



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**FLUJOS DE TRABAJO CASO PRÁCTICO:
PROCESOS ACADÉMICOS DE LA FACULTAD DE INGENIERÍA
DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
UTILIZANDO ORACLE WORKFLOW**

Cristian Eduardo Lavarreda Estrada

Asesorado por el Ing. Luis Alberto Vettorazzi España

Guatemala, junio de 2006

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**FLUJOS DE TRABAJO CASO PRÁCTICO:
PROCESOS ACADÉMICOS DE LA FACULTAD DE INGENIERÍA
DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
UTILIZANDO ORACLE WORKFLOW**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA
POR

CRISTIAN EDUARDO LAVARREDA ESTRADA
ASESORADO POR EL ING. LUIS ALBERTO VETTORAZZI
ESPAÑA

AL CONFERÍRSELE EL TÍTULO DE
INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, JUNIO DE 2006

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympto Paiz Recinos
VOCAL I	
VOCAL II	Lic. Amahán Sánchez Álvarez
VOCAL III	Ing. Julio David Galicia Celada
VOCAL IV	Br. Kenneth Issur Estrada Ruiz
VOCAL V	Br. Elisa Yazminda Vides Leiva
SECRETARIA	Inga. Marcia Ivonne Véliz Vargas

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Herbert Rene Miranda Barrios
EXAMINADOR	Ing. Jorge Luis Alvarez Mejía
EXAMINADOR	Ing. Franklin Antonio Barrientos Luna
EXAMINADOR	Ing. Ricardo Alfredo Girón Solorzano
SECRETARIA	Ing. Gilda Marina Castellanos Baiza de Illescas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

**FLUJOS DE TRABAJO CASO PRÁCTICO:
PROCESOS ACADEMICOS DE LA FACULTAD DE INGENIERÍA
DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
UTILIZANDO ORACLE WORKFLOW,**

tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, febrero de 2005.

Cristian Eduardo Lavarreda Estrada

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	VI
GLOSARIO	X
RESUMEN.....	XX
OBJETIVOS.....	XXII
INTRODUCCIÓN	XXIV

1. FLUJOS DE TRABAJO.....	1
1.1 Introducción a los flujos de trabajo	1
1.1.1 Ventajas de utilizar flujos de trabajo	1
1.1.1.1 Reducción de costos	2
1.1.1.2 Calidad de los procesos	2
1.1.1.3 Estandarización de procesos.....	2
1.1.1.4 Control de los Procesos (<i>process management</i>).....	2
1.1.1.5 Asignación de tareas.....	3
1.1.1.6 Asignación de recursos.....	3
1.1.1.7 Diseño de procesos.....	4
1.1.2 Orquestación de procesos	4
1.1.3 <i>Workflow Management Coalition (WfMC)</i>	6
1.1.4 La necesidad de estandarizar	7
1.1.5 <i>Workflow</i> adaptable.....	8
1.1.6 <i>Workflow Management Systems (WFMS)</i>	9
1.1.6.1 Las funciones de tiempo de diseño.....	10
1.1.6.2 Las funciones de tiempo de ejecución.....	10
1.1.6.3 Las interacciones	11
1.1.7 Arquitectura de un <i>workflow management system</i>	12
1.1.7.1 Motor de ejecución del <i>workflow (workflow engine)</i>	13

1.1.7.2	Herramientas de definición de procesos (interfase 1)	16
1.1.7.3	Aplicaciones clientes (interfase 2)	17
1.1.7.4	Aplicaciones invocadas (interfase 3)	18
1.1.7.5	Funciones de interoperabilidad WAPI (interfase 4)	20
1.1.7.6	Herramientas de administración y monitoreo (interfase 5)	20
1.2	TECNOLOGÍA <i>WORKFLOW</i>	21
1.2.1	Conceptos	21
1.2.1.1	Actividad o Tarea	22
1.2.1.2	Personas (usuarios)	22
1.2.1.3	Roles	23
1.2.1.4	Rutas	23
1.2.1.5	Procesos	24
1.2.1.6	Condiciones de transición	24
1.2.1.7	Agente	27
1.2.1.8	Comportamiento	27
1.2.1.9	Eventos	28
1.2.1.10	Plazos (<i>Timeouts</i>)	28
2.	<i>ORACLE WORKFLOW</i>	29
2.1	ARQUITECTURA DE <i>ORACLE WORKFLOW</i>	29
2.1.1	Cliente de Desarrollo de <i>workflow</i>	30
2.1.2	<i>Oracle Workflow Builder</i>	32
2.1.2.1	El navegador	32
2.1.2.2	Diagramador de procesos	48
2.1.3	<i>Oracle server</i>	56
2.1.3.1	Servicio de directorio	56
2.1.3.2	Sistema de manejo de eventos	58
2.1.3.3	Servicio de notificaciones	60
2.1.4	Servidor de aplicaciones	61

