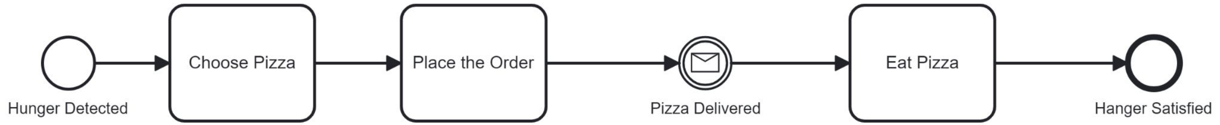
Camunda 7 Training

Day 1: BPM Fundamentals & Basic BPMN

Presented By: Hasan Ghanem | Camunda Champion
hhghanem@innovative-digital.com.sa
https://forum.camunda.io/u/hassang/summary

What is a business process?

A business process is a set of **steps** you follow, one after another, to reach a **goal.**



Why Orchestration Matters

Having many systems and tools isn’t enough. **If they aren’t connected, they only support work** — they don’t deliver real value. Tasks get delayed, information is lost, and it’s hard to see the full picture.

Process orchestration connects people, systems, and tasks from start to finish. It ensures everything happens in the right order, on time, and smoothly.

Why Orchestration Matters

- **End-to-End Control**

  You can oversee the entire process from the first step to the last, ensuring nothing is missed.

- **Full Visibility**

  Managers and stakeholders can see what's done, what's pending, and where bottlenecks occur — in real time.

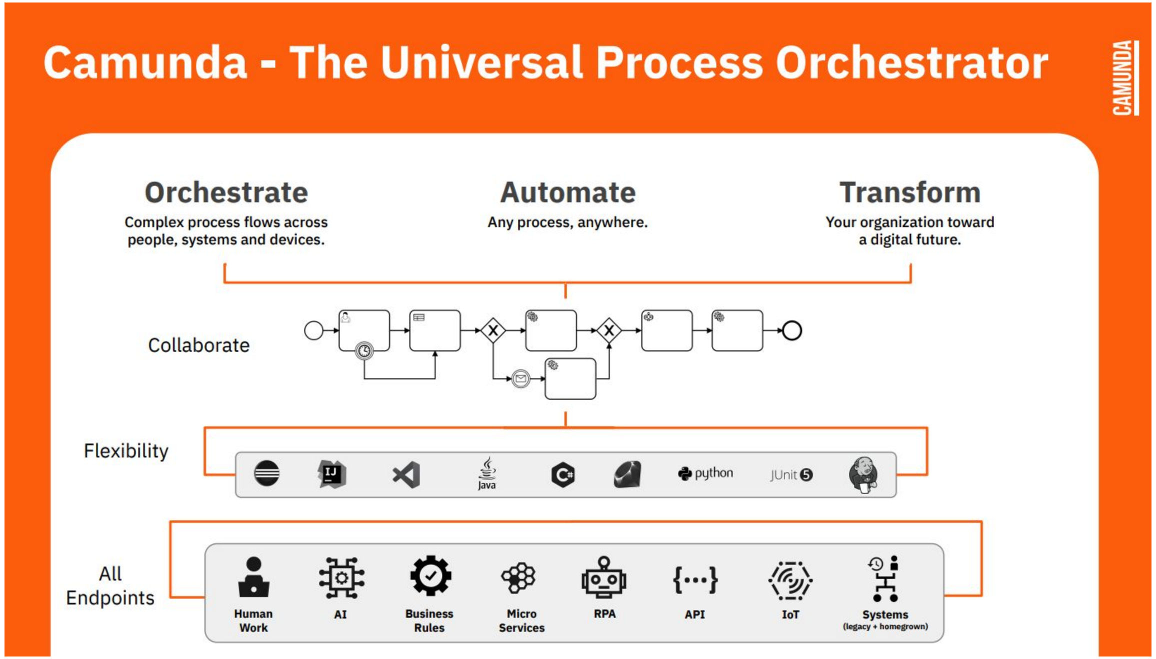
- **Maintainability**

  Instead of messy, manual connections and quick fixes, the orchestration layer becomes the single, maintainable map of your business process.

- **Better Collaboration**

  Orchestration coordinates not just software systems but also people, ensuring tasks and approvals flow smoothly between teams.

Why Orchestration Matters

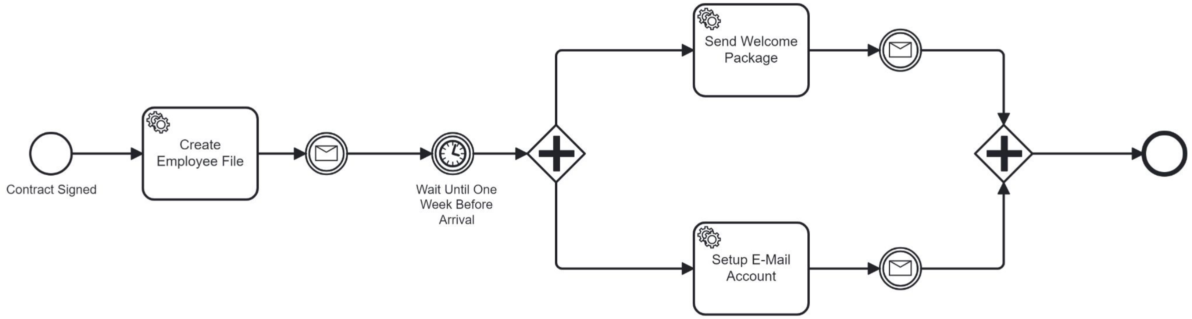

Camunda - The Universal Process Orchestrator
CAMUNDA
Orchestrate
Complex process flows across people, systems and devices.
Automate
Any process, anywhere.
Transform
Your organization toward a digital future.
Collaborate
Flexibility
Java
python
JUnit 5
All Endpoints
Human Work
AI
Business Rules
Micro Services
RPA
API
IoT
Systems (legacy + homegrown)

Why Orchestration Matters

Example: Controlling Manual Tasks with Camunda and Trello

Onboarding Process

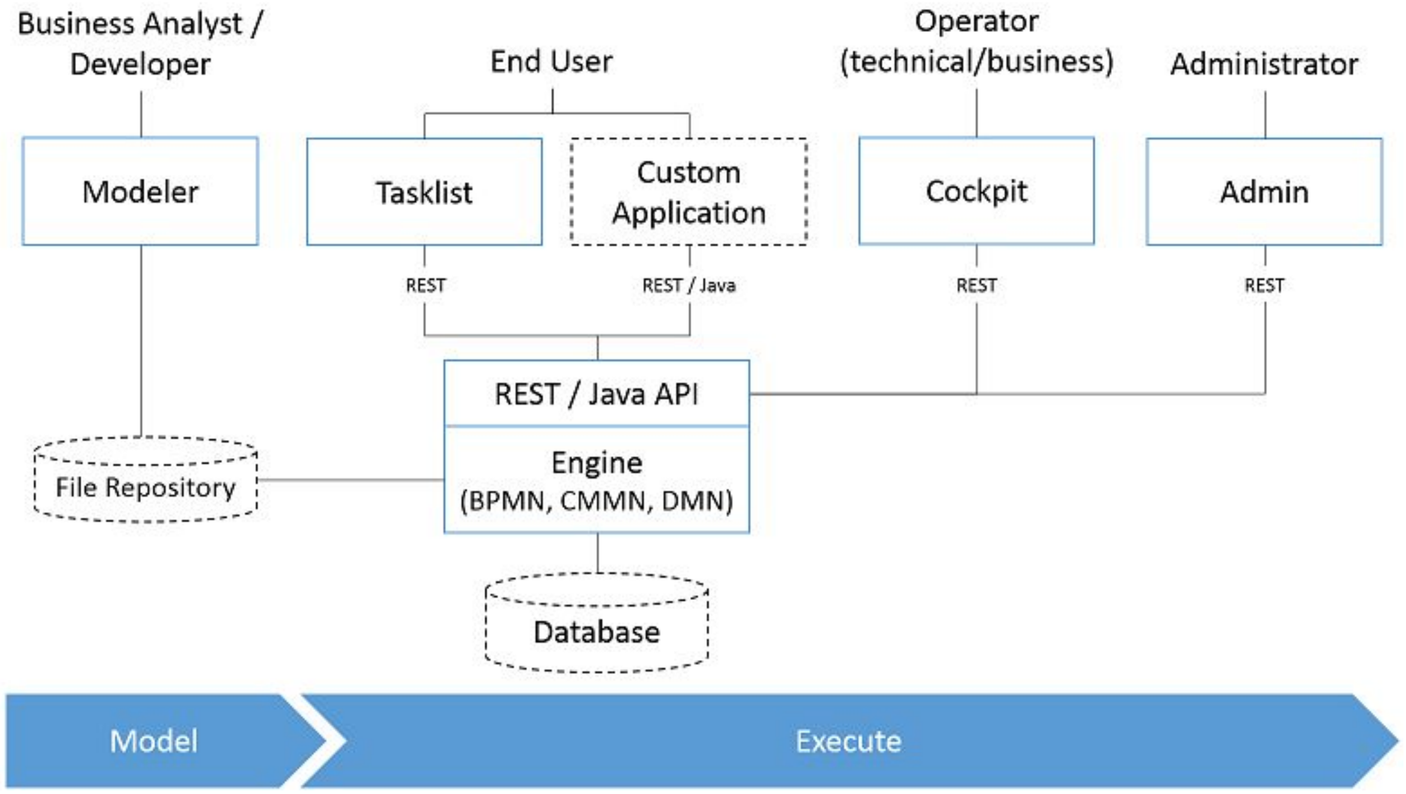
https://camunda.com/blog/2021/02/automate-your-manual-tasks-with-camunda-and-trello/



