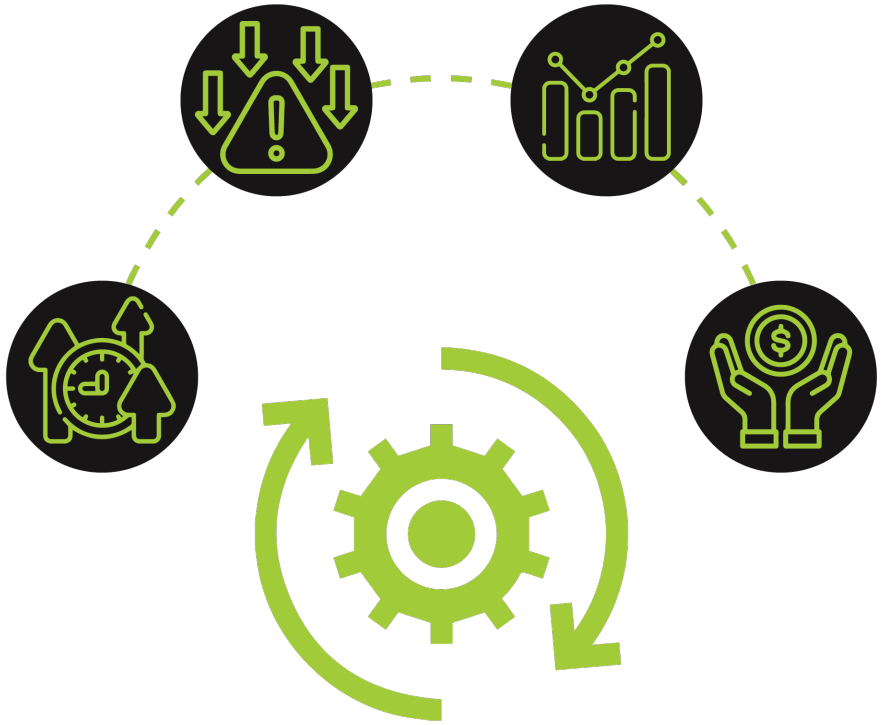
TEKsystems Global Services

WORKFORCE DEVELOPMENT

BUSINESS PROCESS MODELING

This module introduces and explores business process modeling and related areas

- BPMN
- Bottleneck Analysis
- RACI
- Control Weaknesses



BUSINESS PROCESS MODELING (BPM)

- Why model business processes?
  - Organization manage risks from systems, controls and people
  - But often without a full picture of how work actually flows through the business
- BPM helps to visualize, analyze and improve workflows
  - Essential for revealing where risks lie
- In a risk and resilience context
  - Helps identify where inside the process might things go wrong
  - And what the resulting impacts would be

BUSINESS PROCESS MODELING (BPM)

- Modeling a process provides
  - Clarity about inputs, outputs, flows, handoffs, decisions and system dependencies
  - Common vocabulary and visualization across business & IT stakeholders
  - Basis for identifying latent risks and control weaknesses
    - For example: rework loops, human manual steps, system dependencies
- A process map is not just a diagram of work
  - It is also a diagram of risk

BPM IN RISK IDENTIFICATION

- Applying BPM with a risk lens can uncover
  - *Design risks*
    - Does the process design introduce delays, loops, unnecessary manual work or complexity?
  - *Execution risks*
    - Are there unmonitored manual handoffs, people acting without backup, or lack of system support?
  - *Dependency risks*
    - Does the process rely on a single system or person (single point of failure)?
    - Are there external suppliers or upstream/downstream processes that create risk?
  - *Control risks*
    - Where are the monitoring or control points?
    - Are they visible and documented?
    - Are there gaps or overlaps in ownership?

BPM IN RISK IDENTIFICATION

- Applying BPM with a risk lens can uncover:
  - *Data/information risks*
    - How does data move?
    - Is there risk of data loss, corruption, incorrect handoff?
  - *Change risks*
    - If a process changes (e.g., new system, new regulatory requirement), how will the current flows adapt and what is the vulnerability during transition?
- Modeling shifts the risk lens
  - From: *"we hope nothing goes wrong"*
  - To: *"we can see where things can go wrong and can act"*

MODELING TECHNIQUES AND TOOLS

- There are a number of standard toolsets used for business modeling
  - Business Process Model and Notation (BPMN) 2.0
    - The industry-standard notation for modelling complex business processes
    - Uses standard symbols (events, activities/tasks, gateways/decisions, flows) that both business and technical stakeholders can understand
    - Allows modeling from a “high-level overview” down to more detailed subprocesses, enabling the modeler to drill down to the risk source
    - Using BPMN for risk modelling: some researchers suggest enriching BPMN with risk information (likelihood/impact): e.g., in a “risk-annotated BPMN” model

MODELING TECHNIQUES AND TOOLS

- Some tools are adaptations of older existing modeling techniques
  - Flowcharts, SIPOC diagrams, Value-Stream Maps
    - These are useful, especially in earlier or higher-level phases, or for less complex processes
  - Flowcharts
    - Simple, good for mapping steps, but may lack standardisation and clarity for cross-functional work
  - SIPOC (Supplier–Input–Process–Output–Customer) diagrams
    - Good for understanding the boundary/context of a process
    - Who supplies, what comes in, what goes out, who receives
  - Value-Stream Maps
    - Commonly used in Lean environments to show value-add vs non-value-add steps, handoffs, delays
    - Useful for spotting bottlenecks and waste, which are also risk indicators