2.1.4.1	Http server	61
2.1.4.2	<i>Oracle Content Management</i> SDK (CMSDK).....	62
2.1.4.3	Notificador de Correo.....	62
2.1.5	Usuarios finales	62
2.1.5.1	Listas de Trabajo	63
2.1.5.2	Monitor de Procesos.....	65
3.	CASO PRÁCTICO: PROCESOS ADMINISTRATIVOS DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA.....	69
3.1	Metodología utilizada.....	70
3.1.1	Análisis de requerimientos	71
3.1.1.1	Identificación y descripción de las necesidades de información.....	71
3.1.1.2	Identificación de los usuarios y roles involucrados.....	74
3.1.1.3	Tablas de mapeo	77
3.1.2	Diseño de los procesos.....	80
3.1.2.1	Diseño del proceso: Revisión de Examen	81
3.1.2.2	Diseño del proceso: Revisión de Tesis.....	87
3.1.2.3	Diseño del proceso de solicitud de catedrático.....	94
3.2	Resultados esperados.....	97
3.2.1	Resultados de la automatización del proceso de revisión de examen.....	97
3.2.2	Resultados de la automatización del proceso de revisión de tesis.....	98
3.2.3	Resultados de la automatización del proceso de revisión de tesis.....	99
	CONCLUSIONES.....	101
	RECOMENDACIONES.....	103
	BIBLIOGRAFÍA.....	105

ÍNDICE DE ILUSTRACIONES

FIGURAS

1	Flujo de trabajo de la aprobación de un crédito	5
2	Funciones de un WFMS al más alto nivel	10
3	Arquitectura general de los WFMS	13
4	Estados básicos dentro de un esquema de transición para un proceso	14
5	Herramientas de definición de procesos (interfase 1)	17
6	Esquema de los componentes de las aplicaciones cliente (interfase 2)	18
7	Esquema de los componentes de las aplicaciones invocadas (interfase 3)	19
8	Interfase de interoperabilidad del <i>workflow</i> (interfase 4)	20
9	Herramientas de administración y monitoreo (interfase 5)	21
10	Construcción <i>AND-Split</i>	25
11	Construcción <i>AND-Join</i>	25
12	Construcción <i>OR-Split</i>	26
13	Construcción <i>OR-Join</i> .	26
14	Construcción ciclo o iteración	27
15	Arquitectura de <i>oracle workflow</i>	30
16	Componentes del cliente de desarrollo	31
17	El navegador de <i>oracle workflow builder</i>	33
18	Propiedades de un atributo	35
19	Diagramador de procesos con subprocessos	38
20	Propiedades de una notificación	39
21	Propiedades de una función	42

22	Propiedades de un mensaje	45
23	Resultado de un mensaje	46
24	Propiedades de una lista de valores	47
25	Propiedades del código de una lista de valores	48
26	Diagrama de procesos	51
27	Diagrama de proceso de una votación	54
28	Propiedades de una notificación para votación	55
29	Lista de trabajo	63
30	Detalle de una notificación	64
31	El monitor de instancias de procesos	67
32	Búsqueda de instancias de procesos	68
36	Diagrama del proceso de revisión de examen	86
37	Diagrama del proceso de revisión de tesis	93
38	Diagrama del proceso “solicitud del catedrático a los estudiantes”	97

TABLAS

I	Roles del proceso de revisión de examen	75
II	Roles del proceso de revisión de tesis	76
III	Roles del proceso de solicitud del catedrático a los estudiantes	76
IV	Mapeo del proceso de revisión de examen	77
V	Mapeo del proceso de revisión de tesis	78
VI	Mapeo del proceso de solicitud del catedrático a los estudiantes	80
VII	Atributos del proceso de revisión de examen	81
VIII	Mensajes del proceso de revisión de examen	82
IX	Notificaciones del proceso de revisión de examen	84
X	Atributos del proceso de revisión de tesis	87
XI	Mensajes del proceso de revisión de tesis	88
XII	Notificaciones del proceso de revisión de tesis	92
XIII	Atributos del proceso de revisión de tesis	94
XIV	Mensajes del proceso de solicitud del catedrático a los estudiantes	95
XV	Notificaciones del proceso de solicitud del catedrático a los estudiantes	96

