WebSphere Education

Creating a business process diagram

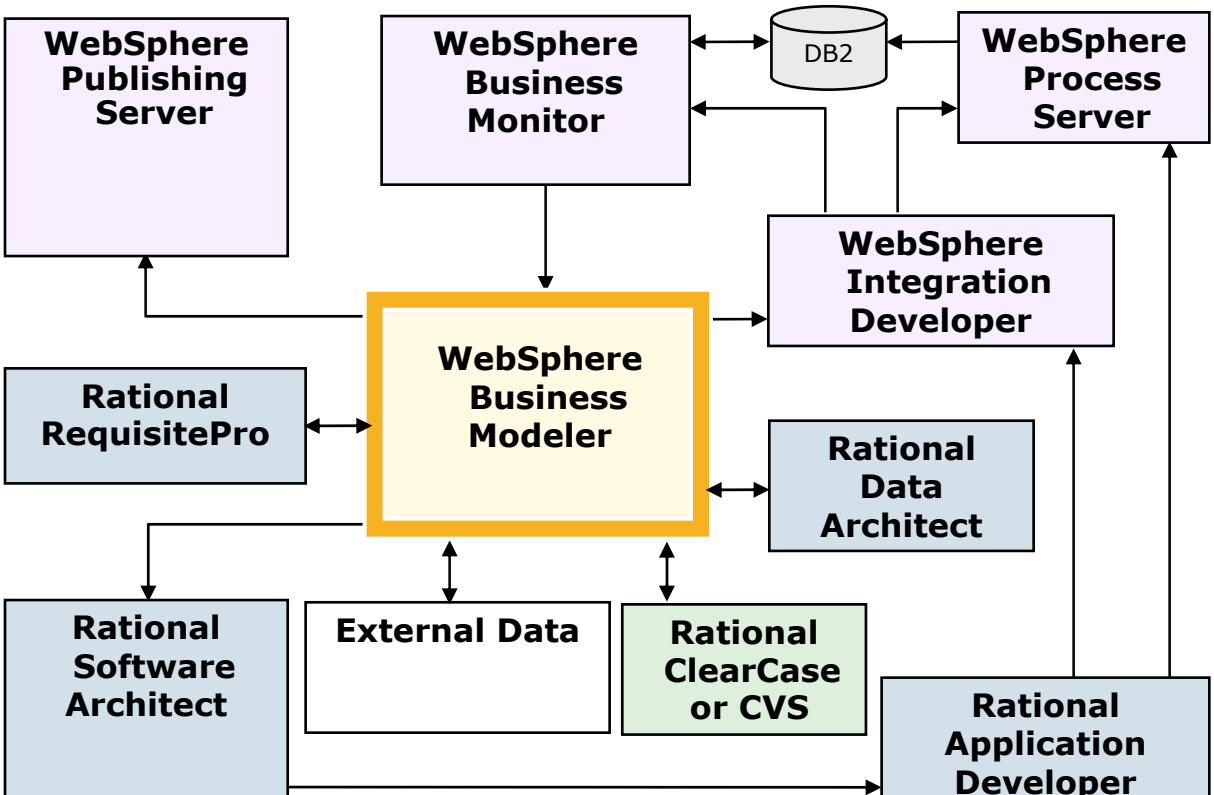
Unit 4

Unit objectives

After completing this unit, you should be able to:

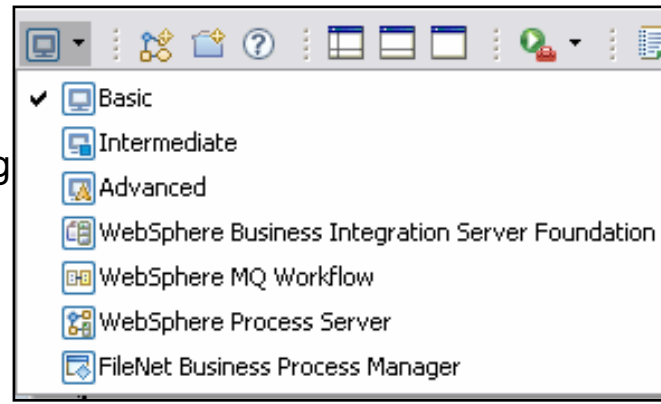
- Identify the modeling modes available in WebSphere Business Modeler
- Describe the contents of the Project Tree
- Explain the process modeling elements
- Describe the relationship between local and global activities

Creating a business process diagram



Modeling modes in WebSphere Business Modeler

- Business modeling modes for process mapping:
  - Basic business modeling
  - Intermediate business modeling
  - Advanced business modeling
- Business modeling modes that apply validation rules:
  - WebSphere Business Integration Server Foundation
  - WebSphere MQ Workflow
  - WebSphere Process Server
  - FileNet Business Process Manager



- When switching modes, the following changes occur:
  - Some options are not available
  - A previously valid model may no longer be valid because of additional validation rules
- No information is lost when switching modes

Business modeling modes for process mapping

Basic (default)

- Focuses on creating and displaying sequence flows
- Filters out fine-grained technical details of process and data modeling
- Seven attribute tabs

Intermediate

- Allows the more technically-focused user to specify and view additional details of process and data models
- Nine attribute tabs

Advanced

- Facilitates the work of technical personnel who prepare models that will be used as the basis for software applications
- Eleven attribute tabs

Business modeling modes for validation

WebSphere Business Integration Server Foundation

- Applies validation rules to support export of models to Application Developer Integration Edition

WebSphere MQ Workflow

- Applies validation rules to support export of model in FlowMark Definition Language (FDL) for IBM WebSphere MQ Workflow

WebSphere Process Server

- Applies validation rules to support export of model to WebSphere Integration Developer (WID) and subsequently deployment on runtime WebSphere Process Server

FileNet Business Process Manager

- Applies validation rules to support export of model to FileNet P8 platform
- The model can then be refined and deployed on FileNet P8 Process Engine