RISK IDENTIFICATION STEP BY STEP

- Select the process to model
  - Choose a process that is critical to business operations or IT-risk related
    - For example: customer onboarding, change-management, incident response, data access provisioning
- Map the “as-is” process
  - Use BPMN or other notation to capture how work actually flows today
  - Include start/end points, tasks, decision points, handoffs, system steps, manual steps, exceptions
  - Use swimlanes/partitions if different departments or systems are involved
  - Annotate where the process uses systems, where people operate manually, where there are delays or dependencies

RISK IDENTIFICATION STEP BY STEP

- Overlay risk indicators on the model and ask:
  - Identify where are the handoffs occur
    - People to people
    - System to people, or vice vera
    - System to system
  - Identify manual steps or processes that could be automated or are error-prone
  - Identify loops or rework steps where there is a higher chance of delay or error
  - Is there a single person or system that if it fails, the process stops?
    - This is a single point of failure
  - Are the decision-points well documented, are there missing controls or lack of visibility?
  - How is data passed? Is it secure, accurate, validated?
  - What happens when exceptions occur? Are there contingency steps?

RISK IDENTIFICATION STEP BY STEP

- Risk modeling approaches often follow the process
  - Model the process
  - Use the model to prompt risk identification.
  - By analyzing each activity of a business process you are more likely to identify risks than by relying solely on unstructured brainstorming

RISK IDENTIFICATION STEP BY STEP

- Identify where control weaknesses exist
  - Highlight where controls are missing, ambiguous, redundant, or overlapping
  - For example
    - A manual approval step that has no logging or oversight
    - A task assigned to a person who also does approvals and execution (lack of segregation of duties)
    - A system dependency with no backup or no change control
- Prioritize risk areas
  - Based on the process model, prioritize:
    - Likelihood of failure/disruption
    - Impact of that failure (financial, regulatory, reputational, operational).
  - The visual depiction of the process is a way to justify your prioritization
    - For example, identifying a single point of failure at a critical step

RISK IDENTIFICATION STEP BY STEP

- Design improvements
  - Once the model is overlaid with the identified risks
  - The process can be re-engineered by
    - Proposing controls
    - Identifying redundant paths
    - Identifying automation opportunities
    - Improving monitoring to enhance detective controls
  - The model becomes a communication tool with stakeholders

FLOW OF RISK

- Traditional process modeling focuses on the flow of work
  - Adding a risk-aware lens, shifts focus to the flow of risk
  - How risk propagates through process steps, handoffs, and system interactions
  - For example:
    - If Task A feeds into Task B, and Task A is manual and error-prone
    - Then Task B can receive incorrect data
    - This is a risk flow because the risk flows from A to B
  - The process map visualizes the pathway of work *and* the pathway of potential failures, delays, rework, data leaks, control gaps
  - Visual modelling makes hidden dependencies visible
  - For example:
    - A system that triggers the next step but if it fails, the work stops
    - A manual check that is executed late or is incomplete
    - A hand-off between departments with unclear responsibility creating a risk blindspot

INTEGRATING BPM INTO RISK ANALYSIS

- BPM can be included foundation activity
  - Identify critical processes, map them, overlay risk and dependencies, feed those into the risk register or assessment
  - Use the diagrams in workshops with stakeholders
    - The visual nature aids understanding, builds engagement, and supports communication across business/IT
  - Treat the process models as living artifacts
    - Update when process changes occur, when new systems are introduced, or when control environment shifts
    - Otherwise they become stale and lose analytic value.
  - Link the process models to control frameworks
    - Mapping where controls reside or should reside
    - Map to resilience procedures like the resilience plan for the failure of a key step
  - Use the models during incident response or root cause analysis
    - When something failed, revisit the process model to see where the breakdown occurred