GLOSARIO

Actividad de proceso	Es un paso lógico o descripción de un trabajo que contribuye a realizar el objetivo del proceso. Una actividad de proceso puede ser manual o automática.
Actividad de <i>workflow</i>	La automatización de un paso lógico que contribuye a realizar el objetivo del proceso.
Actividad manual	Una paso manual que contribuye a realizar el objetivo del proceso. Este paso, por su naturaleza, no puede ser automatizado.
<i>Ad hoc routing</i>	Es la habilidad de definir las transiciones del <i>workflow</i> en tiempo de ejecución. En este caso la construcción de las transiciones del <i>workflow</i> se basa en la información ingresada por el usuario en tiempo real.
AppleTalk Filing Protocol (AFP)	Es el protocolo, en ambientes Macintosh, el cual permite a los usuarios acceder a archivos en otros sistemas Macintosh.

Análisis	La parte del proceso de desarrollo de software, cuyo propósito principal es realizar un modelo del problema. El análisis se enfoca en qué hacer mientras que el diseño en cómo hacerlo.
<i>AND-join</i>	Dos o más actividades que convergen a un punto de control común dentro del proceso.
<i>AND-split</i>	Es el punto dentro de un camino del proceso donde se abre el flujo en varios caminos a seguir simultáneamente.
Aplicaciones empaquetadas	Son aplicaciones desarrolladas de forma muy genérica y que permiten parametrizaciones para adecuarse a cada empresa. Las empresas las prefieren, ya que, han sido, ampliamente, probadas y su costo es menor que el de una aplicación similar desarrollada internamente en la organización.
<i>Application program interface (API)</i>	Es un conjunto de llamadas a funciones de software los cuales les permiten a otras aplicaciones utilizar las funciones de un programa.

Arquitectura

La estructura organizacional de un sistema. Una arquitectura puede ser descompuesta recursivamente en: partes que interactúan entre si por medio de interfases, relaciones que conectan las partes y restricciones para ensamblar las partes.

Base de datos

Datos que han sido organizados y estructurados de acuerdo con los estándares aceptados para que el acceso a la información sea tan rápido como sea posible.

***Binary large object*
(BLOB)**

Es un tipo de dato soportador por *Oracle* capaz de almacenar hasta 4 gigabytes de información binaria.

***Business process reengineering* (BPR)**

El acto de analizar un proceso de empresa existente, juzgando su conveniencia para el ambiente existente en el negocio y cambiar este proceso radicalmente. BPR y *Workflow* pueden ser llevados a cabo, independientemente uno del otro.

***Character large object*
(CLOB)**

Es un tipo de dato soportado por *Oracle* capaz de almacenar hasta 4 gigabytes de información de tipo texto.

***Common object request broker* (CORBA)**

Arquitectura que permite a dos sistemas heterogéneos compartir información. Es un estándar de facto en Internet.

**Content management
Systems (CMS)**

Sistemas de administración de contenido, los cuales permiten publicar, editar y eliminar documentos a través del web. Permite, además, la organización, categorización y estructuración de recursos de información como texto, imágenes y otros tipos de documentos.

**Cuello de botella o
bottleneck**

Dentro de un proceso se dice de una actividad cuya limitada capacidad reduce la capacidad del proceso completo.

Diagrama

La representación gráfica de una colección de elementos del modelo, frecuentemente plasmada como un grafo conectado de arcos o relaciones y vértices.

Diseño

La parte del proceso de desarrollo de un *workflow*, cuyo propósito principal es decidir cómo se construirá el sistema. Durante el diseño se toman decisiones estratégicas y tácticas para alcanzar los requerimientos funcionales y la calidad esperada.

**File Transfer Protocol
(FTP)**

Es un sistema de transferencia de archivos a través de TPC-IP y que se ha extendido para ser utilizado en Internet.

Internet Message Access Protocol versión 4 (IMAP4)	Es un protocolo de internet que permite a los clients manipular mensajes de correo electrónico almacenados en un servidor de correo.
Implementación	Crear la definición de un proceso dentro de una base de datos. Implica la creación de código de programación. Un mismo proceso puede tener múltiples implementaciones.
Instanciar un proceso	Implementación de un proceso dentro de un sistema para que pueda ser utilizado en producción por usuarios finales.
Interacción	Una especificación de comportamiento cuyo fin es lograr un propósito específico. Abarca un conjunto de intercambios de mensajes entre un conjunto de objetos dentro de un contexto particular.
<i>Legacy systems</i>	Es un sistema o aplicación que continúa siendo utilizado debido a que los usuarios o diseñadores no quieren reemplazarlo o rediseñarlo por los costos de tiempo y dinero que implica. La mayoría de veces son sistemas antiguos y potencialmente peligrosos, ya que, normalmente no se cuenta con la experiencia para mejorarlos o corregirlos.