Primary modeling elements used in Modeler

Category	Modeler element	Description
Activity Represents the work being performed	Task	Basic unit of work
	Process	Sequence of activities
	Service	Process external to the organization
Storage Represents storage area	Repository	Location where business items are stored
Flow control Determines the process flow	Simple decision	Routes inputs to one of two paths
	Multiple-choice decision	Routes inputs to one of several paths
	Fork	Splits a path into two or more parallel paths
	Merge	Combines two or more paths after an exclusive decision
	Join	Combines two or more parallel paths
	Connection	Links two elements to represent the flow
Control nodes	Start	Marks the beginning of a process not initiated by another process
	Stop (Required)	Marks the termination of a process
	End	Marks the end of a path in a process

Activity: Task

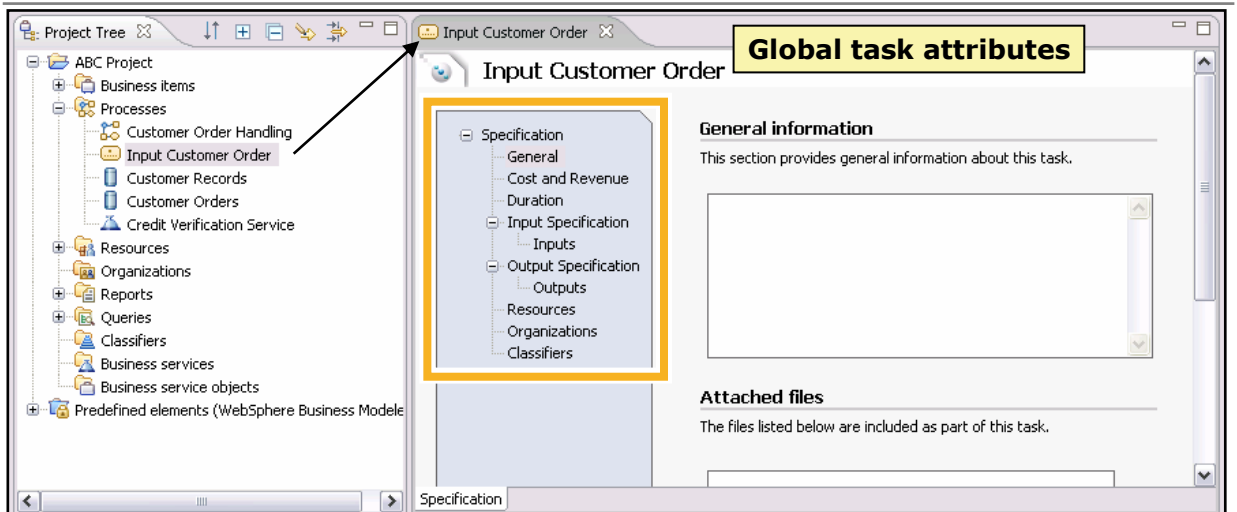
- Basic unit of work performed in the process and cannot be decomposed
- Two types
  - Local: Specific to a process, created on the palette
  - Global: Reusable within a workspace; created in the Project Tree
- Details are described in the Attributes view or global element editor

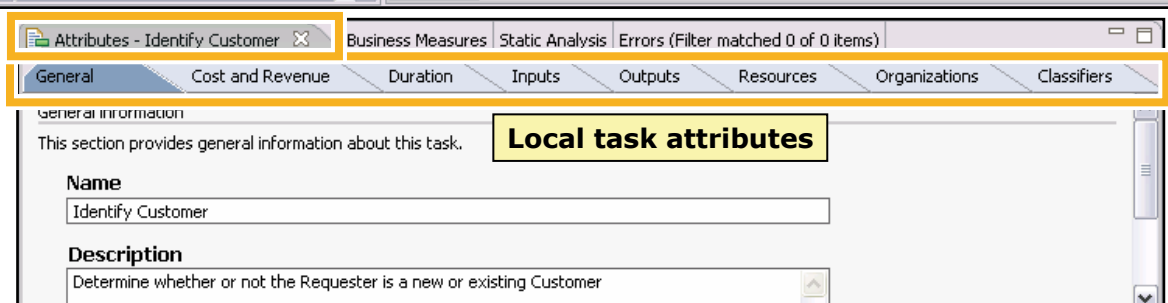
When to use global versus local tasks

- If the activity will be performed within many processes and its specifications are unlikely to change, create a **global** task.
  - Otherwise, create a **local** task.
- A **global** task will always have the same attributes.
  - If a task needs to behave differently or be performed by different roles in different processes, create different **local** tasks instead.
- WebSphere Business Modeler provides the ability to convert local elements to global status.

Global task	**Local task**
Top-level task created in Project Tree view within a process catalog	Owned by a process and can only be used by elements within that process
Can be used by multiple processes within the project	Exists only while the process exists
Must be opened from Project Tree view for editing	Edited using Attributes view