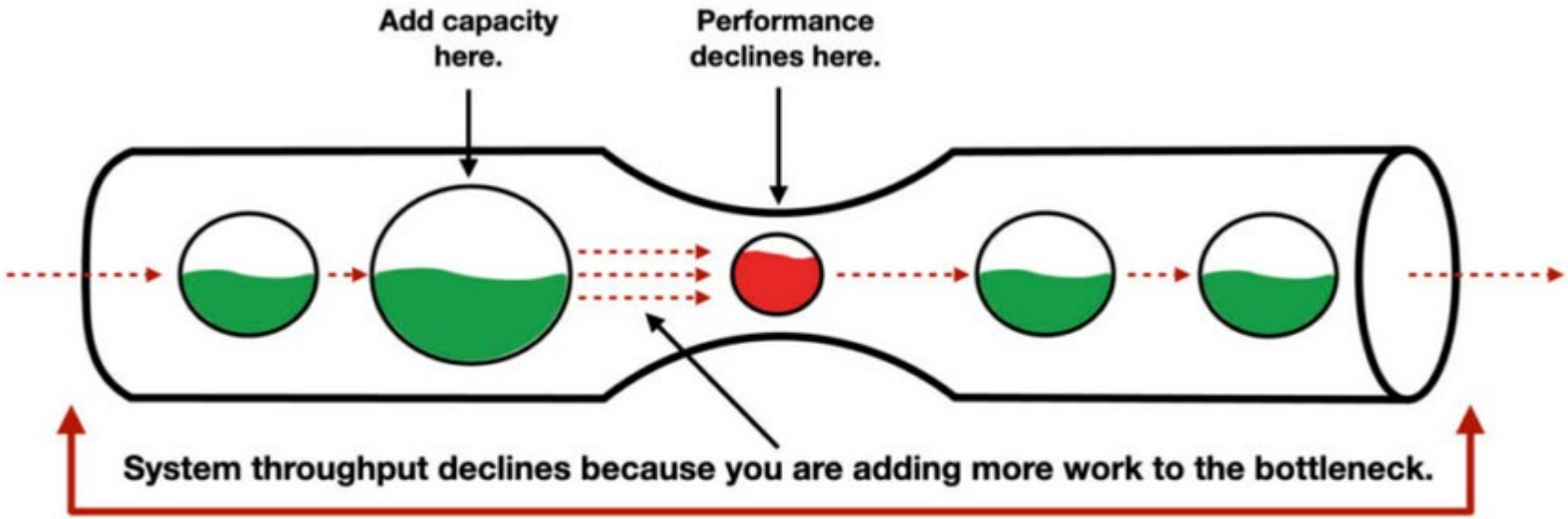
BOTTLENECKS

- Once a process is mapped, it becomes possible to assess operational and systemic vulnerabilities
  - Bottlenecks
    - Stages where tasks accumulate due to resource constraints, approval delays, or inefficient handoffs
    - Indicators include high cycle time, queue buildup, or repeated rework.
  - Single points of failure (SPOFs)
    - Individuals, systems, or tools whose failure halts the entire process
  - Examples
    - A single IT administrator controlling access
    - A critical spreadsheet managed by one person

BOTTLENECKS

- A place in a process where
  - The demand like work, tasks, information exceeds the capacity of that step like workers, system capacity, hand-offs or approvals
  - Slows down the overall flow.
- The rest of the process may be efficient
  - But the bottleneck effectively limits the throughput of the entire end-to-end chain
  - Analogous to the neck of a bottle limiting the flow of liquid.
  - Bottlenecks may be temporary (short-term) or structural/chronic (long-term)
  - Short-term could be a system outage or a person's absence
  - Long-term may be inherent capacity mismatch, workflow design fault, or outdated system

BOTTLENECKS



The theory of constraints defines a constraint as:

"Anything that limits the system from a higher level of performance." (Pretorius, 2014, p. 498)

BOTTLENECKS IN RISK

- From a risk perspective, bottlenecks are critical because they
  - Increase cycle time and lead time, thereby delaying deliverables and reducing responsiveness
  - Create back-logs or work queues, which increase exposure to error, rework, and potential control failures
  - May indicate hidden single points of failure
    - For example, if one stage cannot scale or is highly dependent on a resource
  - Amplify the impact of other risks
    - For example, a bottleneck in a process may mean that when an upstream system fails, the backlog grows and uncontrolled work continues, increasing operational and compliance risk
  - Erode organizational resilience
    - If the bottleneck is not addressed, the system cannot adapt quickly to spikes in demand, external shocks, or changes in process

COMMON TYPES OF BOTTLENECKS

- Capacity constraints:
  - A person, team, or system cannot process work as fast as it arrives.
  - For example: one expert reviewer approves all change requests and is overloaded
- Manual hand-offs or approvals
  - Tasks that require manual intervention, sign-off, or handover tend to be slower and more error-prone, introducing delay
- System performance or outdated technology
  - Legacy systems, slow response times, or batch processing can create bottlenecks

COMMON TYPES OF BOTTLENECKS

- Handoff/transition delays across departments
  - When work moves from one team or system to another (especially across silos) delays often occur
- Redundant, unclear or overly complex workflows
  - Too many steps, rework loops, unnecessary approvals all slow the process
- Resource or role dependency (single person, single team):
  - When only one person or team performs a critical step, the process becomes fragile
- Variability and unpredictable demand
  - Large fluctuations in work volume without matching capacity produce bottlenecks

HOW TO IDENTIFY BOTTLENECKS

- Step 1: Select the process scope
  - Choose a critical process
    - For example new-client onboarding, incident response escalation, change management
  - Define clear start and end points so you know what you're analyzing
- Step 2: Map the "as-is" workflow
  - Create a visual model (flowchart, BPMN, swimlane) showing steps, decision points, hand-offs, systems, manual tasks
  - Annotate
    - Where tasks queue
    - Where hand-offs happen
    - What systems are used
    - Who is responsible
  - Use data where possible
    - For example: cycle times, wait times, queue lengths.

HOW TO IDENTIFY BOTTLENECKS

- Step 3: Collect data and observe metrics
  - Key metrics
    - *Lead time:* time from request to completion, also called transaction time
    - *Cycle time:* time taken at each step
    - *Throughput:* units processed per period
  - Work-in-progress (WIP) queues
    - A step where WIP accumulates or where lead time spikes suggests a bottleneck
  - Obtain feedback from staff:
    - Which steps repeatedly cause frustration or backlog?
    - Which persons are over-utilized?

