ModelOps

Technical Masterclass

ModelOp

Session #4: Designing a Model Lifecycle

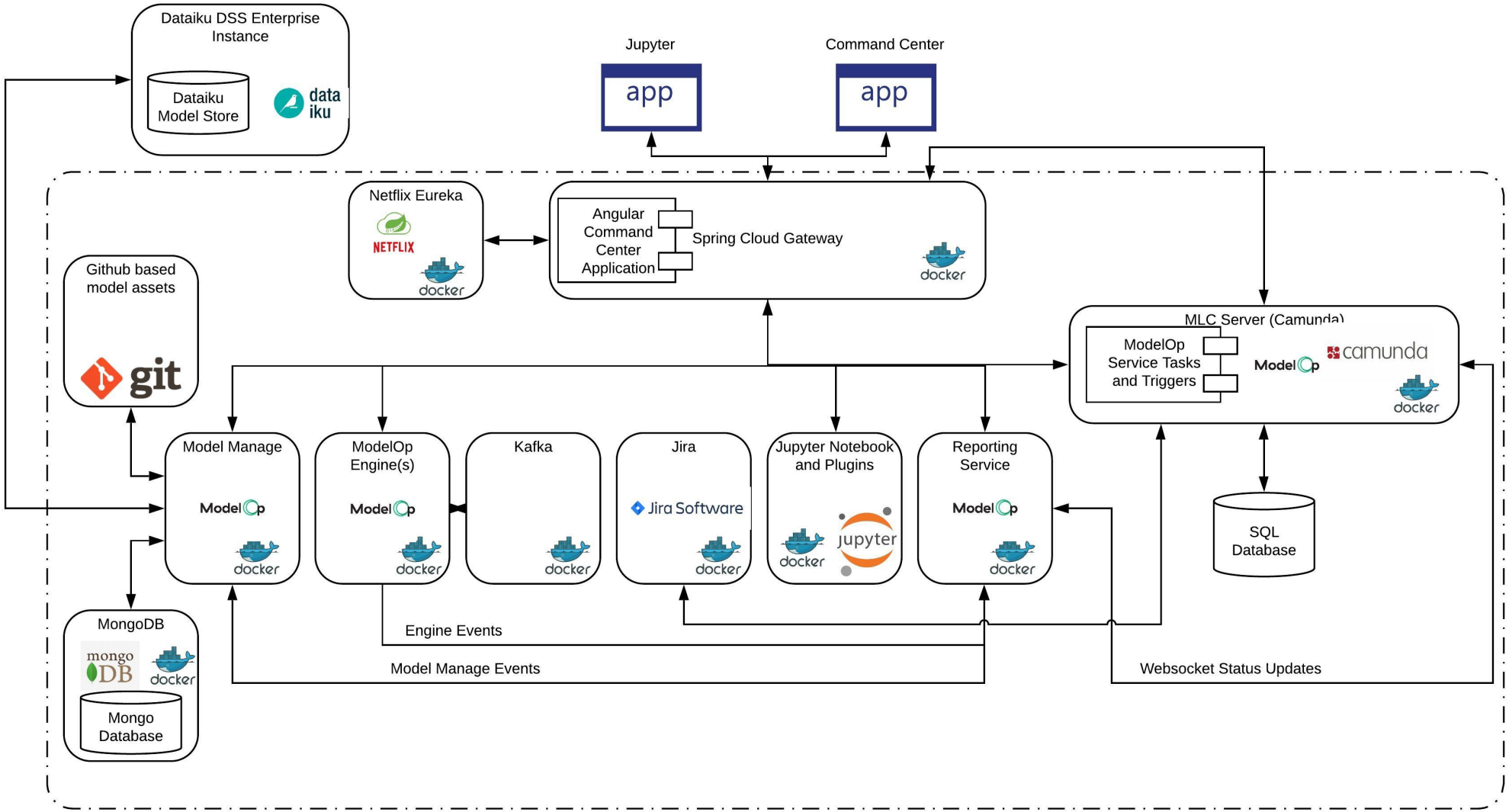
Presented by:
Jim Olsen, CTO

May 11th, 2020

NEXT SESSION

5/20: "ModelOp Center + BI Tools: measure the business value of enterprise AI initiatives"

ModelOp Center Architecture



BPMN – Business Process Model and Notation

A way to describe the flow of business workflow and decisions throughout a long running business process that may include external systems and human interactions.

