Flow Lifecycle

Creating Flow

1. Create Flow ID (flowId), it is an application deal, it is usually a database ID, but that number is not important while processing flow
2. Create Flow Carter, it makes IFlowProcessor itself. If you need change Flow Carter object, you have to override the creator method.
3. Initialize flow, the flow must ever have an initial step. If there is a default inital step in configuration, than it is used. Flow cannot start in a step, that is not marked as initial.
4. If the initial step is a virtual step, than the step is processed until a real step is reached.
5. Flow is waiting until a new transition is invoked.

Finishing Flow

- A transition navigates flow to a final step. More then one step can be marked as final.
- Flow cannot continue any more. If you invoke a transition on finished flow (flow in a final state), the FlowAlreadyFinishedException is thrown.

Running Transition

1. Invoke a transition by invokeTransition method
2. Flow is shifted to the target state of the transition
3. If the the new state is a virtual state, it is processed and continue to step 1
4. Flow is waiting in a real state until a new transition is invoked

Processing Virtual States

Flow cannot stand by a virtual states. So a virtual state is ever processed to shift flow to a next state. Of course if it is a virtual state again, the process repeats until a real state is reached.

1. Shift flow to (or start flow in) a virtual state
2. Process the virtual state and find a transition to shift flow the a next state
3. Shift flow to the next state and continue checks

Infinite Loops

- Flow can possibly has infinite loop(s).
- It cannot be prevent during parsing XML
- Max allowed flow steps can prevent flow to cycle inside, it is recommended to set that property to a real (big) value, not null that means unfinite.
- Virtual states are a danger of cycling the flow.

Processing Listeners

There can be defined more listener those are invoked in the following order>

1. onFlowInitialized (invoked just once after initializing flow)
2. onTransitionStart (invoked on each transition)
3. onStateLeaving (invoked on each state)
4. createStateData (invoked on state that requires state data)
5. onStateEntry (invoked on each state)
6. persistFlowState (invoked only on presistable states)
7. onTransitionFinished (invoked on each transition)
8. onFlowFinished (invoked just once after reaching a final state)