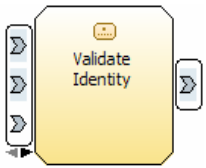
Global versus local task attributes

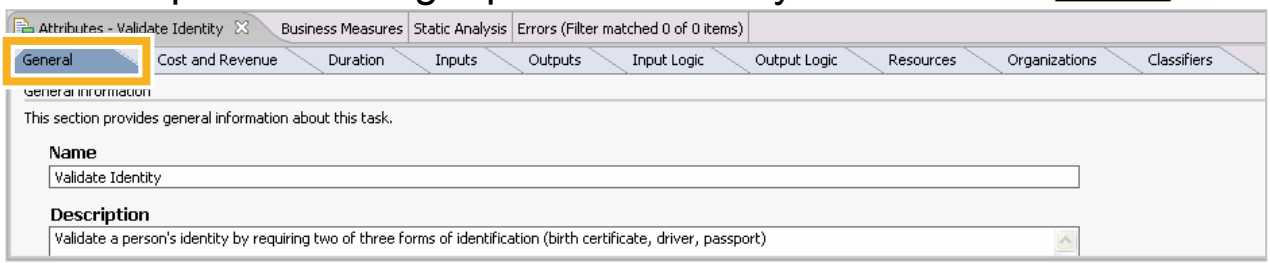


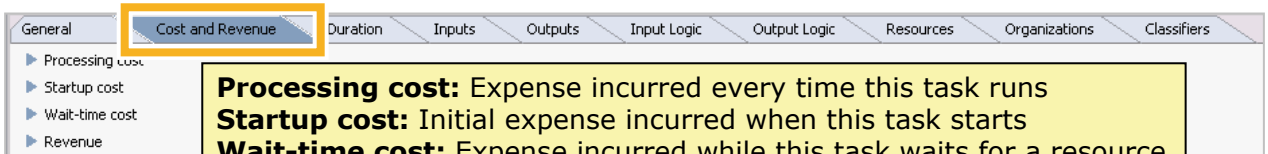


Task: Detailed attributes (1 of 6)

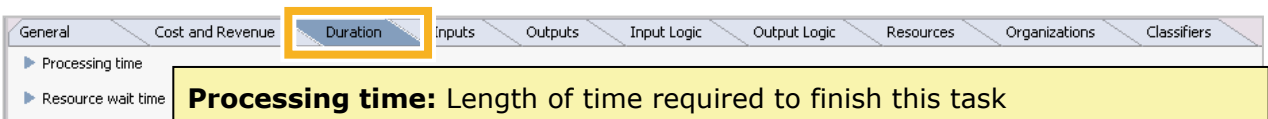
- Detailed attributes complete the definition of the task
- Example: Validating a person's identity







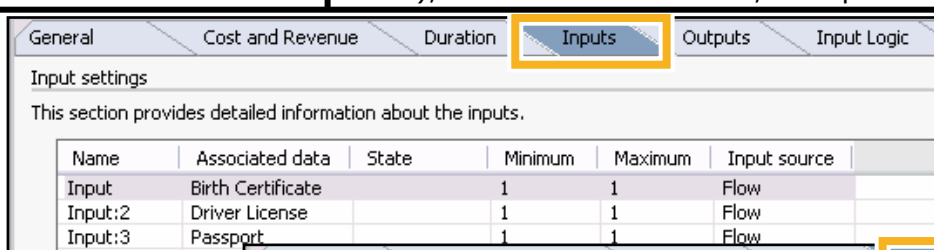
Processing cost: Expense incurred every time this task runs
Startup cost: Initial expense incurred when this task starts
Wait-time cost: Expense incurred while this task waits for a resource
Revenue: Money earned when this task is finished

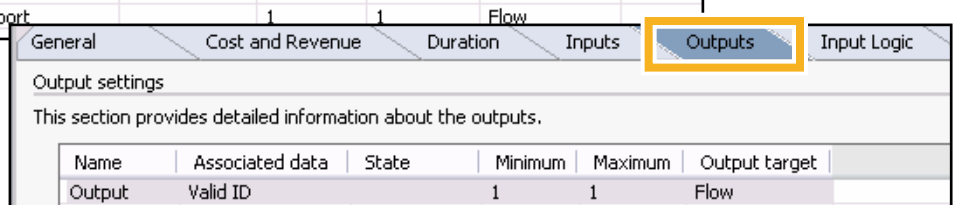


Processing time: Length of time required to finish this task
Resource wait time: Maximum length of time permitted to wait for a resource

Task: Detailed attributes (2 of 6)

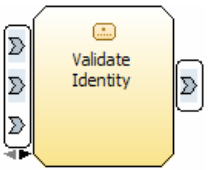
Input or output name	Can be changed to provide more meaning
Associated data	Business item associated with input or output
State	Business item state associated with input or output
Minimum, maximum	In input settings: Number of business items required to execute task In output settings: Number of business items to be created
Input source or output target	**Flow:** Task retrieves data from an upstream element **Repository:** Task retrieves data from a repository **Constant** (input source only): Specify constant value (such as "100"), business item instance, or expression





Task: Detailed attributes (3 of 6)

- Input criteria
  - Specifies the combination of business items needed for the task to start
  - Default uses “AND” logic (all business items are required)
  - “OR” logic is represented by adding separate rows for each criterion
  - Complex logic is represented by small two-headed arrow below the task inputs
- Preconditions
  - Specifies the values of the business item attributes needed for the task to start
  - Each condition is named and described
  - An expression is created to specify how each condition is to be evaluated



General | Cost and Revenue | Duration | Inputs | Outputs | **Input Logic** | Output Logic | Resources

Input criteria

This section shows the input criteria for this element. At least one of these criteria must be satisfied in order for the element to start.

	Name	Input	Input:2	Input:3	Criterion
	Input Criterion	☑	☑	☐	Input AND Input:2
OR	Input Criterion:2	☑	☐	☑	Input AND Input:3
OR	Input Criterion:3	☐	☑	☑	Input:2 AND Input:3

Preconditions

This section shows the preconditions for this element. These conditions must be met before the element can start.