- It is not a simple batch job workflow methodology for implementing short running tasks, such as things like airflow
- Should be used when the job can span many days and many organizations, or complex decisions within the workflow need to occur

An industry standard since 2004, but v2.0 was release in 2011

- We will be using Camunda's modeling tool, but the language is tool agnostic
- XML under the covers, but the graphical representation is extremely important

ModelOp

BPMN – The Process

The overall process defines what is to be done

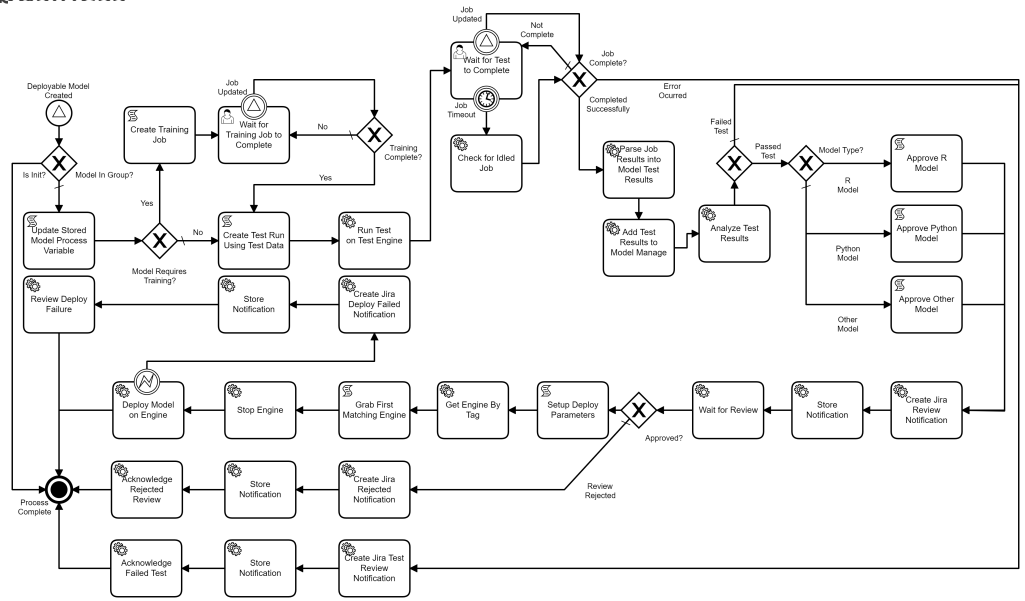
- A collection of all of the different bpm elements that is used to describe a complete business process
- Could be small and execute in seconds, or run for months at a time

Processes have multiple process instances

- Each instance is tracked individually for its current state

Process instances have process variables

- These variables are very important in that they carry state from activity to activity
- Allow the process to store process instance specific variables
- These variables can contain a simple value, or a complete object



BPMN – The Elements (Events)

Start events begin the process on a variety of conditions

Signals are externally generated to the process either by another process or an event within the Model Lifecycle Server. ModelOp center provides a variety of signals from the events occurring in the Model Lifecycle.

Timers are regularly occurring events, or one time date specific events, that can be expressed in a similar manner to cron jobs, or simple reoccurring on a duration type basis.

End events terminate the process

Can be a simple terminate which ends the process and marks it as complete

An end Event can also be a signal to allow other process(es) to start and begin additional processing