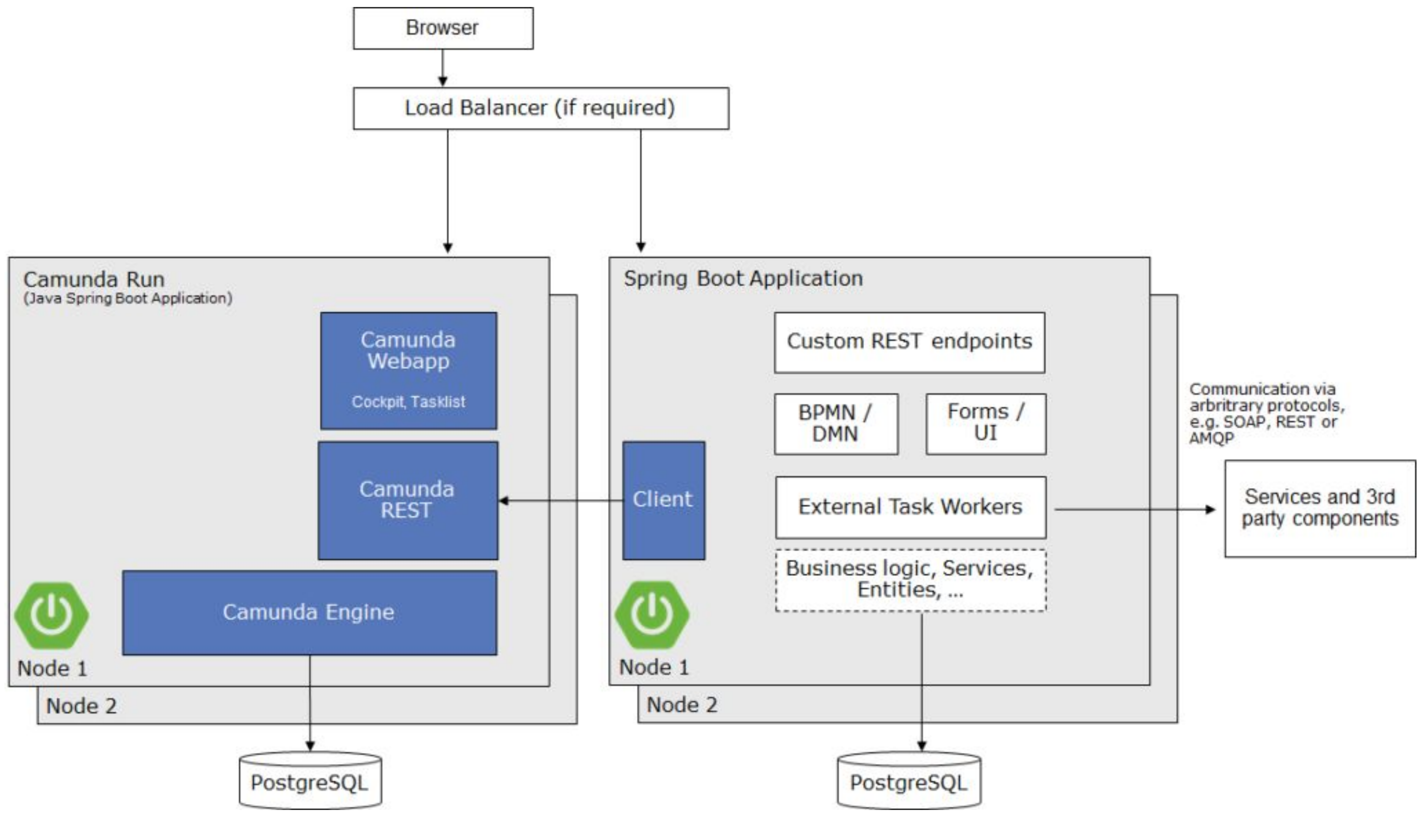
Camunda Platform 7 is a business process automation (BPA) platform that allows users to **design, automate, and manage** business processes. It focuses on workflow and decision automation, offering tools for **modeling, execution, and optimization**.

Camunda 7 provides engines for BPMN workflows and DMN decisions, along with solutions for modeling, operations, and analytics.

Camunda 7 Components



Camunda 7: Recommended Deployment (Remote Engine)



Benefits of a Remote Engine

- **Decoupling:** The workflow engine is provisioned and configured independently of the application and process solution. Problems can be easily located on one of the components, and vulnerabilities are not transcending into other components.
- **Improved scaling patterns:** The workflow engine can be scaled independently of the application code. Camunda can optimize the performance of the core engine, as it has full control of what is running in this scope.
- **Allow software as a service (SaaS):** The workflow engine can be operated as a service for you, either in a public cloud (like Camunda Platform 8) or probably as an in-house service (as customers of ours do). Still, you can develop your application locally or on-premise, as applications can remotely connect to the workflow engine.
- **Easier getting started experience:** You can provision an engine by a simple Docker command and don't need to mess with configurations in your own application.

https://camunda.com/blog/2022/02/moving-from-embedded-to-remote-workflow-engines/

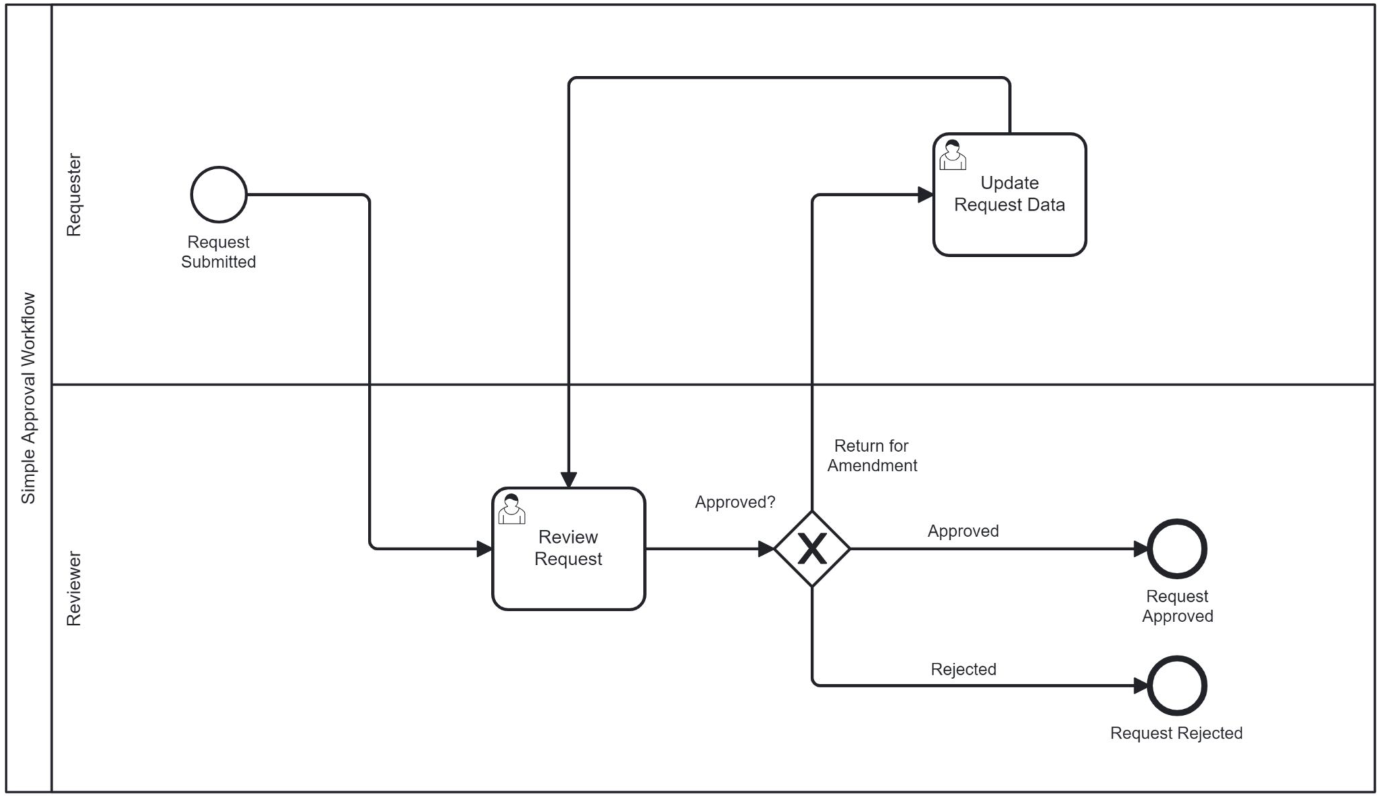
BPMN Basics: Modeling Workflows

In general, certain tasks have to be carried out during a process (**activities**), perhaps under certain conditions (**gateways**), and things may happen (**events**). What connects these **three flow objects** are **sequence flows**, but only within a **pool**. If connections cross pool boundaries, the process resorts to **message flows**.

