

## **Apéndice B**

---

# Apéndice B

## Diagramas UML

### B.1 Diagramas de clase



## NodoOperador

```
-activationTimes: int
-synchronizationMode: String
-politicasCalendarizacion: PoliticasCalendarizacionOp
-tipo: int
```

```
+setTipo(type: int)
+setActivationTimes(times: int)
+setSynchorinzationMode(mode: String)
+setPoliticaCalendarizacionOp( policy:
PoliticaCalendarizacionOp)
+getPoliticaCalendarizacionOp():
PoliticaCalendarizacionOp
+getActivationTimes(): int
+getSynchronizationMode(): String
+getTipo(): int
+activarOperador()
```

## NodoSeq

```
-triggeringActivity: Actividad
-triggeredActivity: Actividad
-politicasCalendar: PoliticasCalendarActs
-banderaTriggering: boolean
-banderaTriggered: boolean
-terminoEjecucionActividad: boolean
-problemasEjecucion: boolean
-actividadSuspendida: boolean
-textoInterfaz: TextArea
-topicName: String
-topicName2: String
-topicName3: String
-jndiContext: Context
-topicConnectionFactory: TopicConnectionFactory
-topicConnection: TopicConnection
-topicSession: TopicSession
-topic: Topic
-topicSubscriber: TopicSubscriber
-topicPublisher: TopicPublisher
-message: TextMessage
-msg: Message
-mensajeRecibido: TextMessage
```

```
+setTriggeringActivity(triggering: Actividad)
+setTriggeredActivity(triggered: Actividad)
+getTriggeringActivity(): Actividad
+getTriggeredActivity(): Actividad
+activarOperador()
+triggeringActivityHandler(admin: AdministradorAgentes,
texto: TextArea, admon: AdministradorListas: admon):
boolean
+evaluarPrecondicion(act: Actividad): boolean
+precondicionFailure(act: Actividad, tiempo: int,
admon: AdministradorListas): boolean
+lanzarActividad(act: Actividad, administrador:
AdministradorAgentes, admon:
AdministradorListas):boolean
+evaluaPostcondicion(act: Actividad): boolean
+postcondicionFailure(act: Actividad, ad:
AdministradorListas): boolean
+mandarMensaje(tipo: int)
+mandarMensajeNodoWf(tipo: int)
+esperoMensaje()
+triggeredActivityHandler(admin:AdministradorAgentes,
texto: TextArea, admon: AdministradorListas): boolean
```

## NodoOrSplit

```
-nombreActLanzadas: Vector
-triggeringActivity: Actividad
-triggeredActivity: Vector
-politicasCalendar: PoliticasCalendarActs
-banderaTriggering: boolean
-banderaTriggered: boolean
-terminoEjecucionActividad: boolean
-problemasEjecucion: boolean
-actividadSuspendida: boolean
-terminoEjecucionSub: boolean
-textoInterfaz:TextArea
-topicName: String
-topicName2: String
-topicName3: String
-jndiContext: Context
-topicConnectionFactory: TopicConnectionFactory
-topicConnection: TopicConnection
-topicSession: TopicSession
-topic: Topic
-topicSubscriber: TopicSubscriber
-topicPublisher: TopicPublisher
-message: TextMessage
-msg: Message
-mensajeRecibido: TextMessage
```

```
+setNombreAcividadesLanzadas(nombres: Vector)
+setTriggeringActivity(triggering: Actividad)
+setTriggeredActivity(acts: Vector)
+agregarTriggered(objeto: Object)
+agregarTriggered(indice: int, objeto: Object)
+eliminarTriggered(indice: int)
+getTriggeringActivity(): Actividad
+getTriggeredActivity(): Vector
+getActividadesLanzadas(): Vector
+obtenerRama(): Vector
+activarOperador()
+triggeringActivityHandler(admin:
AdministradorAgentes,
texto: TextArea, admon: AdministradorListas: admon):
boolean
+evaluarPrecondicion(act: Actividad): boolean
+precondicionFailure(act: Actividad, tiempo: int,
admon: AdministradorListas): boolean
+lanzarActividad(act: Actividad, administrador:
AdministradorAgentes, admon:
AdministradorListas):boolean
+evaluaPostcondicion(act: Actividad): boolean
+postcondicionFailure(act: Actividad, ad:
AdministradorListas): boolean
+mandarMensaje(tipo: int)
+mandarMensajeNodoWf(tipo: int)
+esperoMensaje()
+triggeredActivityHandler(admin:AdministradorAgentes,
texto: TextArea, admon: AdministradorListas): boolean
+lanzarActividadTriggered(act: Actividad, admin:
AdministradorAgentes, admon: AdministradorListas)
+lanzarSubWorkflow(nodo: NodoWf)
```