Country of Birth

Add

Remove

Name

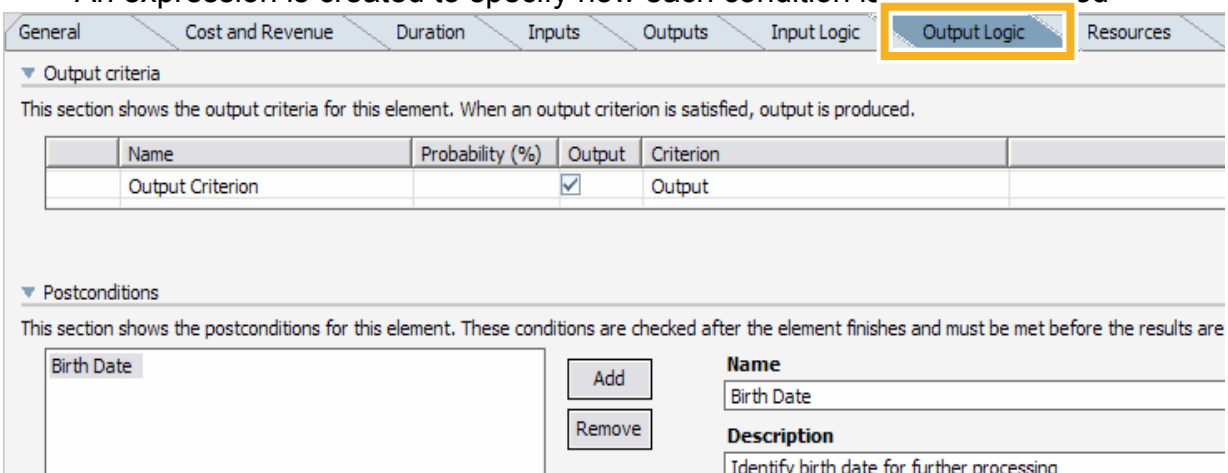
Country of Birth

Description

Determine the country of birth for exceptions to documentati

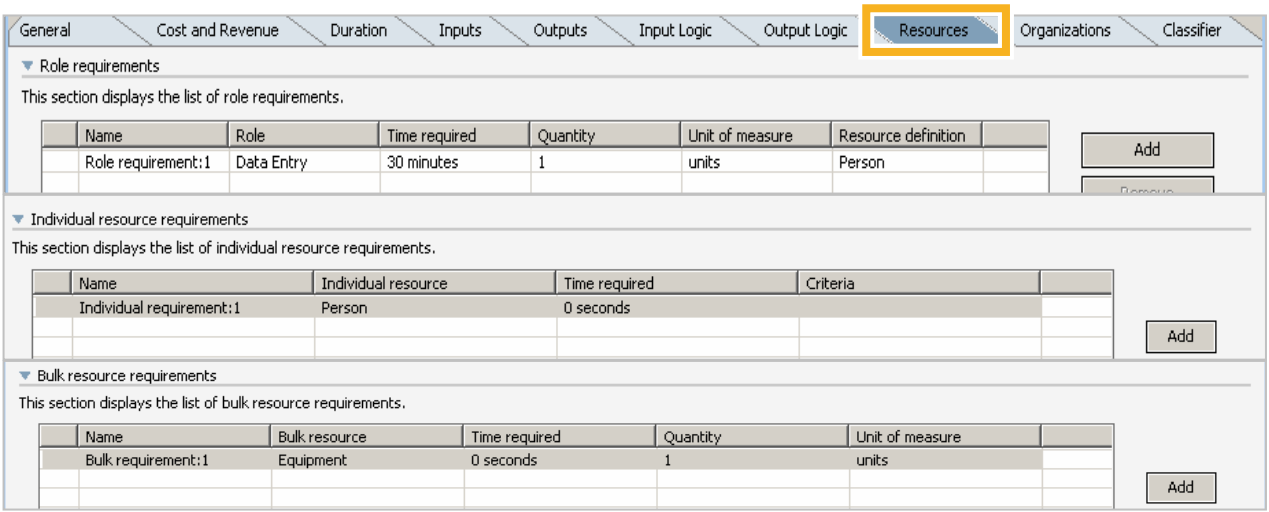
Task: Detailed attributes (4 of 6)

- Output criteria
  - Specifies the combination of business items to be produced
  - Default uses “AND” logic (all business items created)
  - “OR” logic is represented by adding separate rows for each criterion.
  - Complex logic is represented by small two-headed arrow below the task outputs
- Postconditions
  - Specifies the values of the business item attributes to be produced
  - Each condition is named and described
  - An expression is created to specify how each condition is to be evaluated



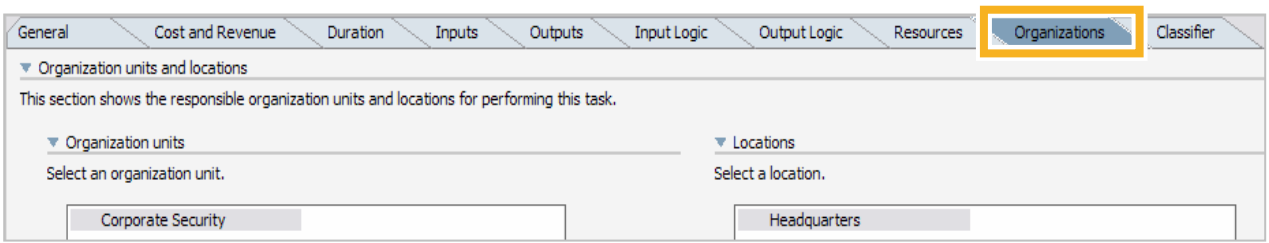
Task: Detailed attributes (5 of 6)

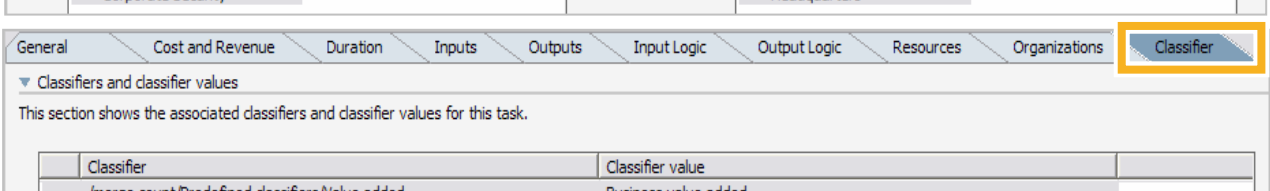
- Resources
  - Role requirements — the role or roles required to complete the task
  - Individual resource requirements — the specific individual resources required to complete the task
  - Bulk resource requirements — the bulk resources required to complete the task
- Time required
  - The time the resource requires to complete the work for the task
    - May be different for each resource and different from the task duration
- Quantity
  - The number of resources needed to complete the specific task, not those available



