

1. Context CommWorkFlow

inv: (Self.nodes.getCommProcNodes() >0)
and (Self.allInstances()->forAll(wf1,wf2 | wf1 <> wf2 **implies** wf1.workflowID <> wf2.workflowID))
and (Self.nodes->forAll(node| not node.ocllsTypeOf(DecisionNode) **implies** node.outgoingEdge->forAll(e| e.edgeType== edgeType::regularEdge))
and (Self.nodes->select(n|n.ocllsTypeOf(DecisionNode))->size() == Self.nodes->select(n|n.ocllsTypeOf(MergeNode))->size())
and (Self.nodes->select(n|n.ocllsTypeOf(ForkNode))->size() == Self.nodes->select(n|n.ocllsTypeOf(JoinNode))->size())
and (**not** (self.nodes->exists (n1,n2...np| n1. outgoingEdge== n2.incomingEdge **and** n2. outgoingEdge== n3.incomingEdge **and** n3. outgoingEdge== n4.incomingEdge **and** np. outgoingEdge== n1.incomingEdge)))

2.Context WF-Node

Inv: (Self.allInstances()->forAll(n1,n2 | n1 <> n2 **implies** n1.nodeID <> n2.nodeID))

3.Context WF-Edge

Inv: (Self.allInstances()->forAll(e1,e2 | e1 <> e2 **implies** e1.edgeID <> e2.edgeID))
and (Self.edgeType == edgeType::DecisionEdge **implies not** (Self.annotation.ocllsTypeOf(OclVoid)))
and (Self.edgeType == edgeType::RegularEdge **implies** Self.annotation.ocllsTypeOf(OclVoid))

4. Context InitialNode

inv: (Self.outgoingEdge->size()==1)
and (Self.incomineEdge->size()==0)

5. Context FinalNode

inv: (Self.outgoingEdge->size()==0)
and (Self.incomineEdge->size()==1)

6. Context DecisionNode

inv: (Self.outgoingEdge->size() >1)
and (Self.incomingEdge->size()==1)
and (Self.outgoingEdge->forAll(edge|edge.edgeType == edgeType::DecisionEdge)
and (Self.outgoingEdge->select(edge| edge. annotation.isElse == true)-> size() ==1)
and (Self.outgoingEdge-> forAll(edge| Self.incomingEdge.source.triggerEvent.one.comProcEvent->exists(ce| ce->exists(ae | ae == edge. annotation. guardedEvent))

7. Context MergeNode

inv: (Self.outgoingEdge->size() == 1)
and (Self.incomingEdge->size() > 1)

8. Context ForkNode

inv: (Self.outgoingEdge->size() >1)
and (Self.incomingEdge->size()==1)

9. Context JoinNode

inv: (Self.outgoingEdge->size() ==1)
and (Self.incomingEdge->size() >1)
and (Self.incomingEdges->forAll(edge| edge.edgeType == edgeType::RegularEdge)

10.Context AtomicEvent

Inv: (Self.allInstances()->forAll(e1,e2 | e1 <> e2 **implies** e1.eventID <> e2.eventID))
and self.isTypeOf(MediumEvent) **implies** self.mediumType->exists(property | property == Self.guard.key)
and self.isTypeOf(MediumEvent) **implies** self.formType->exists(property | property == Self.guard.key)
and self.isTypeOf(NegotiationEvent) **implies** self.controlSchema->exists(property | property == Self.guard.key)

11.Context MediaEvent

inv: not (Self.statusMT == Self. temporalStart)

12.Context FormEvent

inv: not (Self.statusFT == Self. temporalStart)

13.Context NegotiationEvent

inv: not (Self.status == Self. temporalStart)

14.Context CommProcEvent

Inv: (Self.allInstances()->forAll(e1,e2 | e1 <> e2 **implies** e1.eventID <> e2.eventID))

15.Context TriggerEvent

Inv: (Self.allInstances()->forAll(e1,e2 | e1 <> e2 **implies** e1.eventID <> e2.eventID))

16.Context AtomicCommProcNode

inv: (Self.triggerEvent->size()==1)
and (Self.cmlSchema.communicationID == self.triggerEvent.one.communicationID)
and (self.cmlSchema.controlSchema.connections->exists(c| c.connectionID == self.triggerEvent.one.connectionID))
and (Self. nodeID == self.triggerEvent.one.nodeID)
and (Self. containingWF.workflowID == Self.triggerEvent.one.workflowID)

17.Context CompositeCommProcNode

inv: Self.triggerEvent->size()==0