HOW TO IDENTIFY BOTTLENECKS

- Step 4: Identify the bottleneck(s)
  - Look for the step with the longest queue, highest utilization, longest processing time, or which restricts the entire process flow.
  - If this step were faster, would the overall process become faster?
    - If yes, than that step is a bottleneck.
  - Differentiate whether it is a constraint (systemic) or temporary overload

HOW TO IDENTIFY BOTTLENECKS

- Step 5: Analyze root causes
  - For an identified bottleneck, ask
    - Why is the backlog forming?
    - Why is the capacity insufficient?
    - Are there manual approvals?
    - Are resources constrained?
    - Are hand-offs inefficient?
  - Use techniques such as 5 Whys, fish bone diagrams
  - Examine the interfaces between people, process, system
    - For example: maybe the system is fine but hand-off between teams is slow because roles are unclear

HOW TO IDENTIFY BOTTLENECKS

- Step 6: Develop and implement solutions
  - Potential remediation
    - Redistributing workload
    - Adding capacity
    - Automating manual steps
    - simplifying workflow
    - Eliminating redundant steps
    - Establishing backup resources
    - Improving hand-off protocols.
  - Prioritize by impact
    - Fix the bottleneck that will yield the most improvement in overall process flow
    - Improving non-bottleneck steps yields little gain

HOW TO IDENTIFY BOTTLENECKS

- Step 7: Monitor and sustain
  - After implementing changes
  - Monitor metrics again to ensure the bottleneck is relieved
    - For example: lead time, cycle time, queue lengths
  - Bottlenecks may shift
    - Once one is removed, another may emerge elsewhere
    - Continuous monitoring is essential to identify emerging bottlenecks

SINGLE POINTS OF FAILURE

- Bottlenecks are about capacity and throughput
- Single points of failure (SPOFs)
  - SPOF is a person, system, or step whose failure (unavailability, error, outage) will stop or severely impair the process
  - A redundancy gap means there is no backup, alternate path, or contingency for a critical process step, rendering it fragile
- From process modelling:
  - A step may appear fine in normal flow, but if the person responsible is absent, or the system fails, the process halts, that is a SPOF
  - Bottlenecks + SPOFs = elevated risk because when a bottleneck resource fails, the harm is much greater
  - Mitigation: design alternate paths, cross-train resources, ensure fail-over systems, incorporate contingency in process maps

AUTOMATION ISSUES - PROCESS WEAKNESSES

- Automation issues are often linked to bottlenecks and SPOFs, automation plays an important role:
  - Manual tasks are often slower, error-prone, less visible, more susceptible to capacity constraints
  - These can become bottlenecks
  - Automation can improve throughput, consistency and reduce human dependency
  - But automation poorly designed or not maintained can itself become a bottleneck or failure point

AUTOMATION ISSUES - PROCESS WEAKNESSES

- When modelling a process from a risk view
  - Annotate where manual work occurs
  - Where systems are used
  - Where automation might reduce delay and risk
  - Where automation is missing (gap), or outdated (risk of failure)
- Example:
  - A manual approval system via email becomes overloaded and delayed; automating approval or providing self-service may reduce that bottleneck and improve resilience.