Task: Detailed attributes (6 of 6)

- Organizations
  - Organization units
    - Organization units responsible for this task
  - Locations
    - Locations responsible for this task
- Classifier
  - Classifier
    - Multiple classifiers per task
  - Classifier value
    - One classifier value per classifier





Activity: Process

- Processes are more complex than tasks
  - Represented as a sequence of activities that are linked by flows of control and data
  - Activities can be other processes as well as tasks and other elements that control flow
- Decisions in a process determine the way a process is executed
  - Each possible path of execution is known as a case
- Two types: Local and global
- Described in the Attributes view
- Associated cost, revenue, and duration can specified at the process level
  - Task level details may be unknown or too detailed in some analyses

When to use global versus local processes

- Creating **global** processes should be the general rule.
  - If the process is so unique that it is unlikely to be reused in any other process or is needed for functional structuring only, use a **local** process.
- Model at a high level of detail and then iteratively add details to the model.
  - Do not get sidetracked by modeling a nested process before completing the parent process.
  - Create placeholders for the processes first, and then work on the details of each process after the parent process is complete.
- WebSphere Business Modeler provides the ability to convert local elements to global status.

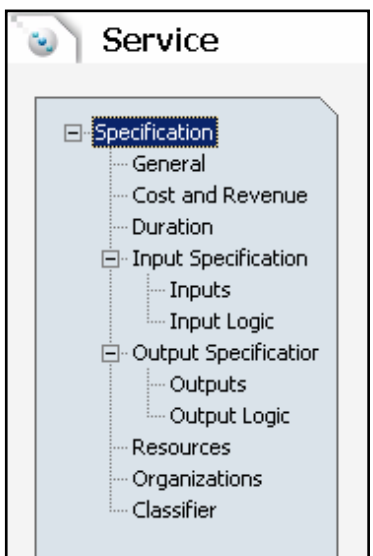
Global process	**Local process**
Enables a collection of activities to be reused in multiple processes across functional areas	Logical collection of activities that only exists within its parent process
Displayed in Project Tree	Displayed only within process diagram of parent process

When to use processes versus tasks

- Both processes and tasks represent activities that are performed in a business.
- Processes are more complex than tasks and can be represented as a sequence of activities that are linked by flows of control and data.
- Tasks are atomic activities that cannot be divided into smaller actions.
- If details of the activity are critical for communication and analysis, create a **process** so that the intent of the process can be precisely documented and the relevant data can be analyzed.

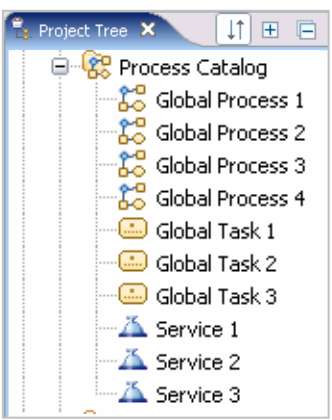
Activity: Service

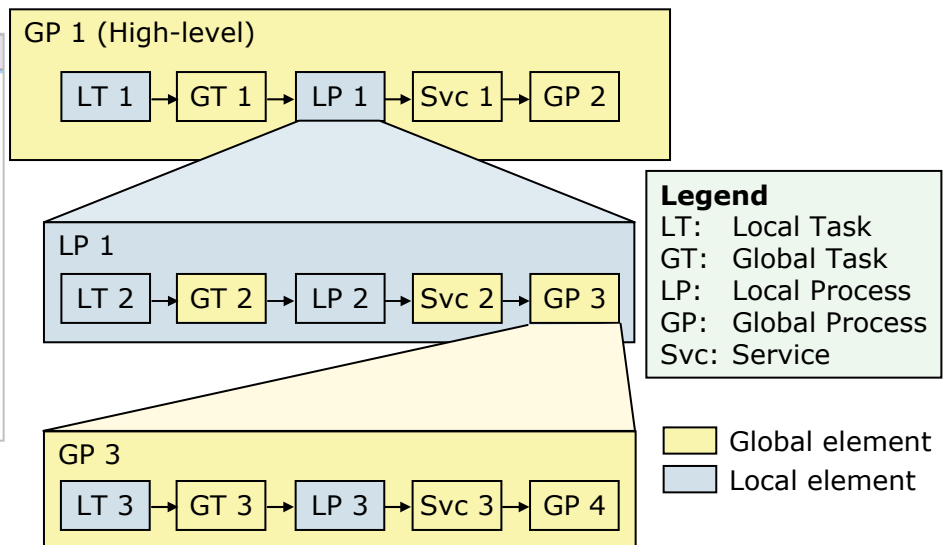
- Services are external processes executed outside the organization
  - Can be used within or by the organization's processes
- Services are similar to processes, but the internal activities are unknown
  - Treated as a "black box," outside the control of the organization
  - Example: Credit check
- Essential to modeling the realities of business-to-business integration
  - Represent organizations like credit bureaus, payroll companies, or shipping companies
- Can only be created in Project Tree
- Described in the global element editor
- Associated cost, revenue, and duration are specified for the whole service
  - Task level details are unknown ("black box")