***Lightweight directory
access protocol (LDAP)***

Es un estándar de servicio de directorio, proporcionando una manera simple de almacenar y consultar información de usuarios y grupos de usuarios. Implementa un subconjunto del estándar X.500.

***Java 2 Enterprise Edition
(J2EE).***

Es un entorno para desarrollar y publicar aplicaciones. Consiste en una serie de servicios, interfases de programación (API's) y protocolos que proporcionan la funcionalidad requerida de las aplicaciones para el web.

Metamodelo

Un modelo que define el lenguaje para expresar un modelo.

Modelo

Una abstracción semánticamente consistente de un sistema.

**Network File Systems
(NFS)**

Es un sistema de archivos distribuido que se utiliza en ambiente cliente servidor para compartir archivos en red en forma transparente.

Oracle

Empresa de software que comercializa las herramientas de software que se utilizaron en la presente tesis. También, es el nombre de la base de datos de la misma empresa.

Orquestación de procesos	Es la abstracción que permite al programador constituir múltiples servicios web en un flujo continuo de pasos y que soporta interacciones asíncronas, flujos de control y compensación de transacciones distribuidas.
Procesos de negocios	Es un conjunto de uno o más procedimientos o actividades directamente ligadas, que, colectivamente, realizan un objetivo del negocio, normalmente dentro del contexto de una estructura organizacional que define roles funcionales y relaciones entre los mismos. También, llamados procesos de empresa, <i>workflow</i> , procesos de <i>workflow</i> o <i>workflow</i> del negocio.
Rol	El nombre que identifica al comportamiento específico de una entidad, cuando ésta participa de un contexto particular. Para una empresa es la responsabilidad asignada a un individuo en una organización o un grupo.
Sistema	Una colección de unidades conectadas entre sí, que están organizadas para llevar a cabo un propósito específico. Un sistema puede describirse mediante uno o más modelos, posiblemente desde puntos de vista diferentes.

Server Message Bloks (SMB)	Es un protocolo de alto nivel para compartir archivos e intercambiar información dentro de un ambiente de red Microsoft.
Simple Mail Transfer Protocol (SMTP)	Es un protocolo que controla el intercambio de mensajes de correo electrónico entre dos servidores de correo.
Tecnología	Sistematización de los conocimientos y prácticas aplicables a cualquier actividad.
Web Distributed Authoring and Versioning (WebDAV)	Es un protocolo que permite la edición compartida de archivos a través de http. Proporciona servicios de control de versiones.
<i>Workflow</i>	Flujo de trabajo. Un workflow especifica las actividades que constituyen un proceso, el orden en que se ejecutan, el flujo de datos entre las actividades y los agentes que las realizan y que colaboran para ejecutarlo en su totalidad. Consiste en una serie de actividades, fuertemente, ligadas y que tienen la característica de que cada actividad determina la siguiente a realizar.

Workflow management coalition (WfMC)

Es una organización internacional no lucrativa que agrupa a vendedores de aplicaciones workflow, usuarios, analistas, universidades y grupos de investigadores. Con el fin de estimular y desarrollar el uso de workflow a través de establecer estándares de desarrollo, terminología, interoperabilidad y conectividad entre los diferentes productos de workflow.

Workflow management system (WFMS)

Es un sistema que permite definir, instalar y ejecutar procesos de flujos de trabajo o workflows de una manera adaptable.

Tarea

Es la unidad lógica de todo flujo de trabajo. Es atómica y por esta razón debe ser llevada a cabo, completamente, y no puede dividirse en partes más pequeñas.

RESUMEN

Un flujo de trabajo o *workflow* es la representación computarizada de un proceso de negocios, el cual especifica las diferentes actividades que forman dicho proceso, el orden en que éstas deben ejecutarse, el flujo de datos entre ellas y los múltiples agentes que colaboran para llevar a cabo el proceso completo.

Todas las empresas o instituciones reconocen que lo único inmutable en sus organizaciones es el cambio. Existen cambios legislativos, económicos, políticos, tratados comerciales o bien innovaciones en los procesos de las organizaciones. Ante este panorama es necesario revisar, continuamente, los procesos internos de cada organización para lograr adaptarlos a estos continuos cambios. Es evidente que el contar con una herramienta que nos permita realizar estos cambios rápida y, eficientemente, proporciona grandes beneficios a las organizaciones que lo emplean.