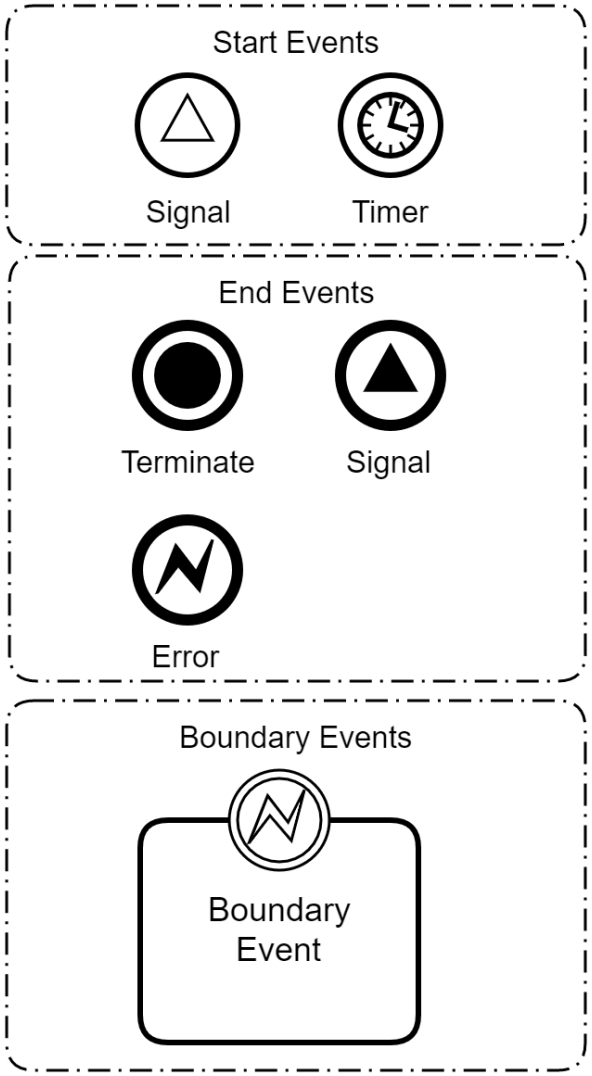
They can also end in an error state indicating the process has failed and requires attention

Intermediate events interrupt activities

These events can be any of the different types, such as timers, signals, or errors.

An example is allowing an activity to only wait 3 days, and escalate to another individual if the activity has not been completed.

Also can be used to specify the error path if an error occurs during an activity



BPMN – The Elements (Activities)

Activities are the workhorses of the bpmn world

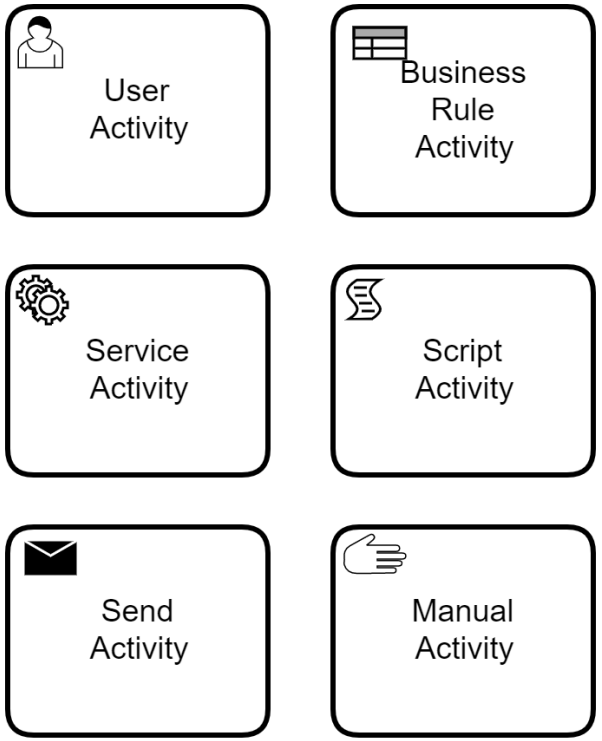
- User activities require the completion of the activity by a human
- Business rule activities make decisions using dmn or cmmn
- Script activities execute a variety of scripting languages
- Send activities send messages
- Manual activities occur outside of the bpm engine

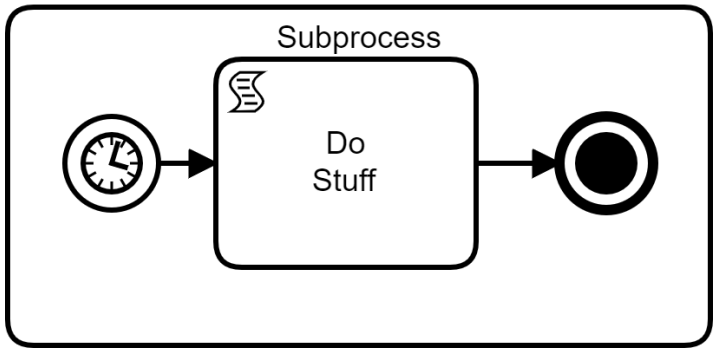
Service activities have a variety of types and are heavily used in ModelOp Center

- Java class activities are implemented as delegates. ModelOp Center provides many delegates for interacting with our services.
- External activities occur outside of the bpm engine but the progress is tracked by the engine to determine when the activity is complete