Relationship between local and global activities

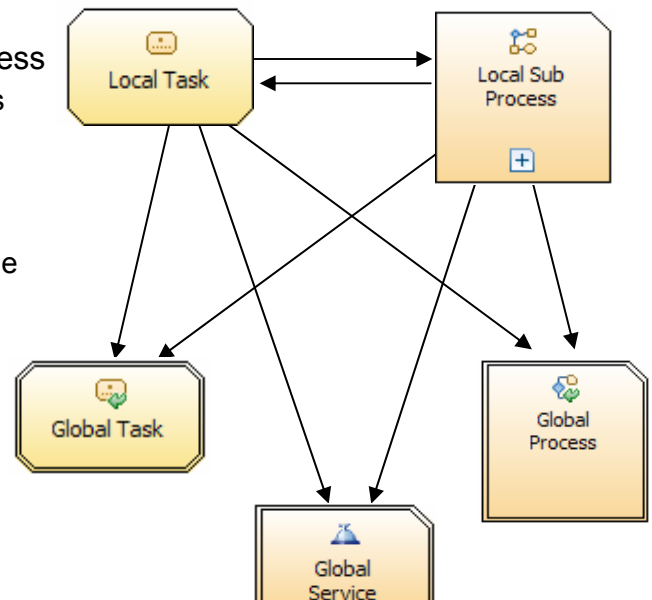
- Every global or local process can reuse other global elements along with its own local elements
- Only global elements are listed in the Project Tree





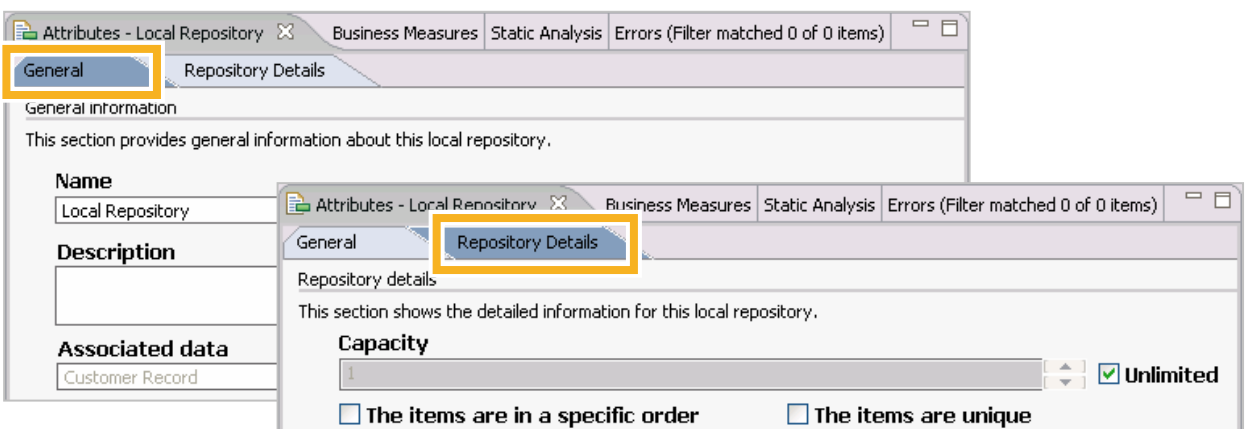
Converting elements

- Elements can be changed during the documentation or design process
- Converting a local task to a local process
  - Justification: The task actually contains several tasks
- Converting a local process to a local task
  - Justification: The process is really at the lowest level that can be displayed
- Converting local tasks and processes to global tasks, processes, or services
  - Activities are reusable or will require specific analysis
  - New global elements will appear in Project Tree
    - All existing attributes and connections are maintained
- Some information is lost when converting elements



Storage: Repository

- Repositories represent storage areas for business items (data)
  - Business items are either taken from or placed in a repository
- A repository can only hold a single type of business item
  - Multiple business items must be combined into a new business item before they can be stored in a repository
- Two types: Local and global
- Described in the Attributes view or global element editor

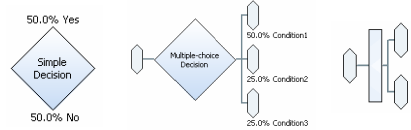


When to use global versus local repositories

- Global repository provides the mechanism to model the sharing of information across processes.
  - Use global repositories whenever information is to be shared across disparate processes.
- With local repositories, the information does not persist outside the boundaries of the parent process.
- Unlike global processes and tasks, where only the definition is shared, the repository instance is shared at run time.

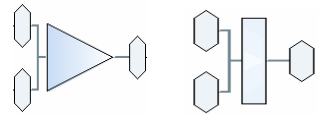
Global repository	Local repository
Top-level repository created in the Project Tree view	Owned by a process and can only be used by elements within that process
Can be used by multiple processes	Exists only while the process exists

Flow control: Decision and fork



- Decisions are flow control constructs rather than activities like tasks or processes
  - No costs or duration
  - Used to show alternate paths from a preceding activity
  - Named in the form of a question: “Type of order?”
- Simple decision
  - Has one incoming branch with one input and two exclusive outgoing branches, each with one output
- Multiple-choice decision
  - Has one incoming branch and multiple outgoing branches
  - Incoming branch allows more than one input
  - Each outgoing branch has an associated condition, can be inclusive
  - Output branches can be labeled
- Fork
  - Splits the process flow into parallel paths
  - Enables two or more paths to be performed independently of each other
  - A fork makes copies of its inputs and forwards them along each of the parallel paths
- Branching and splitting conditions are specified in the Attributes view

Flow control: Merge and join

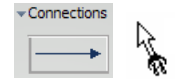


- Merges and joins combine multiple processing paths into a single flow
- Merge
  - Combines two or more alternative paths back into a single flow
  - Usually used after exclusive decisions
    - Only one outgoing branch of an exclusive decision is ever taken
    - A merge runs whenever one of its incoming branches is satisfied
- Join
  - Combines two or more parallel flows back into a single flow
  - Synchronizes the paths, by waiting for each input to arrive
    - Only permits the output to continue after all inputs have arrived
  - Usually used after forks
    - All outgoing branches of a fork are taken
    - A join runs only when all of its incoming branches are satisfied
- Merging and joining conditions are specified in the Attributes view

When to use merges versus joins

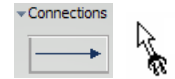
- Use a merge or join to redirect processing paths to a single connection after a process has been divided into more than one path through a decision or fork.
- The use of a merge or a join influences the number of times the elements downstream will run.
  - Because a merge runs whenever one of its incoming branches is satisfied, downstream elements can potentially run more than once.
  - Because a join runs only when all of its incoming branches are satisfied, downstream elements run once.
- If there are parallel paths, use a merge when you expect that only one of the parallel paths will be taken, and a join when you expect that all of the parallel paths will be taken.
- Both merges and joins merely combine (and, in the case of joins, synchronize) flows in the process; they do not combine business items.
  - Example: A join that receives two Order business items, one from each branch, will produce two Orders. If you want business items consolidated into one input, you must add a specific task to do that.