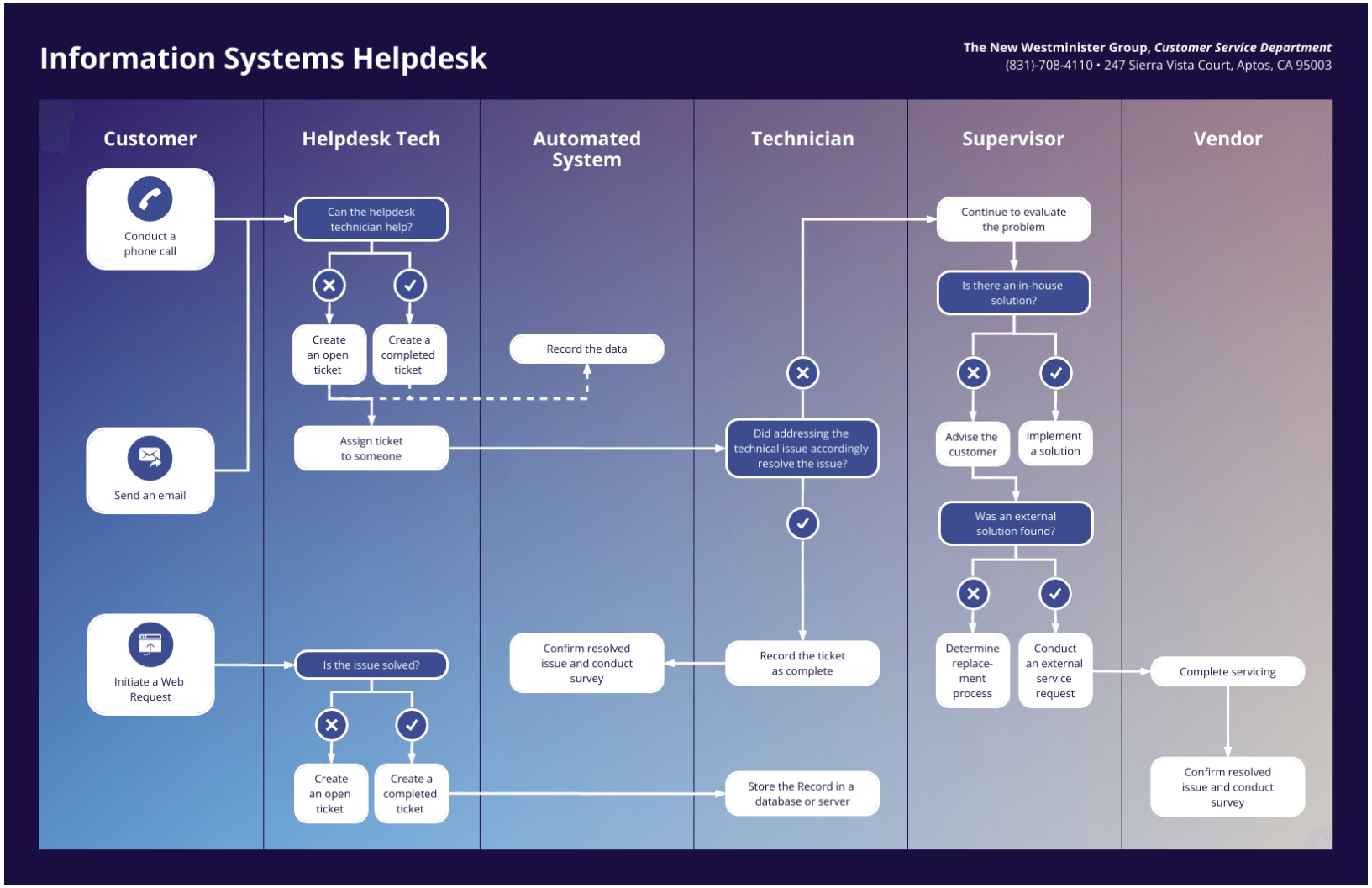
SUMMARY

- To recap
  - Bottlenecks are systematic constraints in a process that limit throughput, slow flow, increase backlog and introduce risk
  - They may be due to people, process, or system issues (or combination thereof) and often highlight underlying fragility or resilience gaps
  - Identifying bottlenecks requires mapping the process, collecting data (lead time, cycle time, WIP), gathering feedback, and analyzing root causes
  - From a risk and resilience viewpoint, watch for single points of failure and lack of redundancy, as these amplify the impact of bottlenecks
  - Automation (or lack thereof) is a key factor: manual tasks increase bottleneck risk, but automation must be well-designed or it can itself be a fault
  - Regular monitoring is essential: once one bottleneck is relieved, another may emerge because the process needs to be a living risk-aware artifact