## NodoOrJoin

```
-nombreActLanzadoras: Vector
-triggeredActivity: Actividad
-triggeringActivity: Actividad
-politicasCalendar: PoliticasCalendarActs
-banderaTriggering: boolean
-banderaTriggered: boolean
-terminoEjecucionActividad: boolean
-problemasEjecucion: boolean
-actividadSuspendida: boolean
-textoInterfaz: TextArea
-topicName: String
-topicName2: String
-topicName3: String
-jndiContext: Context
-topicConnectionFactory: TopicConnectionFactory
-topicConnection: TopicConnection
-topicSession: TopicSession
-topic: Topic
-topicSubscriber: TopicSubscriber
-topicPublisher: TopicPublisher
-message: TextMessage
-msg: Message
-mensajeRecibido: TextMessage
```

```
+setNombreActividadesLanzadoras(nombres: Vector)
+setTriggeringActivity(acts: Vector)
+setTriggeredActivity(triggered: Actividad)
+agregarTriggering(indice: int, objeto: Object)
+agregarTriggering(objeto: Object)
+eliminarTriggering(indice: int)
+getNombreActividadesLanzadoras(): Vector
+getTriggeredActivity(): Actividad
+activarOperador()
+triggeringActivityHandler(): boolean
+evaluaPrecondicion(act: Actividad): boolean
+precondicionFailure(act: Actividad, tiempo: int,
admon: AdministradorListas): boolean
+lanzarActividad(act: Actividad, administrador:
AdministradorAgentes, admon:
AdministradorListas):boolean
+evaluaPostcondicion(act: Actividad): boolean
+postcondicionFailure(act: Actividad, admon:
AdministradorListas): boolean
+mandarMensajes(tipo: int)
+mandarMensajeNodoWf(tipo: int)
+esperoMensaje()
+triggeredActivityHandler(admin:
AdministradorAgentes, texto: TextArea, admon:
```

## PoliticaCalendarizacionOp

```
-tiempoMaximoActivacion: int  
-matching: int
```

```
+setTiempoMaximoActivacion(tiempo: int )  
+setMatching(match: int)  
+getTiempoMaximoActivacion(): int  
+getMatching(): int
```

## Grafo

```
#raiz: NodoWf  
-NumeroOperadores: int  
-temporalNum: int  
-maximoActividades: int  
-triggeringActivity: Object [ ]  
-temporalTriggering: Object [ ]  
-numeroTriggeringActivities: int  
-estado: int  
-temporal2: NodoWf  
-nombreWf: String  
-matching: int  
-politicas: PoliticasCalendarizacionOp  
-workflowTerra: Workflow  
-flujos: Vector  
-agentes: Vector  
-actividadesWorkflow: Vector  
-db1: Database  
-txn1: Transaction  
-query: OQLQuery  
-result: Object
```

```
+estado(): int  
+getNodeInicial(): NodoWf  
+comenzarConstruccion(wf: NodoWf): NodoWf  
+obtenerFlujo()  
+cerrarTransaccion()  
+obtenerAgentes()  
+agregarOperador(grafo: NodoWf): NodoWf  
+construirGrafo(grafo: NodoWf, flujoConstruir: Flow):  
NodoWf  
+construirSubWorkflow( grafo: NodoWf, sub: Flow,  
condicion: boolean): NodoWf  
+construirOrSinRamas(grafoWorkflow: NodoWf,  
flujoSinRamas: Flow): NodoWf  
+recuperarMatching()  
+recuperarActividad(actividadFlujo: Activity):  
Actividad  
+dibujaGrafo(g: Graphics, grafo: NodoWf, coordenadaX:  
int, coordenadaY: int)  
+dibujaNodo(g: Graphics, coordenadaX: int,  
coordenadaY: int, ancho: int, etiqueta: String)  
+dibujaSubGrafo(g: Graphics, grafo: NodoWf,  
coordenadaX: int, coordenadaY: int)  
+ejecutar(grafo: NodoWf, texto: TextArea): boolean  
#vacioNodo(a: NodoWf): boolean
```

## Actividad

```
-nombreActividad: String
-nombreWorkflow: String
-nombrePrecondicion: String
-nombrePostcondicion: String
-nombreInvariante: String
-politicasCalendarizacion: PoliticasCalendarActs
-politicasAsignacion: PoliticasAsignacion
-estado: int
```

```
+setNombreActividad(nombre: String)
+setNombreWorkflow(nombre: String )
+setNombrePrecondicion(precondicion: String)
+setNombrePostcondicion(postcondicion: String)
+setNombreInvariante(invariante: String)
+setPoliticasCalendarizacion(politicas:
PoliticasCalendarActs)
+setPoliticasAsignacion(politicas:PoliticasAsignacion
)
+setEstado(est: int)
+getNombreActividad(): String
+getNombreWorkflow(): String
+getNombrePrecondicion(): String
+getNombrePostcondicion(): String
+getNombreInvariante(): String
+getPoliticasCalendarizacion():PoliticasCalendarActs
+getPoliticasAsignacion():PoliticasAsignacion
+getEstado(): int
```