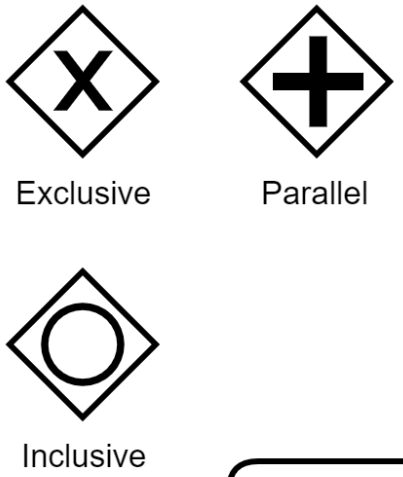
Subprocesses allow reuse throughout your organization

- Organizes a complete sub unit of execution into a reusable item that can be shared across multiple business processes to simplify commonalities.





BPMN – The Elements (Gateways)

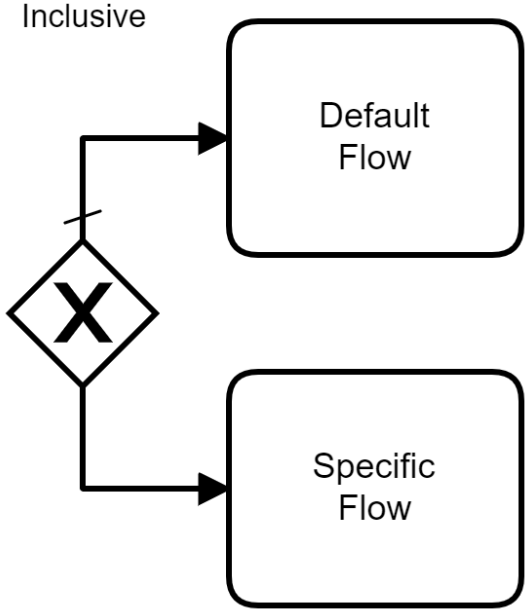


Gateways control the flow of the business process

- Exclusive gateways, the most common, control the flow based on values of process variables, scripting results, or other values within the process
- Parallel gateways allow multiple flows to occur at the same time
- Inclusive gateways make sure that all flows are evaluated through the business process

Exclusive gateways allow specific flows, as well as a default flow

- Specific matching flows will be followed if the criteria is met. This can be specified as a simple expression, or a more complex script
- Default paths, indicated with the hash, will be followed if no specific matching flow is found



Subprocesses allow reuse throughout your organization

- Organizes a complete sub unit of execution into a reusable item that can be shared across multiple business processes to simplify commonalities.

ModelOp Center Signals

Used to start processes based on events within the ModelOp Center

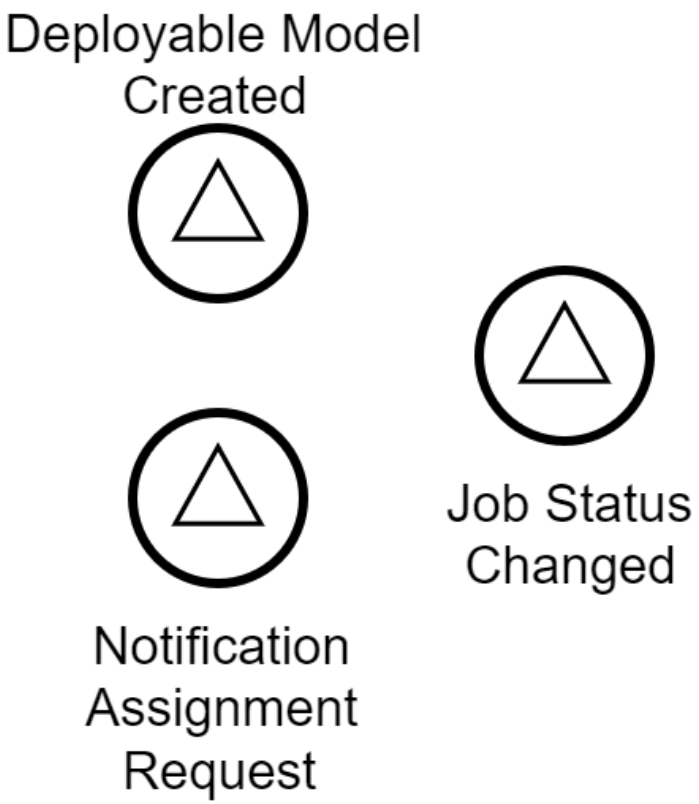
MODEL_CHANGED – A stored model has been updated