Source: Real-Life BPMN (4th edition)

https://page.camunda.com/wp-bpmn-2-0-business-process-model-and-notation-en
https://docs.camunda.org/manual/latest/reference/bpmn20/

Create, Deploy and Test a Simple Workflow



Hands-On

Model a simple Leave Request process with:

- Start ➔ User Task ➔ Manager Approval ➔ End

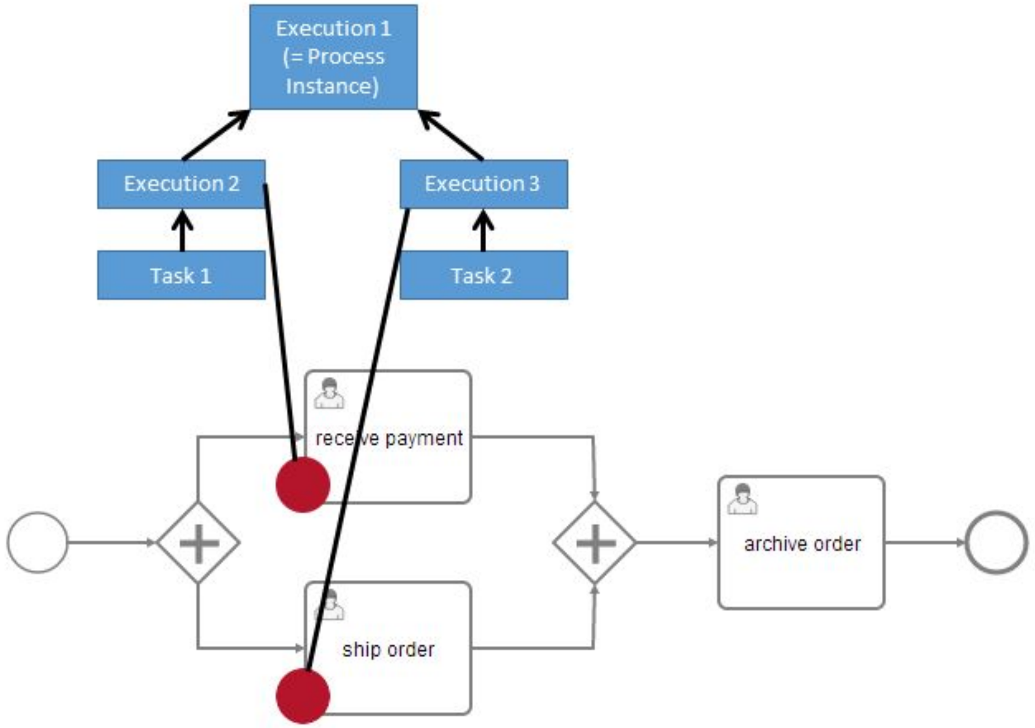
Deploy and run on Camunda Tasklist

Camunda 7 Training

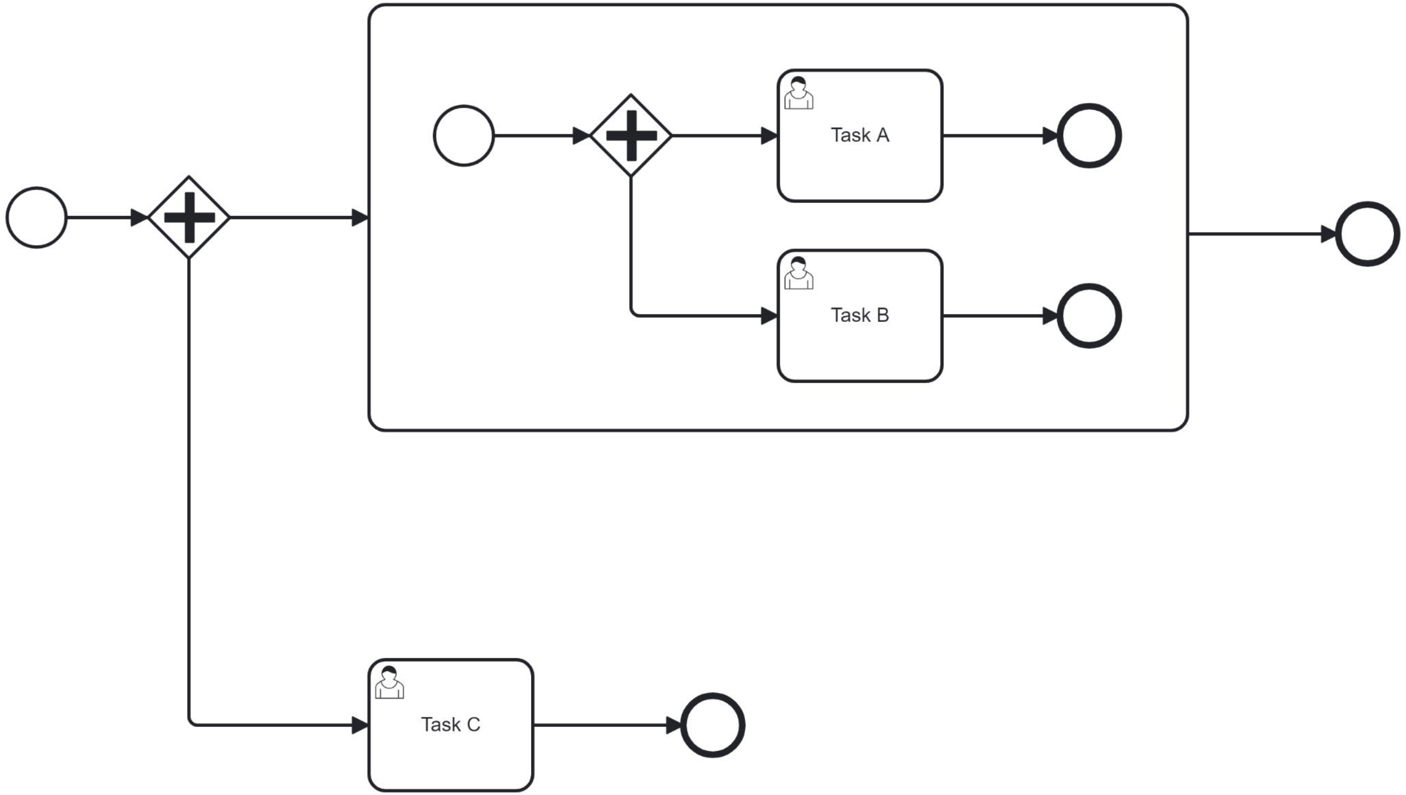
Day 2: Variables, User Tasks & External Task Intro

Presented By: Hasan Ghanem | Camunda Champion
hhghanem@innovative-digital.com.sa
https://forum.camunda.io/u/hassang/summary

Variable Scopes and Variable Visibility



Variable Scopes and Variable Visibility



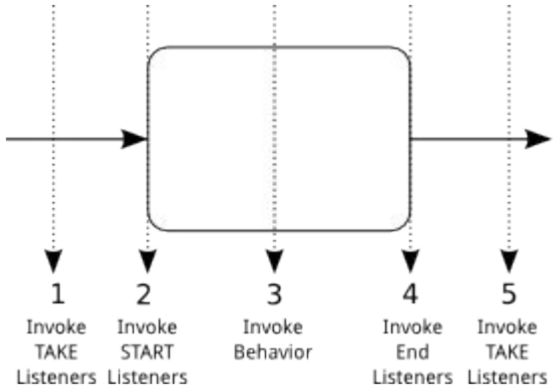
Process variables (global, local, transient)

- Setting variables to specific scope
- Transient variables: They are not saved into the database and exist only during the **current transaction**. Every waiting state during an execution of a process instance leads to the loss of all transient variables

https://docs.camunda.org/manual/latest/user-guide/process-engine/variables/#setting-variables-to-specific-scope
https://docs.camunda.org/manual/latest/user-guide/process-engine/variables/#transient-variables

Execution Listeners

- “**TAKE**” listeners are invoked on the **sequence flow** entering the activity.
- “**START**” listeners are invoked on the **start** of the **activity**.
- “**END**” listeners are invoked on the **end** of the **activity**.



Execution Listeners

Execution listeners allow you to execute external Java code or evaluate an expression when certain events occur during process execution. The **events** that can be captured are:

- Start and end of a process instance.
- Taking a transition.
- Start and end of an activity.
- Start and end of a gateway.
- Start and end of intermediate events.
- Ending a start event or starting an end event.

https://docs.camunda.org/manual/latest/user-guide/process-engine/delegation-code/#execution-listener

Execution Listeners

Execution listeners and service tasks in Camunda serve different roles:

1. **Execution listeners** handle general tasks like logging, metrics, or resource cleanup. They’re great for small, reusable actions that don’t affect the process flow and keep business logic clean.
2. **Service tasks** are for key process actions, such as external service calls or complex business logic, and appear in the BPMN diagram to make the process clear.

Use **execution listeners** for lightweight operations, cross-cutting tasks, or consistency across multiple processes. Use **service tasks** for major process steps, long-running tasks, or when error handling and retries are needed.

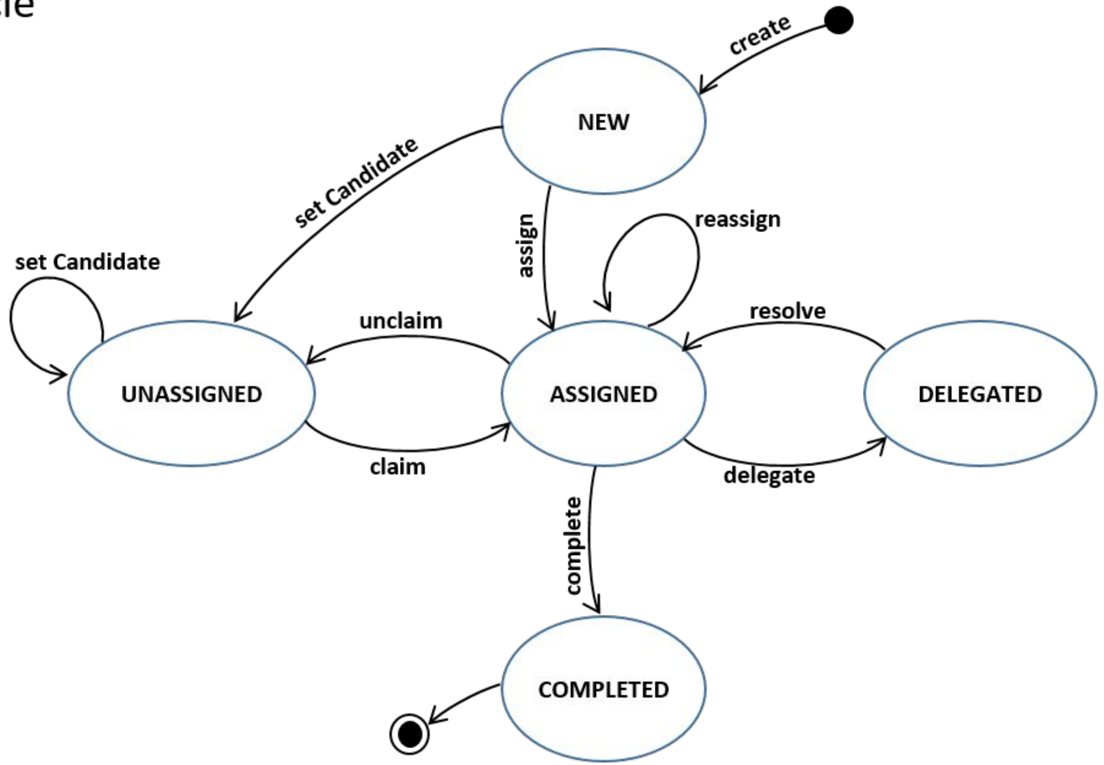
Execution Listeners

Some **use cases** could be:

- Generate a UID as soon as the process get started
- Update process status in external database as soon as the process instance is completed

User Task Lifecycle

Task Lifecycle



Task Lifecycle States

- New
- Unclaimed
- Claimed/Assigned
- Delegated
- Completed
- Deleted

Task Lifecycle Operations

- Create
- Set Candidate
- Claim
- Unclaim
- Assign
- Reassign
- Delegate
- Resolve
- Complete
- Delete
- Suspend

Task Listeners

A task listener is used to execute custom Java logic or an expression upon the occurrence of a certain task-related event (create, update, assignment, timeout, complete and delete)

https://docs.camunda.org/manual/latest/user-guide/process-engine/delegation-code/#task-listener-event-lifecycle
https://docs.camunda.org/manual/latest/user-guide/process-engine/delegation-code/#task-listener

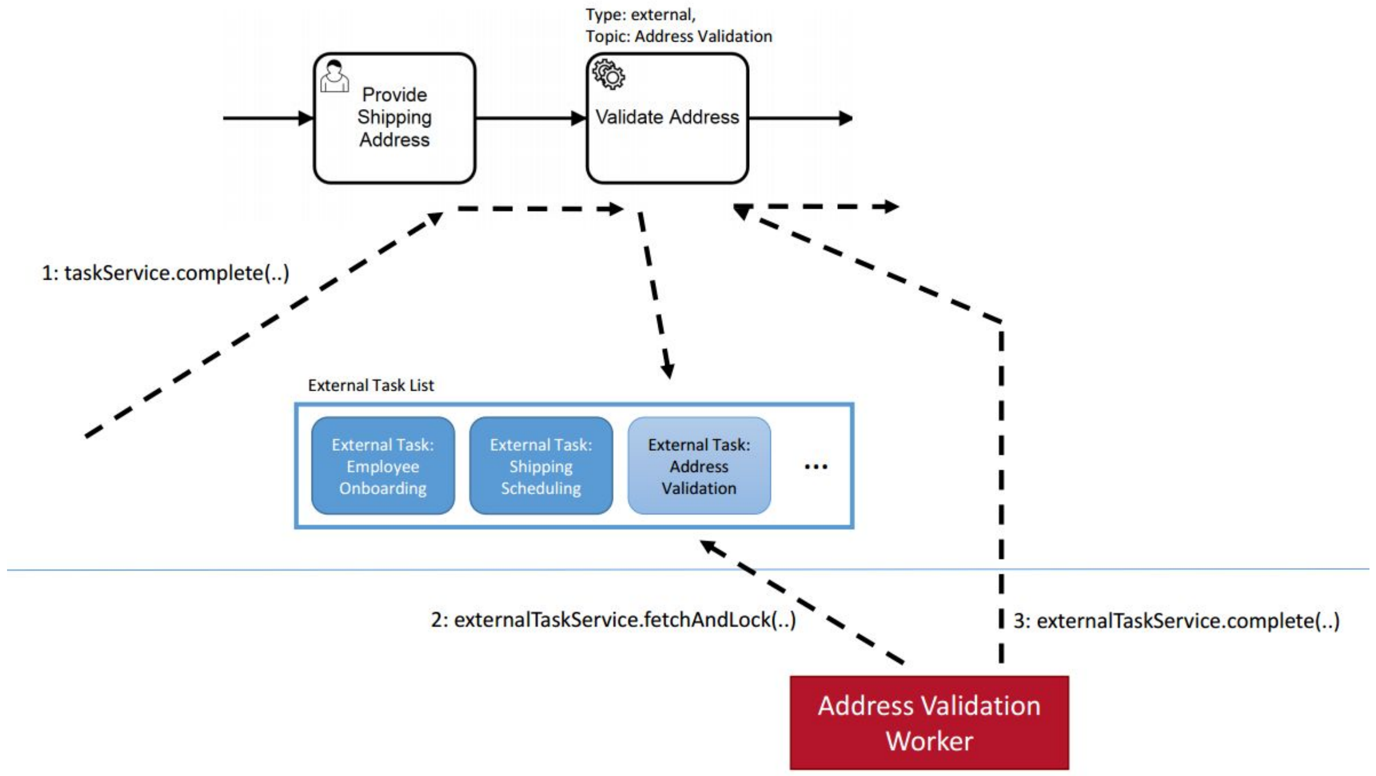
Task Listeners

Some **use cases** could be:

- Send an email/Slack notification to the assignee’s manager that a new approval task has been created (**create**).
- Send email notification to the assignee with task details and a link to complete it (**assignment**).
- Escalate to a supervisor via email or notification when a task reaches its due date without completion (**timeout**).
- Clean up related records in external systems (**delete**).

https://docs.camunda.io/docs/components/best-practices/architecture/extending-human-task-management-c7/#implementing-a-custom-processtask-info-entity

External Task Pattern


Type: external,
Topic: Address Validation
Provide Shipping Address
Validate Address
1: taskService.complete(..)
External Task List
External Task: Employee Onboarding
External Task: Shipping Scheduling
External Task: Address Validation
...
2: externalTaskService.fetchAndLock(..)
3: externalTaskService.complete(..)
Address Validation Worker

External Task Pattern

External tasks let an external worker or application execute process work instead of Camunda's job executor. This provides several advantages:

- **Full control over execution** – You decide how and when tasks run.
- **Scalability** – Easily scale workers or threads up and down to match workload.
- **Efficient handling of heavy work** – Complex logic or expensive network calls can be handled externally without blocking Camunda threads.
- **Reliable state management** – Camunda waits in a persistent state until workers pick up the task, ensuring no work is lost.
- **Decoupling from the engine** – The engine no longer calls your business logic directly; instead, workers pull tasks via the API, giving you flexibility in deployment and load balancing.

Hands-On

Extend the Leave Request process with:

- Form fields
- Task listeners for logging
- Variable mappings between tasks

Camunda 7 Training

Day 3: Implementing External Task Workers and Integrating Camunda APIs with .NET

Presented By: Hasan Ghanem | Camunda Champion
hhghanem@innovative-digital.com.sa
https://forum.camunda.io/u/hassang/summary

External Task Handlers

- Completing Tasks
- Extending the Lock Duration of Tasks
- Unlocking Tasks
- Reporting Task Failure & Specifying a Retry Strategy
- Reporting BPMN Error

https://docs.camunda.org/manual/latest/user-guide/process-engine/external-tasks/
https://docs.camunda.org/manual/latest/user-guide/ext-client/

External Task Throughput

For a high throughput of external tasks, you should balance between the number of external task instances, the number of clients and the duration of handling the work.

https://docs.camunda.org/manual/latest/user-guide/ext-client/#external-task-throughput

Developing External Task Workers

NuGet Package: Camunda.Worker

Purpose:

Fetch, lock, and complete external tasks for specific topics in Camunda 7.

https://www.nuget.org/packages/Camunda.Worker

Interacting with Camunda 7 REST APIs

NuGet Package: Camunda.Api.Client

Purpose:

Start processes, query tasks, complete tasks, and manage variables in Camunda 7 via REST API.

https://www.nuget.org/packages/Camunda.Api.Client

Hands-On

Build a C# Worker that:

- Processes a "Send Notification" task
- Handles BPMN errors and retries
- Logs output to console

Camunda 7 Training

Day 4: Call Activities, Events, and Timers

Presented By: Hasan Ghanem | Camunda Champion
hhghanem@innovative-digital.com.sa
https://forum.camunda.io/u/hassang/summary

Call Activity vs Embedded Subprocess

From a conceptual point of view, both will call a subprocess when process execution arrives at the activity.

The difference is that the **call activity** references a process that is external to the process definition, whereas the **embedded subprocess** is embedded within the original process definition.

https://docs.camunda.org/manual/latest/reference/bpmn20/subprocesses/call-activity/
https://docs.camunda.org/manual/latest/reference/bpmn20/subprocesses/embedded-subprocess/

Call Activity

Reusable: Can be used by multiple parent processes.

Version control: Can choose “latest” version or a fixed version of the called process.

Loose coupling: The sub-process can be modified independently, without changing the parent process definition.

Reusable approval flow — e.g., a “Manager Approval” process used in multiple processes.