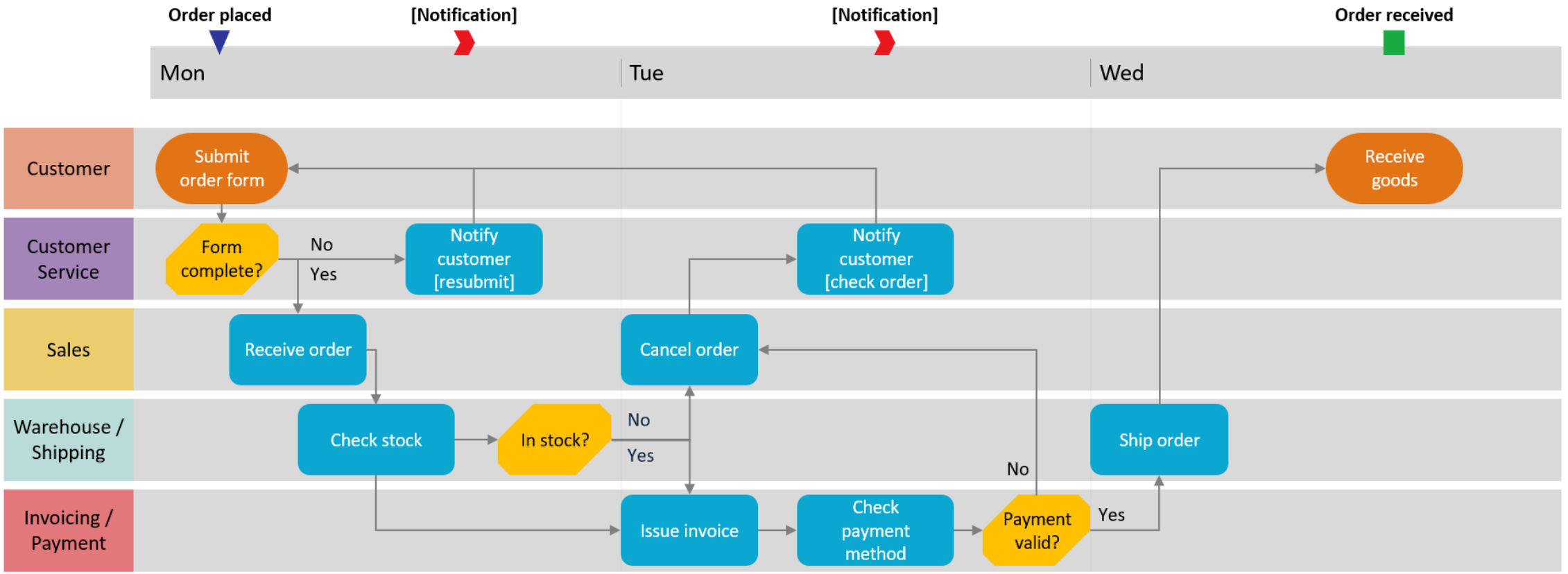
SWIMLANE DIAGRAMS

- A type of process map
  - Divides the workflow into “lanes” representing different roles, departments, teams or systems
  - Then places tasks/activities inside those lanes according to who/what is responsible
  - Also called a cross-functional flowchart or role-activity diagram
  - These can be applied to any sort of diagram that maps out a flow
    - Data flow diagrams, business process models, flowcharts, etc.
- The visual metaphor
  - Like a swimming pool, where each lane is a different actor and the tasks “swim” across lanes
  - Unlike a diagram that just shows sequence, a swimlane diagram adds an extra dimension of responsibility/ownership, making “who does what” and “hand-offs” explicit
  - It is used especially when processes span multiple functions/teams or when hand-offs between teams/systems are common

SWIMLANE DIAGRAMS



SWIMLANE DIAGRAMS



SWIMLANE DIAGRAMS

- In risk analysis, swimlane diagrams serve several important purposes
  - Visualize how work and information move across organizational boundaries
    - A risk often arises when there's a boundary crossing
    - Department to department, system to person, vendor to internal
    - Making that explicit helps highlight potential weak points
  - Clarify roles and responsibilities
    - If a task is shown in a lane labelled "IT Support"
    - But the subsequent decision appears under "Operations"
    - It is obvious who is accountable for what.
    - This helps uncover ambiguity in responsibilities, duplication of tasks, or gaps (no one assigned)

STEP-BY-STEP

- Step 1: Define the scope
  - Choose the process to model
    - For example: change-management workflow, incident response, vendor onboarding
  - Identify start and end points; determine the high-level boundaries of the process
- Step 2: Identify the participants / roles / systems (Lanes)
  - List all the roles, departments, systems, external parties involved.
  - Decide how to organize lanes
    - For example: by department, by role, by system, or by both
  - Label each lane clearly
    - For example: “Business Unit”, “IT Security”, “Compliance”, “Operations”, “Vendor”

STEP-BY-STEP

- Step 3: Map the steps/activities
  - For each step place it in the appropriate lane, depending on who/what executes it
  - Use arrows/flows to show how tasks move from lane to lane (hand offs), or loop back
  - Include decision points or gateways where relevant
  - Annotate if a system step, manual step, or external hand off
- Step 4: Highlight hand offs and dependencies
  - Identify where flows cross from one lane to another
  - Those are hand off points
  - Identify where activities depend on another role/system completing a task before proceeding
  - Mark these in the diagram and note possible risks

STEP-BY-STEP

- Step 5: Review for role clarity and gap identification
- Once mapped, ask
  - Does every task have a clear owning lane?
  - Are there tasks that span lanes without a clear hand-over?
  - Are there “white-spaces” (steps with no lane/owner)?
  - Are there loops or back-and-forth between lanes?
    - These may indicate confusion or inefficient handoffs
- Ask also
  - Who is accountable (versus simply responsible) for each lane/task?
  - Where might someone think “it’s someone else’s job”?
  - Where might communication fail (between lanes/roles)?

STEP-BY-STEP

- Step 6: Use the diagram to spot risks
  - Look for
    - Lanes with many incoming/outgoing flows → heavy hand-off load → higher risk of delay/communication error
    - Lanes that are lightly populated (few steps) but receive many inputs or send many outputs → potential bottleneck or dependency
    - Cross-lane arrows with no clearly annotated control or check step → risk of drop
    - External lane or vendor lane handoffs (less control visibility) → elevated risk
  - For each risk spot identify:
    - What if the lane fails (person absent, system down, vendor unresponsive)?
    - What is the impact and how many other lanes depend on it?

STEP-BY-STEP

- Step 7: Communicate and maintain
  - Use the diagram in stakeholder workshops
    - It helps non-technical people like business executives visualize roles and dependencies
  - Treat it as a living artifact
    - When process or person roles change, update the diagram rather than leaving it stale
  - Link it to your risk register, controls library, or accountability frameworks
    - For example,connect a lane to a RACI chart

CONSIDERATIONS

- Risk considerations
  - Within departments often there's clearer role definition, standard operating procedures, closer supervision
    - This usually means fewer surprises but it is still possible to overlook something
  - Boundaries or hand-offs between departments/roles often create risk because
    - It may be unclear who is responsible for the next step
    - Communication may be informal or undocumented
    - Systems may change across boundaries
    - There might be delays while waiting for another team.
    - The receiving team may not have full context or may assume something is already done

RACI CHARTS

- A RACI chart (or matrix)
  - A responsibility assignment matrix (RAM) that maps tasks (or activities/decisions) against roles or stakeholders
  - It and assigns for each intersection one of the four RACI categories
    - Responsible (R)
    - Accountable (A)
    - Consulted ©
    - Informed (I).
  - In practice, the tasks are listed down the rows (left column)
  - Stakeholders/roles across the columns at the top
  - Then each cell is used to show each role's involvement in that task
  - The purpose is to ensure clarity of roles and avoid confusion, overlaps, omissions, or ambiguity in responsibility and decision-making

RACI CHARTS

Areas of focus are the activities, deliverables, departments, or channels in the content process. These vary based on the content RACI type.