Flow control: Connection (1 of 2)



- A connection is a link between two elements
- Connections are used to specify the chronological sequence of activities in a process
  - Each task, subprocess, decision, or other element passes control to the next task or element along a connection
  - You can associate business items with connections to pass data from element to element
  - Each connection can have only one associated business item.
    - You can use multiple connections if you need to pass multiple business items between two elements

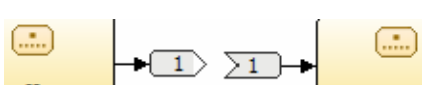
Flow control: Connection (2 of 2)



- The connection icon appears as a plug with an arrowhead indicating the connection point
  - The “Not” symbol appears when the cursor is hovering over a position that cannot be connected

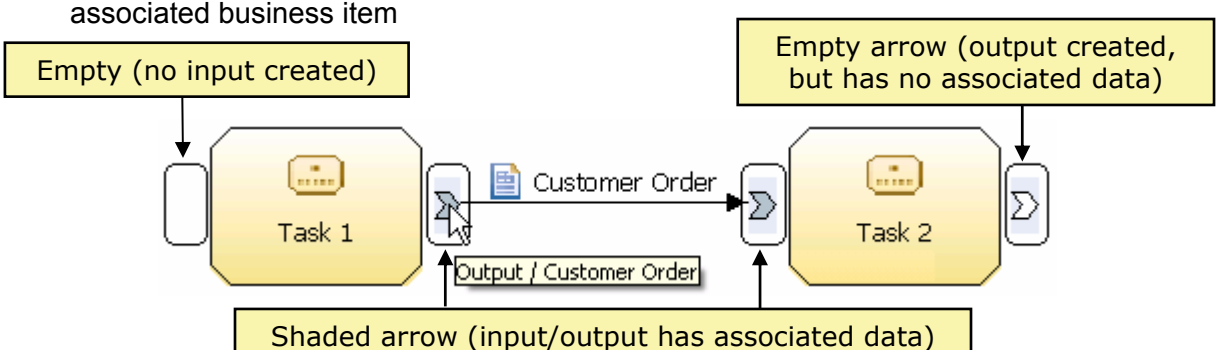
  - The “Not” symbol disappears when hovering over a valid connection point

- In a very large diagram, you can split the connection into two individual nodes and connections



Understanding connection graphics

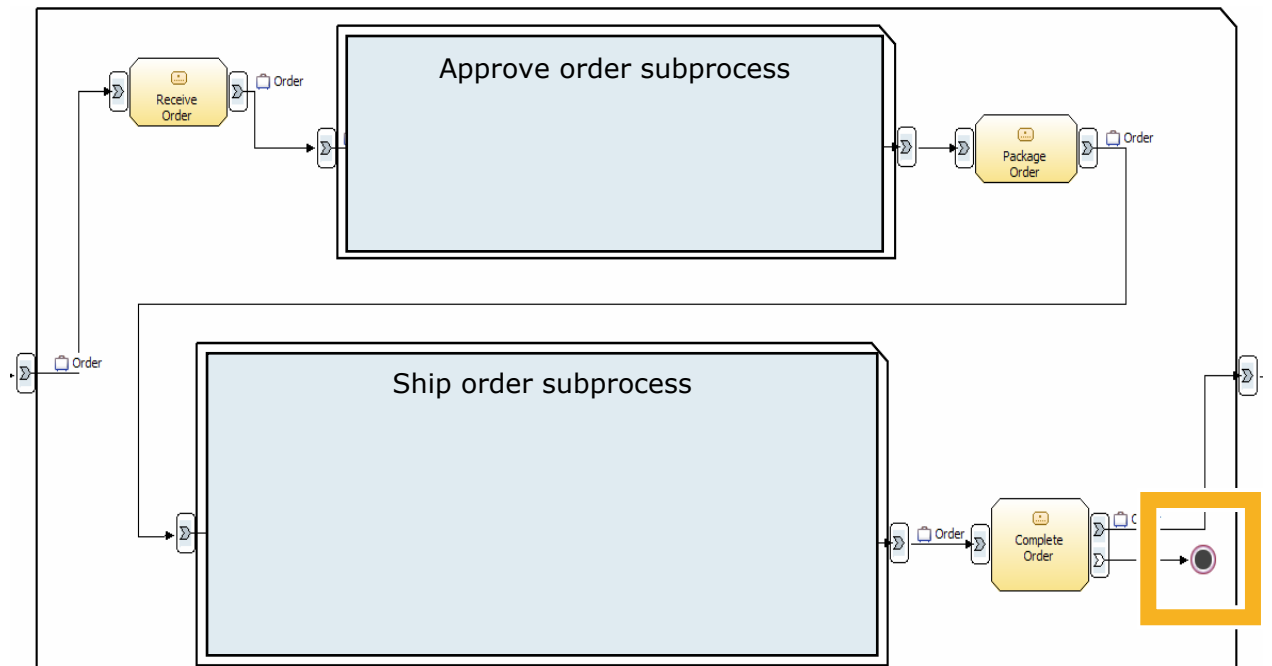
- Terminology
  - Input criterion, output criterion: Small rounded rectangle attached to the right and left of a task; tasks can have multiple criteria
  - Input, output: Small triangular icon in the criterion box; a criterion can contain multiple inputs or outputs
- What the graphics represent
  - Empty input/output criterion (rounded rectangle) indicates there is no input assigned
  - White input/output icon indicates an assignment was made but no business item associated
  - Shaded input/output icon indicates a business item is assigned
- How to view the assignments
  - Hover over the inputs and outputs to see the name of the input or output and the associated business item



