

Tasks

The workflow is a sequence of steps, executed by the workflow engine. The sequence of the workflow is made of tasks or [blocks](#). Tasks can use the data stored in [context stores](#) defined for this workflow.

Attributes

StartRule	Defines a rule which is evaluated before the task is performed. If the rule returns <ul style="list-style-type: none">• True – The step is executed• False – The step is skipped
StartRuleContextStore	Context store upon which the start rule is evaluated.
EndRule	Defines a rule which is evaluated after the step is finished. If the rule has thrown any exceptions, the containing block is stopped from performing. If no exceptions are thrown, the workflow continues on to the next steps.
EndRuleContextStore	Context store upon which the end rule is evaluated.
Trace	Read only. True if the data about the execution of this step are recorded to the trace log for debugging purposes.
TraceLevel	Specifies how the data about the execution of this step will be recorded in the trace log for debugging purposes. All steps have InheritFromParent set as default, so e.g. if you turn on tracing for the workflow or one of its blocks, all steps will inherit this setting and will be traced.
OutputContextStore	Defines a context store into which any output data will be merged after the step is finished. E.g. which context store will get filled by the data returned by the data service.
OutputMethod	Specifies how the data will get merged into the <code>OutputContextStore</code> . For possible values see Output Methods below.

Output Methods

Ignore	Any output data will be ignored and no context will be updated when the task finishes.
AppendMergeExisting	Any updates to the existing data will be updated using the entity primary key. Any new data (not found in the original context by the primary key) will be appended to the context. No data will be deleted from the context. This method is the default for the Service Method Call task, because e.g. the data transformation will typically return subset of the XML data, so they can be updated in the context. If FullMerge is used, all missing data would get deleted.
FullMerge	The context will get updated, so that it will contain only the data returned by the finished task. The data found by using the primary key will get updated, the data not found by the primary key will get inserted and the data which exist in the context but do not exist in the data returned by the task will get deleted from the context. This method is the default for the User Interface task, because users are allowed to do any updates to the data, including deletions. It is intended that the context shows exactly the same data after editing that the user saw on the screen, therefore a full comparison and merge must be performed between the data returned from the task and the context.

Trace Levels

None	Nothing will be recorded.
InheritFromParent	Default. The data will be recorded only if the parent steps specify that the data should be recorded.
TraceArchitect	The data will be recorded only if the workflow is executed from ORIGAM Architect. This is important when you want to debug workflows, but you do not want other users, running ORIGAM Desktop Client, to interfere with the trace log.

TraceClientAndArchitect

Data will always be recorded. This is important if you cannot reproduce a bug that your customers can. You turn this option on and after the user reproduces the behavior, you can turn the tracing off and analyze the trace log.

Task Types

- Service Method Call
- Show User Interface
- Sequential Workflow Call
- Set Workflow Property
- Check Rule
- Wait