	ROLES				
Area of focus	**Writer(s)**	**Translator(s)**	**Editor**	**Subject matter experts**	**Producer**
Write content	R	C	C	C	A
Translate content	C	R	C	I	A
Content accuracy	R	R	C	A	I
Content completeness	R	R	A	C	I
Content quality	R	R	C	I	A
Tone and voice	R	R	C	I	A
Digital standards	R	R	C	I	A
Final approval	C	C	C	C	A/R

Roles vary depending on the content process. One person can have several roles.

The grid identifies the type of responsibility assigned to each role.

R Responsible

The people who actually do the work. One or more people can be responsible.

A Accountable

The Approver. The person who signs off on the work. Usually, only one person is accountable.

C Consulted

The subject matter experts who are consulted and sometimes contribute to creating content.

I Informed

The people who are informed when a deliverable is complete. Communication only goes one way.

RACI ROLES

- Responsible (R):
  - The person or people who actually do the work to complete the task
  - The “doers”
  - Everyone doing the work should be identified
- Accountable (A):
  - The person who is ultimately answerable for the correct completion of the task, deliverable, or decision
  - They own the outcome, must approve/sign-off, and bear consequences for success/failure
  - Per best practice, each task should have exactly one Accountable person

RACI ROLES

- Consulted (C):
  - People or roles whose opinions are sought
  - Typically subject-matter experts, stakeholders who must give input before the task/decision is done
  - They are engaged in a two-way communication
- Informed (I):
  - People or roles who need to be kept up-to-date on progress or results
  - They are not directly involved in doing the work or inputting decisions, but they need awareness
  - One-way communication.

STEP-BY-STEP

- Step 1: Identify Activities / Tasks / Decisions
  - List the set of tasks or decision points for the process or system you are analyzing
  - For instance
    - Initiate change request
    - Review risk impact
    - Approve change
    - Implement change
    - Monitor change outcome
  - Decide on the level of granularity, you may want to limit to high-level tasks for clarity

STEP-BY-STEP

- Step 2: Identify Roles / Stakeholders
  - Identify all relevant roles (not necessarily individual names) that will be involved
  - Example
    - Business Owner
    - IT Security Lead
    - Compliance Manager
    - Change Manager
    - Operations Team
  - Capture roles rather than always names, so the chart remains valid over time

STEP-BY-STEP

- Step 3: Build the matrix structure
  - Create a table/spreadsheet with rows = tasks/activities, columns = roles
  - For each cell, assign R, A, C, or I as appropriate.
  - Ensure that there is at least one “R” per task and exactly one “A” per task
    - Consulted and Informed may have multiple assignments
- Step 4: Validate with stakeholders
  - Review the draft with relevant stakeholders to confirm the assignments
  - Lack of input from stakeholders may mean there are hidden gaps
  - Check for common problems:
    - No “A” assigned for a task
    - Too many “R” (means duplication or ambiguity)
    - Same person in multiple roles without clear justification

STEP-BY-STEP

- Step 5: Risk identification and control mapping
  - Once the RACI chart is populated, analyze for risk
    - For example: tasks where “I” stakeholders are missing could mean decision makers aren't being informed, leading to lack of control oversight
  - Link tasks that have high risk or critical control to the “A” role
    - That person then owns risk acceptance for that control
  - Assign monitoring, assurance, escalation responsibilities via “R/A”
- Step 6: Maintain and update
  - The RACI chart becomes a living document
    - When processes change, roles evolve, or new tasks are added, updates are necessary
    - Otherwise it becomes stale and misaligned
  - Make it accessible and reference in governance meetings

DIAGNOSTIC USES

- Use the RACI chart to identify potential problems with controls
  - No Accountable assigned or multiple people assigned to Accountable
    - When nobody or too many people own a task, decision-making stalls
  - Too many Responsible roles
    - Several people marked “R” for same task can create duplication, confusion and lack of clarity.
  - Excessive Consulted/Informed roles
    - Over-communication can slow processes
    - Too many “I” roles may add noise

COMMON ERRORS

- Rigid/static chart
  - If not updated, the RACI chart becomes outdated and misleading
- Too granular
  - If every small step has a RACI assignment, the matrix becomes heavy and unwieldy
  - Apply suitable level of granularity
- Misalignment with process map
  - If tasks in process map and tasks in RACI chart don’t correspond, then there is the potential for control gaps

IDENTIFYING CONTROL WEAKNESSES

- Processes don’t fail only because of technical or system flaws
  - Often the weakest link is people, communication, or organizational structure
  - Any shortcoming or gap in the design or operation of a control that increases the risk of error, fraud, or non-compliance
- Communication breakdowns often occur
  - Example: lack of timely, accurate information flowing between roles, teams, systems
  - These can turn what might be a minor issue into a major incident
  - Because necessary escalation, monitoring or remediation does not happen

KEY DIAGNOSTIC ISSUES

- Role confusion
  - When duties overlap, when it is unclear who is supposed to perform or approve a task
  - Potential issues: duplicated work, missed work, or absence of control
    - For example: Two teams think the other will perform the hand-off check.
    - As a result none performs it resulting in a control gap
  - Possible design deficiency of control: lack of clarity as to who is responsible
  - Possible operational deficiency: control exists but how to execute it is unclear