JOB_STATUS_CHANGED – A job has changed its status

DEPLOYABLE_MODEL_CHANGED – A new version of a model has been published

NOTIFICATION_ASSIGNED – A notification has been assigned

DEPLOYED_MODEL_CHANGED – A deployed model has been created or updated



ModelOp Center Delegates

All of the different activities that can be performed within the ModelOp Center

Perform actions within the ModelOp Center framework and runtimes

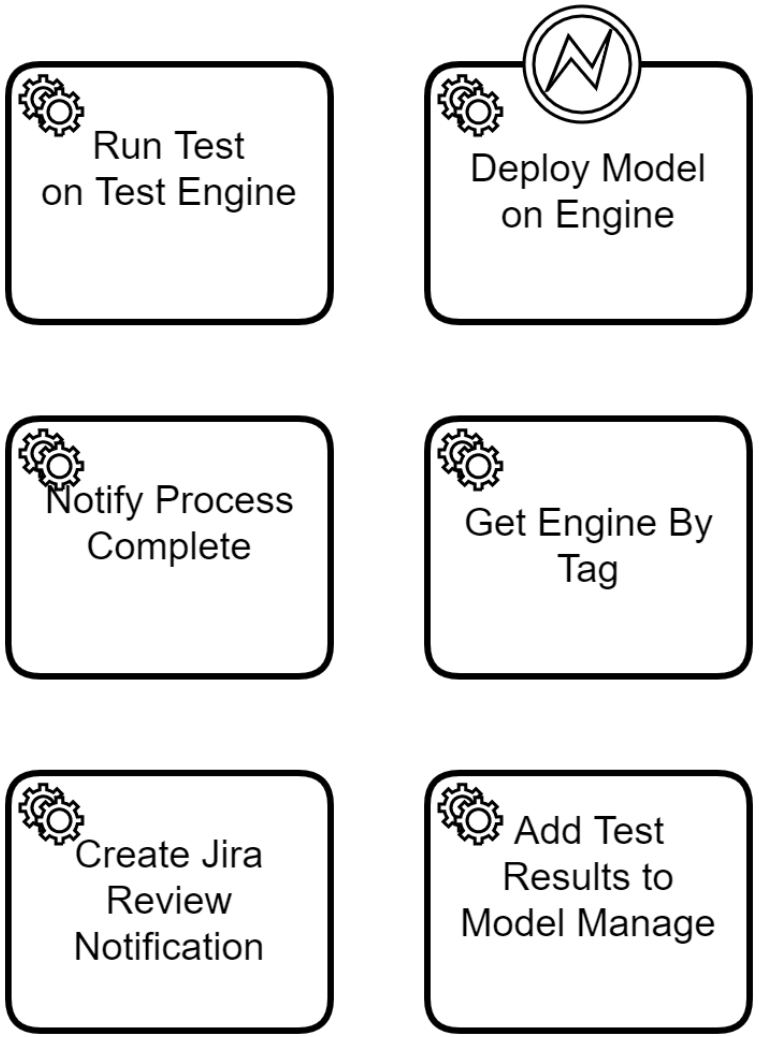
- Deploy a model to execute long term on a runtime environment
- Execute a predictive batch run on a runtime environment
- Execute a training or metrics job to train or test a model and capture results

Interact with the data contained in ModelOp Center

- Read or update any of the data model items such as runtimes, models, notifications, test results, and many many more.
- Get updated status or runtimes, jobs, and any other kinds of tasks running within ModelOp Center
- Create notifications within the system to notify of varying kinds of events

Manage external tasks and processes critical to your business

- Link Jira and Service Now tickets to your business process and ModelOp Center data
- Interact with any RESTful or SOAP based service within your enterprise



Hands On – Build Two Example Model Lifecycles

Build a simple Hello World Process to add a notification to ModelOp Center every minute

- Build the process using Camunda Modeler and ModelOp Center Delegates
- Deploy to ModelOp Center Lifecycle Manger using the tool

Build a more complex process to run a test on a new model version and seek a review of the test results

- Use a ModelOp Center signal to start the process
- Execute a metrics job to test a model and capture results
- Create a Jira ticket to notify someone of the test results for review

ModelOp

Thank you

modelop.com

vimeo.com/modelop

learn@modelop.com

@ModelOp_Co

@ModelOp

@ModelOp

ModelOp