Embedded Subprocess

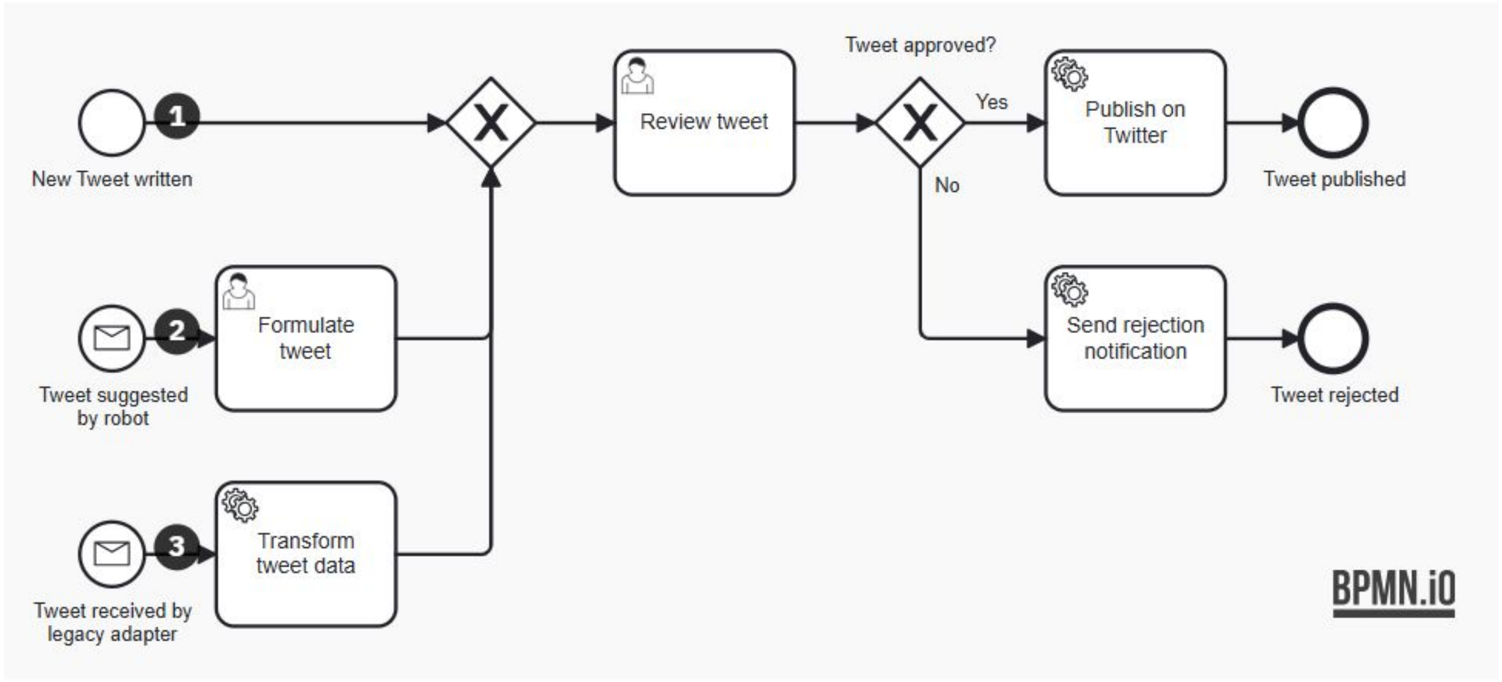
Tightly coupled: Changes require redeploying the main process.

Variable scope: Variables inside remain accessible unless scoped locally.

Good for grouping: Especially when certain tasks need to handle boundary events.

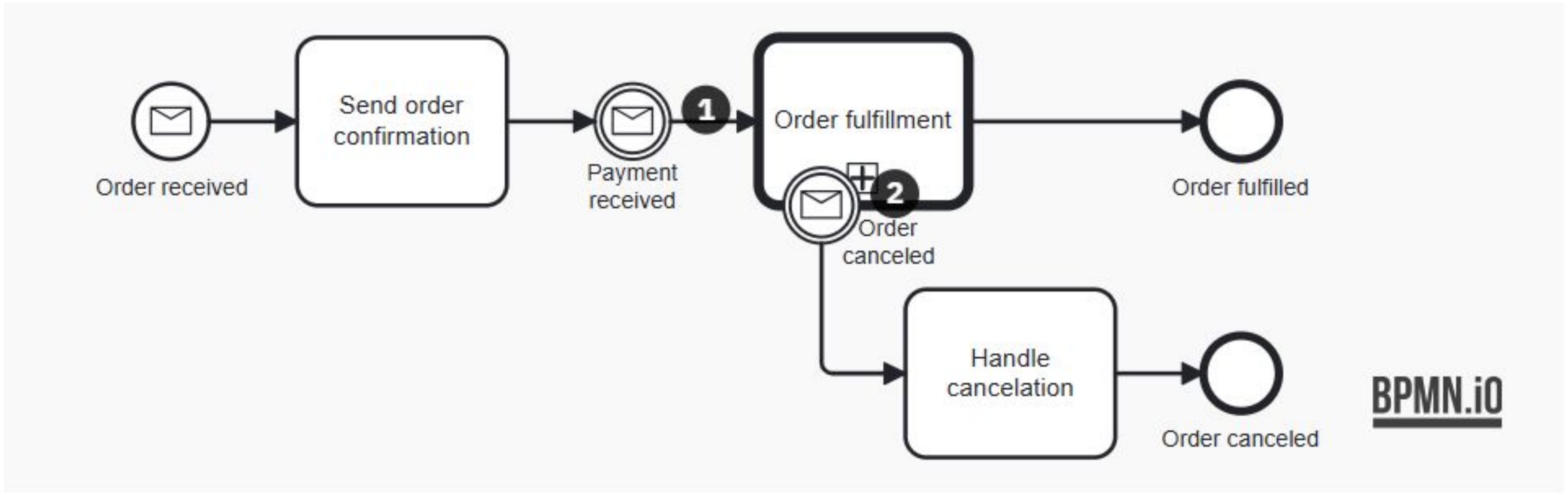
Error boundary handling — e.g., wrap a group of tasks in a sub-process to catch specific errors.

Message Events: Start Events



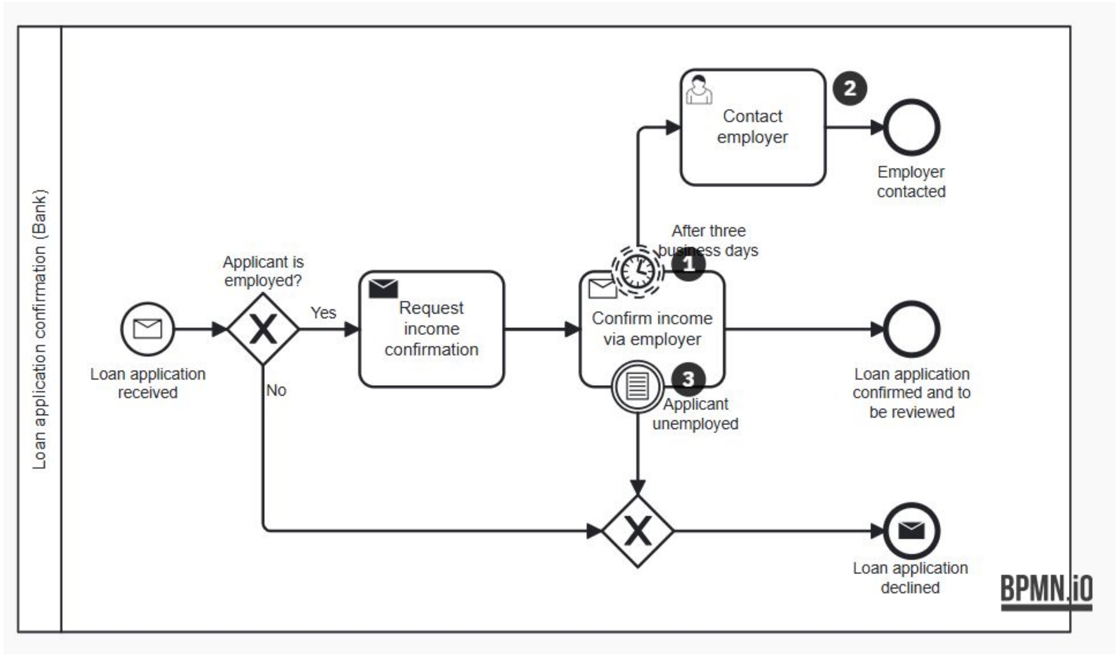
https://docs.camunda.io/docs/components/best-practices/development/routing-events-to-processes-c7/#start-events

Message Events: Intermediate Events



https://docs.camunda.io/docs/components/best-practices/development/routing-events-to-processes-c7/#intermediate-events

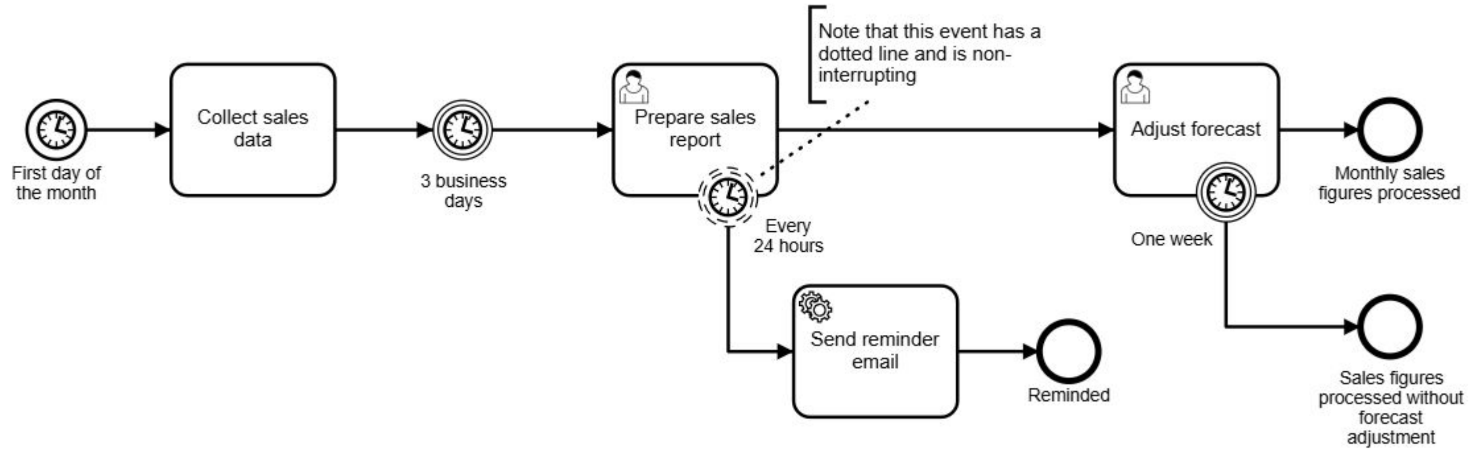
Message Events: Boundary Events


Loan application confirmation (Bank)
Loan application received
Applicant is employed?
Yes
No
Request income confirmation
Confirm income via employer
After three business days
1
Contact employer
2
Employer contacted
3
Applicant unemployed
Loan application confirmed and to be reviewed
Loan application declined
BPMN.io

https://docs.camunda.io/docs/components/best-practices/development/routing-events-to-processes-c7/#reacting-to-process-internal-events