KEY DIAGNOSTIC ISSUES

- Lack of Accountability
  - Often when tasks are assigned to “everyone” or “anyone” then there is no clear owner of the outcome
  - Without an accountable person, no one drives it, no one maintains oversight, no one escalates when things go wrong
  - Absence of “Accountable” in the RACI or missing “A” for a task means that control is weak or ineffective
  - Frequently shows up in control testing as “no one signed off” or “no owner of the exception” when a failure occurs

KEY DIAGNOSTIC ISSUES

- Communication Gaps
  - Key information not shared or delayed across teams, departments or levels
    - Example: A security vulnerability is discovered but is not communicated to the business until a change request is processed which results in the hand-off failing
  - Communication gaps are especially dangerous because they often don’t show up in routine operations but when an incident or exception occurs
    - These are the situations where speed and clarity matter
  - Edge cases and outliers are often where communication breaks down
    - No one has done a walk through of rare event occurrences because they were overlooked
  - Missed exceptions are often handled by ad hoc initiatives
    - These have to be converted to SOPs and run books to update the controls

KEY DIAGNOSTIC ISSUES

- Cultural Factors
  - If the organizational culture is a “blame culture”, employees may not escalate issues, near-misses or process deviations
    - They may hide or ignore them
    - This delays remediation, meaning control weaknesses persist
  - A risk-aware culture, encourages reporting, learning, escalation
    - Without this attitude, the “people” and “process” pieces of the risk triangle weaken
  - A well-designed process will still fail if people don’t feel safe to raise issues
    - Or if they lack clarity on roles and communication protocols

STEP-BY-STEP

- Step 1
- Use process models and RACI charts as a baseline
  - These are the artifacts reviewed to check alignment
  - Are responsibilities clearly assigned?
  - Are hand-offs clear?
  - Where does communications between roles need to happen?

STEP-BY-STEP

- Step 2
- Review control points and hand offs
  - On the process map, identify every control step: approvals, reviews, checks, data validations, hand-offs
  - For each control step, ask
    - Who is performing this? Do we have a clear owner?
    - What happens if this doesn't occur? Is there a backup?
    - Are there hand-offs (cross-team, cross-system)? If yes, is the receiving party clearly defined?
  - On the RACI chart, ask
    - Are tasks with controls assigned a clear "A" (Accountable)?
    - Are the "C" (Consulted) or "I" (Informed) roles too many (information overload) or missing?
    - Where are the overlaps or omissions?

STEP-BY-STEP

- Step 3
- Survey and interview stakeholders
  - Speak with people doing the tasks; ask
    - Do you know your role clearly?
    - When you hand-off to another role/team, do you always follow the same procedures?
    - Have you ever waited on someone else to complete a task you thought you'd done?
    - Have you ever found you weren't aware of a change and you carried on in error because you weren't told?
  - Look for signals of communication breakdown
    - For example: repeated rework, tasks waiting on outside team, missing documentation, undocumented exceptions

STEP-BY-STEP

- Step 4
- Analyze control effectiveness
  - For each control step, evaluate
    - *Design*: Is the control well explained, documented, and aligned with the risk it is intended to mitigate?
    - *Operation*: Is the control executed reliably, consistently? Are there exceptions, workarounds?
  - Use frameworks that control weaknesses into design vs operational vs administrative weaknesses
  - Identify communication weaknesses
    - For example, controls that rely on hand-written log entries, manual emails with no receipt, or informal verbal escalation

STEP-BY-STEP

- Step 5
- Categorize, prioritize and document weaknesses
  - Create a table or register
  - Each row = identified weakness or communication gap
  - Columns (suggested)
    - Process step
    - Description of weakness
    - Risk impact
    - Root cause (people/process/system)
    - Proposed remediation
    - Priority
  - Typical categories
    - Role confusion, missing accountability, unclear hand-off, un-documented escalation, culture/behaviour issue
  - Prioritize on impact (how bad if fails) and likelihood (how likely based on current state)
    - Using the standard risk scoring used in the risk analysis

STEP-BY-STEP

- Step 6
- Develop mitigation and remediation strategies
  - For each identified weakness, propose targeted actions
    - Clarify or rewrite task/role descriptions; update RACI and process map
    - Assign clear Accountable roles; eliminate “everyone’s job”
    - Formalize communication/handoff protocols (who informs whom, what format, timeline)
  - Conduct training and change management to shift culture (from blame to reporting)
  - Automate where possible to reduce hand-offs and manual communication
  - Add monitoring/controls to catch missed hand-offs or uninformed stakeholders
  - Embed review mechanism:
    - The control environment must be monitored and updated regularly
    - Continuous monitoring helps surface weaknesses early

STEP-BY-STEP

- Step 7
- Monitor and sustain
  - After remediation, embed KPIs/metrics to monitor the effectiveness of the controls
    - For example: number of hand-off delays per month, number of un-approved tasks, number of escalations raised vs expected
  - Conduct periodic process walkthroughs with stakeholders
  - Review culture indicators
    - For example: number of risk/incident reports raised, near-miss reports, time to escalate
  - Use the process map + RACI + control register together as a living document
    - Revisit when process or organization changes

Q&A AND OPEN DISCUSSION