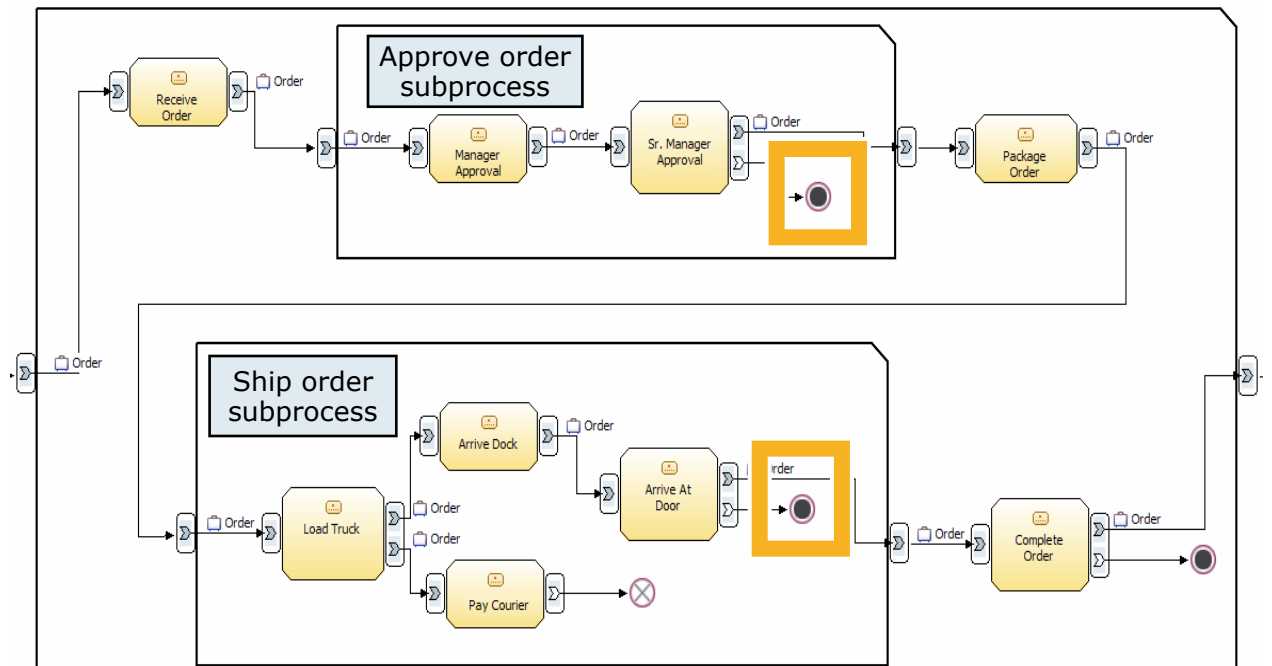
Control nodes: Start, stop, and end

- Start
  - Identifies the beginning of a process flow
  - Usually used in top-level process within project
    - There are also many processes that start without data, usually because they are triggered by events in the real world, such as a customer arriving or an employee beginning work.
  - Automatically appears in any process you create in Modeler
  - You can delete the start node and connect the process input to the first activity in a process
- Stop
  - Marks the termination of a process
    - Every process, subprocess, and loop **must** have at least one stop node
    - If there are multiple points at which a process can be completed, each of these points must have a stop node
  - Can also be used to stop a process at any point where termination of all process activity is required, such as order cancellation
  - During simulation, the control flow and any business items are passed to the calling process when the stop node is reached
- End
  - A visual marker within a process that identifies where a particular flow ends
    - Concurrent flows within the same process will still continue executing
    - An end node should only be used to mark the end of a particular process path and not the completion of a process
  - End nodes are optional; they permit you to express logic directly in the diagram

In highest level, flow must be terminated by a stop node



Each subprocess must also contain a stop node



*Note – Expanded view only available in simulation

Checkpoint: Creating a business process diagram

Your instructor will review these questions with you as a group. If time permits, the instructor may provide you time to answer the questions on your own before the group discussion.

1. What are the three modeling modes you can use when building a process model?
2. What is the function of a repository in WebSphere Business Modeler?
3. What is the difference between a task and a process?
4. What is the difference between a stop and an end?
5. What is a service in WebSphere Business Modeler?
6. What is the purpose of converting local elements to global ones?
7. Name the flow control elements that can be used in a process model.

Checkpoint solutions: Creating a business process diagram

1. Basic, intermediate, and advanced
2. Repositories represent locations for business items
3. A task is a basic unit of work performed in the process, and cannot be decomposed
4. Processes are more complex than tasks, and are represented as a sequence of activities that are linked by flows of control and data. Activities can be other processes as well as tasks and other elements that control flow.
5. A stop node marks the termination of a process.
   An end node is a visual marker within a process that identifies where a particular flow ends.
6. Services are external processes executed outside the organization
7. Activities are reusable or will require specific analysis, and the new global elements will appear in the Project Tree
8. Simple decision, multiple-choice decision, merge, fork, join, and connection

Unit summary

Having completed this unit, you should be able to:

- Identify the modeling modes available in WebSphere Business Modeler
- Describe the contents of the Project Tree
- Explain the process modeling elements
- Describe the relationship between local and global activities

Exercise overview

In this exercise, you will:

- Add more elements to the process diagram
- Associate business items and connecting process elements
- Add a repository
- Convert local elements to global elements