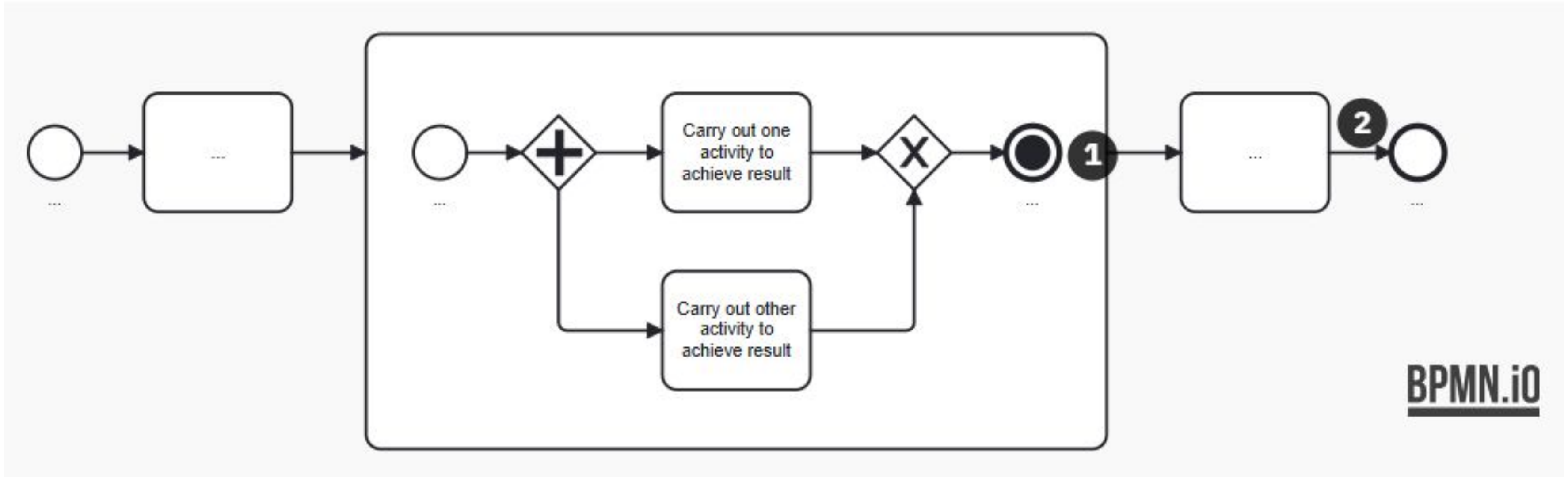
Timer Events

Timer events are events which are triggered by a defined timer. They can be used as start event, intermediate event or boundary event. Boundary events can be interrupting or not.
https://docs.camunda.org/manual/latest/reference/bpmn20/events/timer-events/


First day of the month
Collect sales data
3 business days
Prepare sales report
Note that this event has a dotted line and is non-interrupting
Every 24 hours
Send reminder email
Reminded
Adjust forecast
One week
Monthly sales figures processed
Sales figures processed without forecast adjustment

Terminate End Event

A **terminate** event ends the complete scope it is raised in and all contained inner scopes.
https://docs.camunda.io/docs/components/best-practices/modeling/building-flexibility-into-bpmn-models/#terminate-end-events


...
...
...
Carry out one activity to achieve result
Carry out other activity to achieve result
...
1
...
2
...
BPMN.io

Hands-On

- Build parent-child process using Call Activity
- Add a boundary message to interrupt a task
- Add timer to auto-escalate after 1 minute

Camunda 7 Training

Day 5: Exceptions, Compensation & Job Executor

Presented By: Hasan Ghanem | Camunda Champion
hhghanem@innovative-digital.com.sa
https://forum.camunda.io/u/hassang/summary

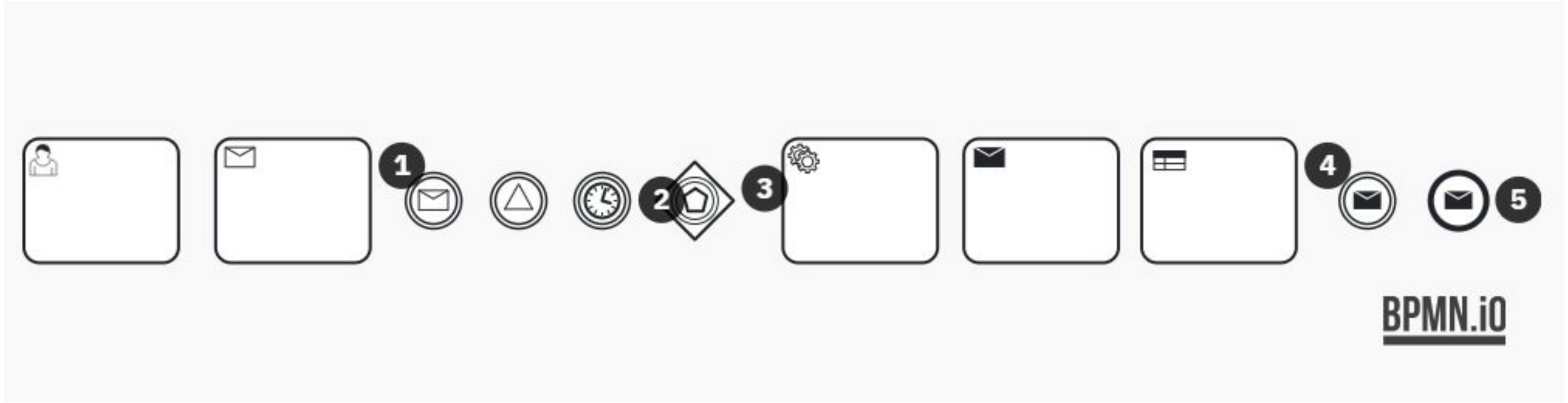
BPMN Error Events vs Exceptions

Does the error have some **business meaning** and causes an alternative process flow (like “**item not on stock**”) or is it a **technical malfunction** (like “**network currently down**”)?

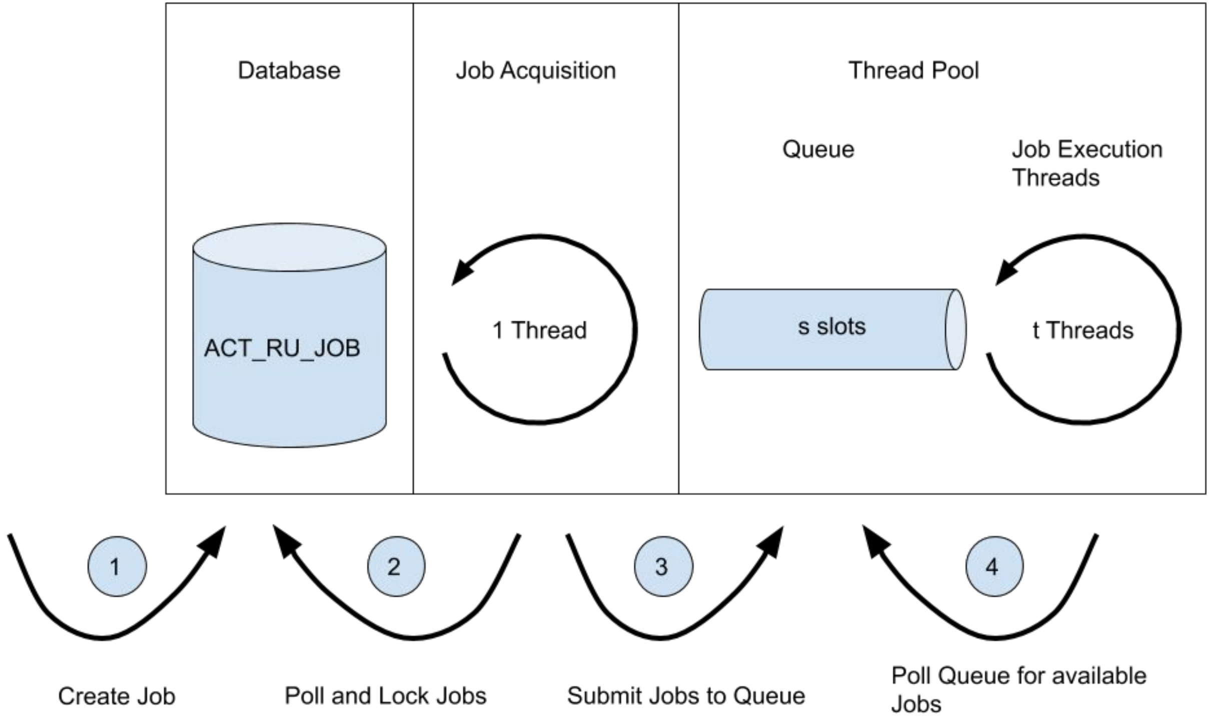
https://docs.camunda.org/manual/latest/user-guide/process-engine/error-handling/

Understanding Transaction Handling

Every time we use the Camunda 7 API to ask the workflow engine to do something (like e.g. starting a process, completing a task, signaling an execution), the engine will advance in the process until it reaches **wait states** on each active path of execution, which can be:
https://docs.camunda.io/docs/components/best-practices/development/understanding-transaction-handling-c7/



Job Executor Explained



https://camunda.com/blog/2019/10/job-executor-what-is-going-on-in-my-process-engine/

Compensation

We execute tasks in our processes that sometimes have to be **canceled** later under certain conditions.