## PoliticasCalendarActs

```
-nombreActividad: String
-nombreWorkflow: String
-triggeringInstant: int
-preconditionEvaluationInstant: int
-postconditionEvaluationInstant: int
-preconditionFailure: int
-postconditionFailure: int
-persistenceScope: int
-persistenceScopeExceed: int
```

```
+setNombreActividad( nombre: String)
+setNombreWorkflow(nombre: String )
+setTriggeringInstant(instant: int)
+setPreconditionEvaluation( instant: int)
+setPostconditionEvaluation( instant: int)
+setPreconditionFailure( action: int)
+setPostconditionFailure( action: int)
+setPersistenceScope( scope: int)
+setPersistenceScopeExceed( action: int)
+getNombreActividad( ): String
+getNombreWorkflow( ): String
+getTriggeringInstant( ): int
+getPreconditionEvaluation( ): int
+getPostconditionEvaluation( ): int
+getPreconditionFailure( ):int
+getPostconditionFailure( ):int
+getPersistenceScope( ):int
+getPersistenceScopeExceed( ):int
```

<b>PolíticasAsignacion</b>
<pre> -nombreActividad: String -selectionCriteria: int -filtering: int </pre>
<pre> +setActividad( act: String) +setSelectionCriteria(criteria: int ) +setFiltering (filter: int) +getActividad( ): String +getSelectionCriteria( ): int +getFiltering( ): int </pre>

<b>AdministradorAgentes</b>
<pre> -politicas: PolíticasAsignacion -selectionCriteria: int -filtering: int -administradorListas: AdministradorListas -agentesDisponibles: Vector -selectedValue: Object -possibleValues: Object [ ] -possibleRolValues: Object [ ] </pre>
<pre> +inicializaActividades() +obtenRoles() +recorrerValores(p1: int, p2: int) +buscarAgente(nombreRol: String, actividad: Actividad) +obtenerPolíticasAsignacion(actividad: Actividad) +asignacionActividad(actividad: Actividad) </pre>



## AdministradorListas

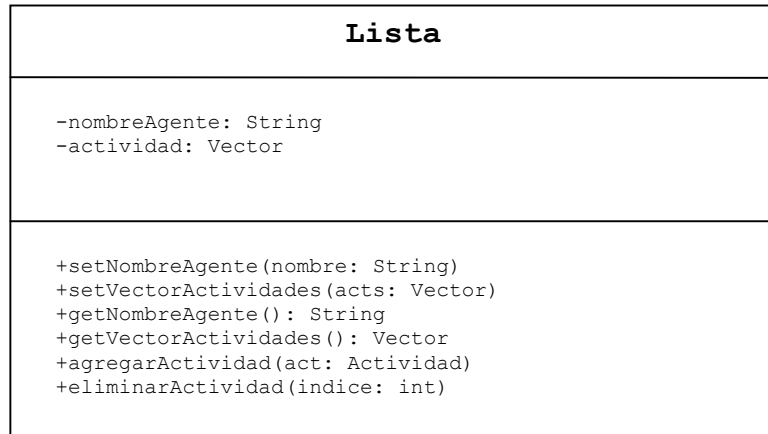
```
-activitySelection: int
-listas: Vector
-bandera: boolean
-pararEjecucion: boolean
-mandarExcepcion: boolean
-excepcion: boolean
-posicion: int
-topicName: String
-topicName2: String
-jndiContext: Context
-topicConnectionFactory: TopicConnectionFactory
-topicConnection: TopicConnection
-topicSession: TopicSession
-topic: Topic
-topicSubscriber: TopicSubscriber
-topicPublisher: TopicPublisher
-message: TextMessage
-msg: Message
-mensajeRecibido: TextMessage
```

```
+setActivitySelection(seleccion: int)
+setAgentes(agentes: Vector)
+getActivitySelection(): int
+agregarActividad(actividad: Actividad, agente:
String)
+asignarActividadAgente(listaAgente: Vector)
+ejecutarActividad(act: Actividad)
+mandarMensaje()
+esperoMensaje()
+realizoAccion()
```

## Agente

```
-nombreAgente: String
-puesto: String
-rolesAgente: Vector
```

```
+setNombreAgente(nombre: String)
+setPuesto(post: String)
+setRoles(roles: Vector)
+getNombreAgente(): String
+getPuesto(): String
+getRoles(): Vector
+buscarRol(nombreRol: String): boolean
```



## B.2 Casos de uso