Typical examples are:

- Booking a train or airline ticket.
- Reserving a rental car.
- Charging a credit card.

https://docs.camunda.org/manual/latest/reference/bpmn20/events/cancel-and-compensation-events/

Hands-On

- Model an Order Process:
    - Includes cancel path with compensation
    - Throws BPMN error on validation failure
- Use asyncBefore to define transaction boundary
- Simulate stuck job and analyze via Cockpit

Camunda 7 Training

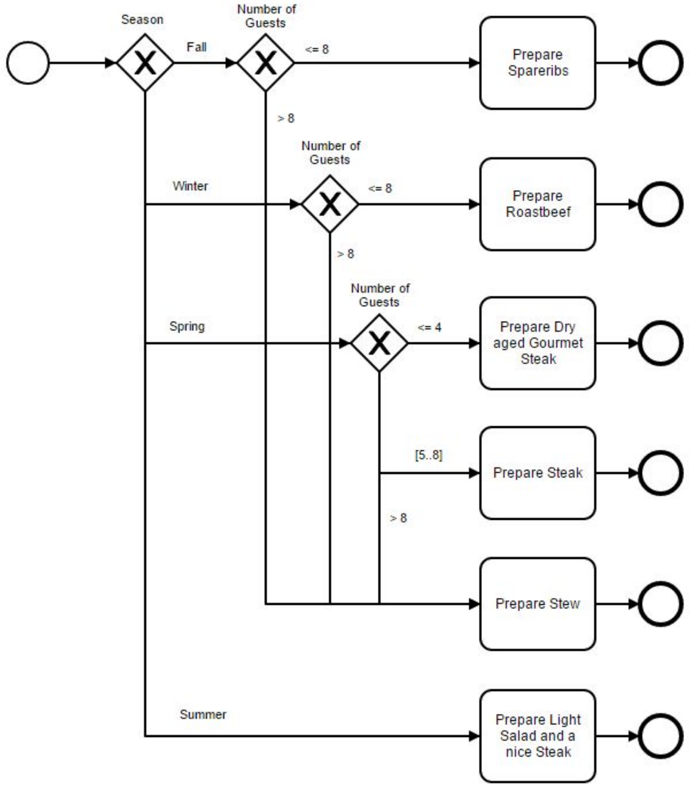
Day 6: DMN, Best Practices & Advanced Modeling

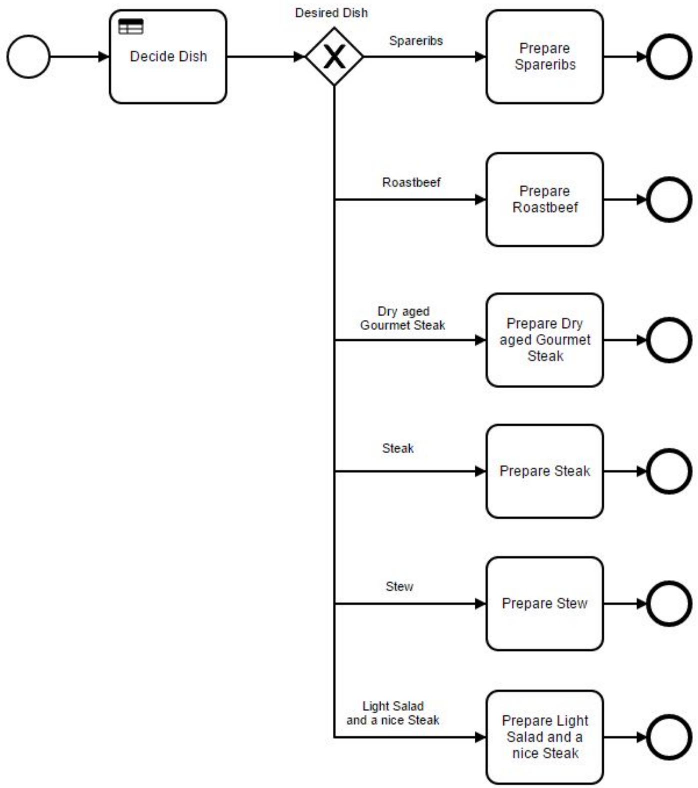
Presented By: Hasan Ghanem | Camunda Champion
hhghanem@innovative-digital.com.sa
https://forum.camunda.io/u/hassang/summary

What is DMN?

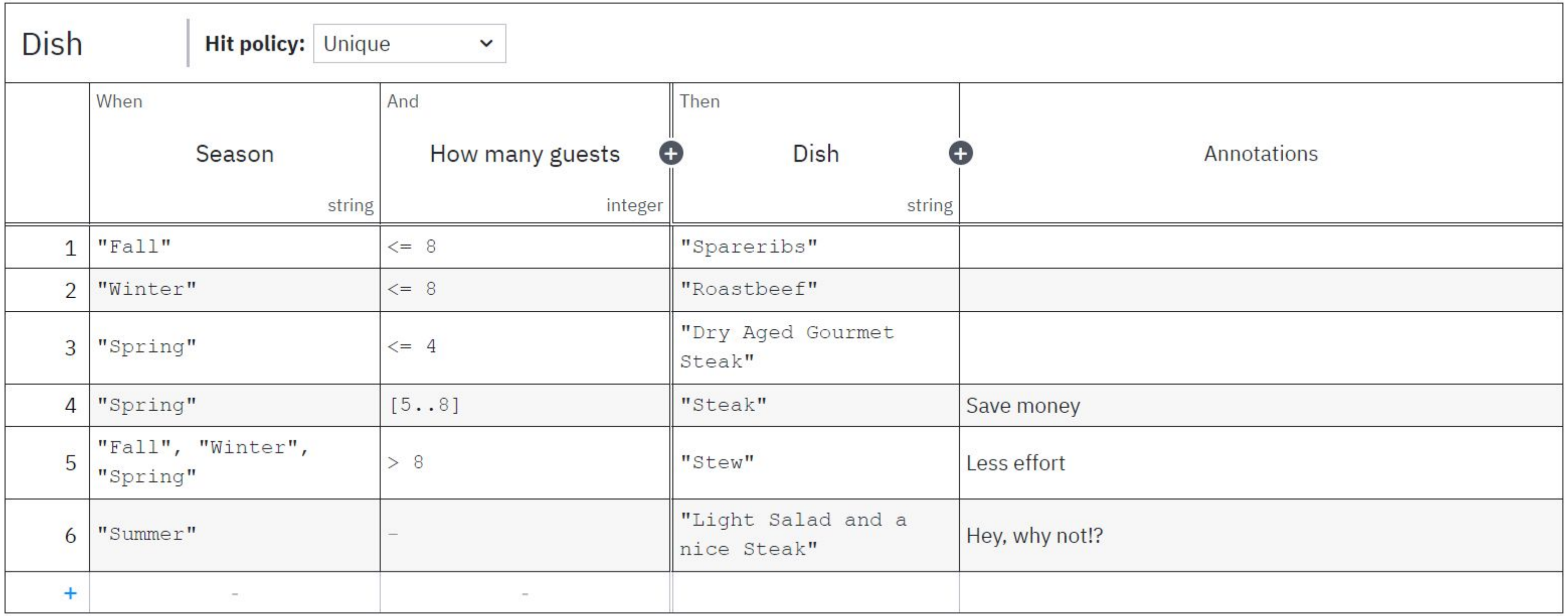
Decision Model and Notation (DMN) is a modeling approach owned by an institution called the Object Management Group (OMG), which also operates worldwide standards for BPMN. In DMN, decisions are modeled and executed using a language both business analysts and developers can understand.

Without DMN vs With DMN





Decide Dish DMN Table



Dish | Hit policy: Unique

	When	And	Then	
	Season	How many guests	Dish	Annotations
	string	integer	string	
1	"Fall"	<= 8	"Spareribs"	
2	"Winter"	<= 8	"Roastbeef"	
3	"Spring"	<= 4	"Dry Aged Gourmet Steak"	
4	"Spring"	[5..8]	"Steak"	Save money
5	"Fall", "Winter", "Spring"	> 8	"Stew"	Less effort
6	"Summer"	-	"Light Salad and a nice Steak"	Hey, why not!?
+	-	-		

https://docs.camunda.io/docs/components/best-practices/modeling/choosing-the-dmn-hit-policy/

https://docs.camunda.org/manual/latest/reference/dmn/decision-table/

Modeling Best Practices

- Creating readable process models
- Naming BPMN elements
- Modeling beyond the happy path

https://docs.camunda.io/docs/components/best-practices/modeling/creating-readable-process-models/
https://docs.camunda.io/docs/components/best-practices/modeling/naming-bpmn-elements/
https://docs.camunda.io/docs/components/best-practices/modeling/modeling-beyond-the-happy-path/

Events Subprocesses

Dealing with issues occurring almost anywhere:

Some issues can occur almost anywhere on the way through our process. The event subprocess allows us to fork off a problem path modeled separately from our main process to deal with such issues.

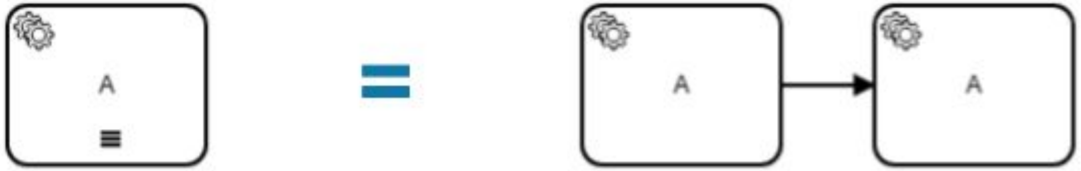
https://docs.camunda.io/docs/components/best-practices/modeling/modeling-beyond-the-happy-path/#dealing-with-issues-occurring-almost-anywhere

Multi-Instance

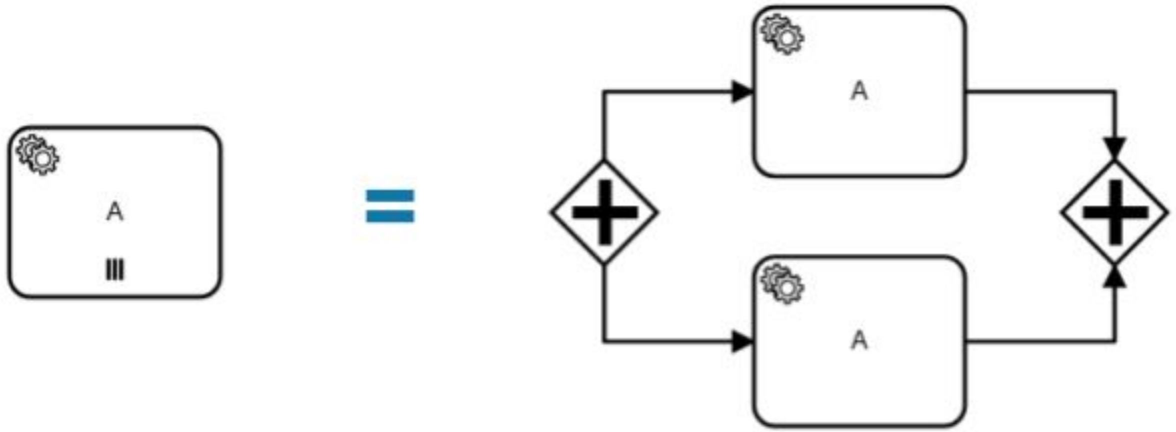
A multi-instance activity is executed multiple times - once for each element of a given collection.

A multi-instance activity is executed either **sequentially** or in **parallel** (default). In the BPMN, a sequential multi-instance activity is displayed with three horizontal lines at the bottom. A parallel multi-instance activity is represented by three vertical lines.

Multi-Instance (Sequential)



Multi-Instance (Parallel)



Hands-On

Create a DMN for Leave Approval:

- Inputs: Role, Days, Urgency
- Result: Auto-approval or escalation

Use DMN in process and visualize decisions

Camunda 7 Training

Day 7: Final Project & Camunda 8 Intro

Presented By: Hasan Ghanem | Camunda Champion
hhghanem@innovative-digital.com.sa
https://forum.camunda.io/u/hassang/summary

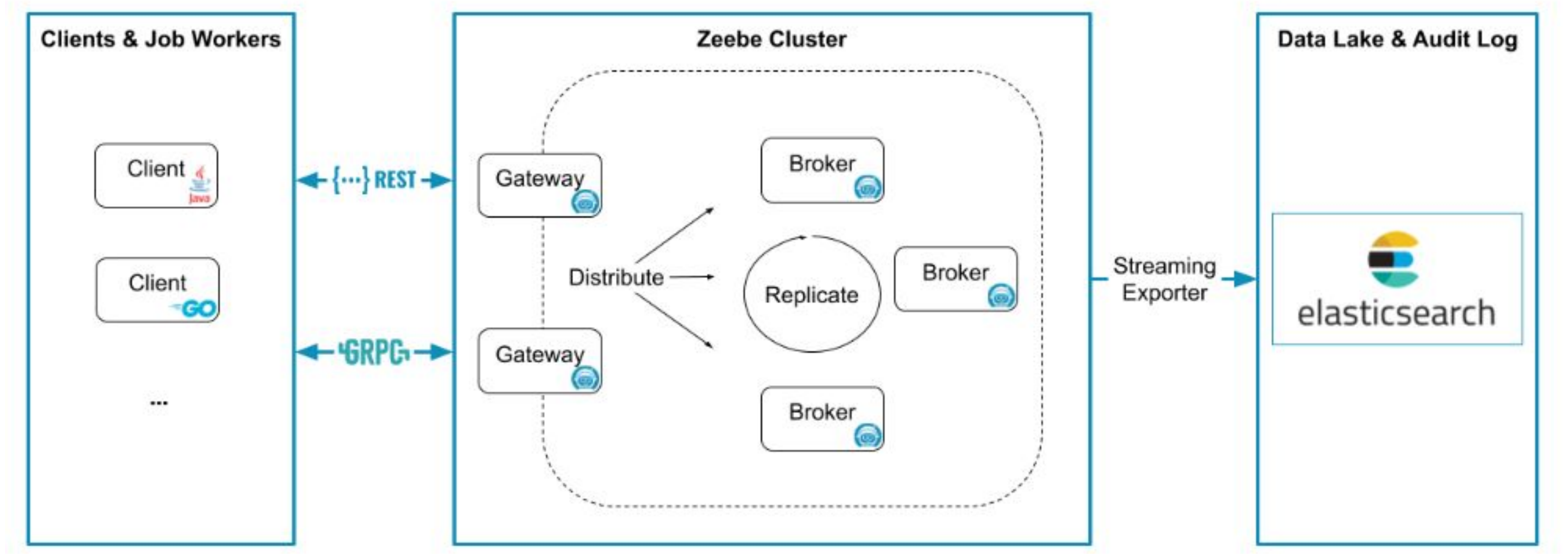
Camunda 8 Self-Managed (Local development environment)

https://signup.camunda.com/self-managed-download

Camunda 8 is a cloud-native, stateless workflow engine based on Zeebe. It is designed for distributed systems and high scalability, using event sourcing and streaming for data persistence.

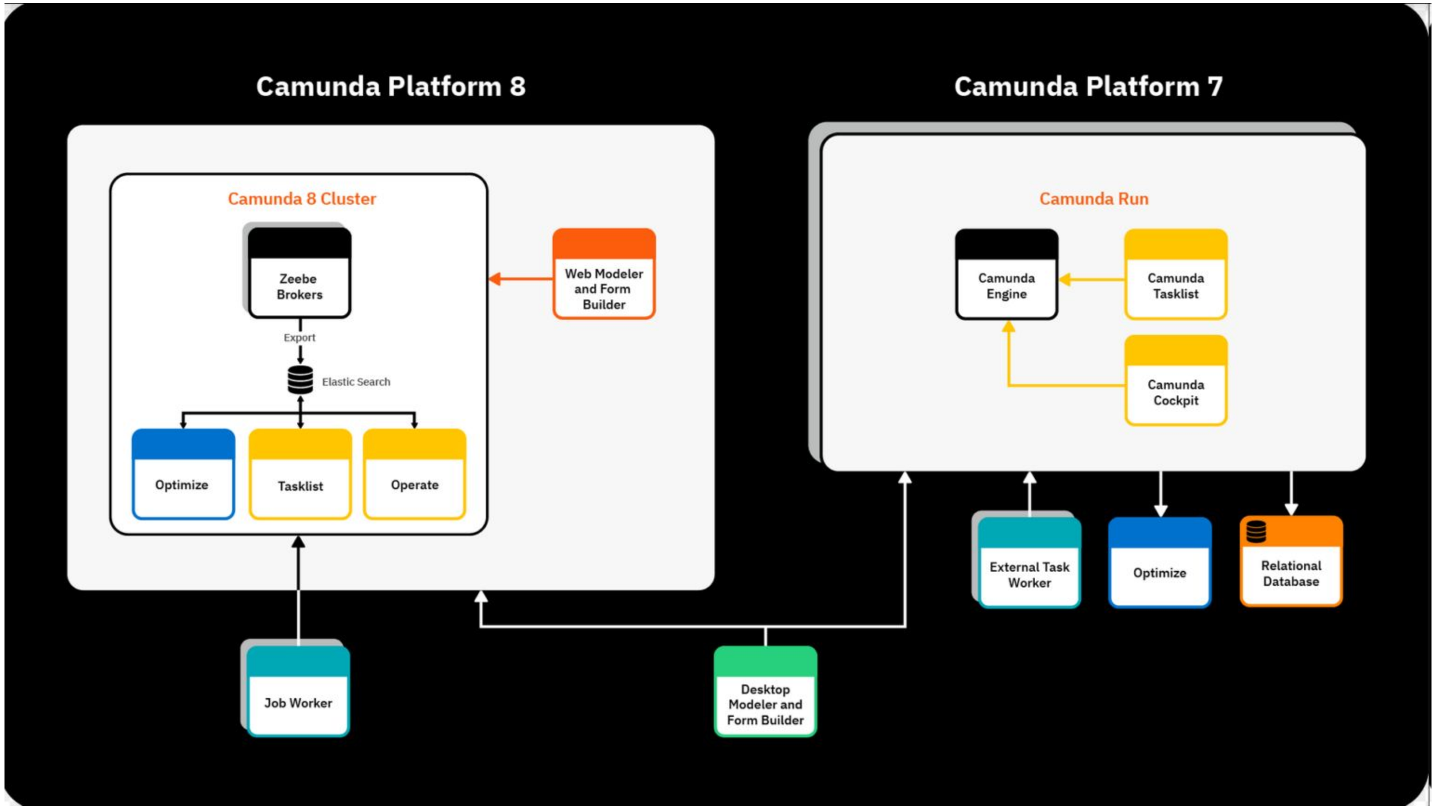
The primary difference between centralized and distributed systems is the communication pattern between the system’s nodes. The state of a centralized system is contained within a central node. Nodes of a centralized system all access the central node. A centralized system has a single point of failure while a distributed system has no single point of failure.

Architecture



https://docs.camunda.io/docs/components/zeebe/technical-concepts/architecture

Key differences between Camunda 7 & 8



Key differences between Camunda 7 & 8

In Camunda Platform 7, the clustering model was to run multiple workflow engine instances and point them to **the same relational database**. Then, you could add a load balancer upfront to distribute incoming traffic between the nodes. **The database was the single point of failure** and could become the bottleneck in this architecture.

The Zeebe engine has clustering built-in, and the default recommendation is to always run at least three brokers (a broker is similar to a Camunda Platform 7 workflow engine node) for resilience, as this **eliminates any single point of failure**.

https://camunda.com/blog/2022/04/camunda-platform-8-for-camunda-platform-7-users-what-you-need-to-know/
https://docs.camunda.io/docs/guides/migrating-from-camunda-7/conceptual-differences/

Hands-On

- Model a multi-stage process with:
    - User tasks
    - External tasks
    - Message events
    - DMN decision
- Present results to group

Camunda 7 Training

Day 8: Migration Hands-On Workshop

Presented By: Hasan Ghanem | Camunda Champion
hhghanem@innovative-digital.com.sa
https://forum.camunda.io/u/hassang/summary

Camunda 7 to 8 Migration Example

https://github.com/camunda-community-hub/camunda-7-to-8-migration-example

https://docs.camunda.io/docs/guides/migrating-from-camunda-7/

Hands-On

Migrate a selected Camunda 7 BPMN process to Camunda 8

Test and validate the migrated process on Camunda 8