Para ilustrar mejor éste punto se han modelado, diseñado e implementado tres procesos administrativos en la facultad de Ingeniería. Utilizando una de las herramientas más poderosas y populares del mercado: Oracle Workflow, la cuál cumple con el estándar definido por la *Workflow management coalition (WfMC)* la cuál aglutina más de trescientas organizaciones, empresas y universidades.

El primer proceso modelado fue el de revisión de examen, en el cual el estudiante inicia una solicitud para que el catedrático de su curso revise su nota explicando las razones por las que considera que se debe cambiar la nota. El coordinador del área y el director de la escuela monitorean y reciben notificaciones sobre el avance de cada proceso.

El segundo proceso es el de la revisión de tesis donde existen varios actores involucrados: el estudiante, la comisión de revisión de privados y tesis, el asesor, el revisor y el director de escuela. El proceso modela todos los pasos desde que el estudiante envía su protocolo de tesis a la comisión hasta que el revisor termina la corrección. Este flujo ejemplifica las transiciones condicionadas, las iteraciones y notificaciones que se dan en flujos de trabajo complejos.

Y, por último, el proceso de solicitud del catedrático donde el catedrático somete a votación una iniciativa que debe ser aprobada por la mayoría de los estudiantes. Este proceso se eligió para ejemplificar el uso de funciones externas, en éste caso el conteo de la votación, dentro de un flujo de trabajo.

Estos procesos están ya a disposición de todos los estudiantes, ya que, se contó con la donación, por parte de la empresa Datum S.A. de un servidor y de las licencias de *Oracle Workflow*.

OBJETIVOS

General

Modelar, diseñar e implementar tres procesos administrativos de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala utilizando Oracle Workflow.

Específicos

1. Definir los conceptos más utilizados para diseñar procesos.
2. Mostrar las ventajas de la utilización flujos de trabajo en cualquier organización.
3. Mostrar las ventajas del uso de herramientas informáticas para el modelado, diseño e implementación de flujos de trabajo.
4. Diseñar una metodología para la implementación de flujos de trabajo.
5. Facilitar a los estudiantes de la Facultad de Ingeniería de la Universidad San Carlos de Guatemala la realización de los procesos administrativos: solicitud de revisión de examen, solicitud de catedrático y revisión de tesis.
6. Proporcionar a las autoridades de la Facultad de Ingeniería una manera de monitorear y administrar el seguimiento de los procesos administrativos.

INTRODUCCIÓN

Todas las empresas o instituciones reconocen que lo único inmutable en sus organizaciones es el cambio. Existen cambios legislativos, económicos, políticos, de hábitos o bien innovaciones en los procesos de las organizaciones. Ante este panorama es necesario revisar, continuamente, los procesos internos de cada organización para lograr adaptarlos a estos continuos cambios.

Para poder ser una organización adaptable a este ambiente cambiante, es importante contar con herramientas informáticas que permitan definir procesos y cambiarlos cuando sea necesario, de una forma fácil y segura. Esta es la función que realizan los sistemas de administración de flujos de negocios.

Sin embargo, el nuevo reto es que estos sistemas nos permitan incorporar actividades que se realizan fuera de nuestra organización y que forman parte de un proceso mayor. Con la amplia difusión de los servicios web y su coordinación, también, conocida como orquestación de procesos, se ha logrado la integración entre procesos de diferentes empresas y su coordinación.

Existen pocos productos en el mercado que cumplan con la capacidad de orquestación de procesos; uno de los que lo hacen es *Oracle Workflow*, una herramienta muy popular y que cumple con los estándares internacionales para el modelado de procesos. Es por éstas razones que se eligió como herramienta para la modelación y diseño en el presente trabajo de graduación.

En el primer capítulo se definirán los conceptos generales, la terminología utilizadas en las herramientas de administración de flujos de trabajo, los objetivos que persiguen y su arquitectura. Todo esto de acuerdo con el estándar definido por la *Workflow Management Coalition* (WfMC) que es la institución encargada de definir los estándares utilizados en la industria.

Luego en el segundo trataremos sobre la manera en que *Oracle Workflow* implementa el estándar, su arquitectura y herramientas. Y, en el tercer capítulo, modelaremos, diseñaremos e implementaremos tres procesos administrativos de la Facultad de Ingeniería de la Universidad San Carlos de Guatemala que se tratarán como un caso práctico que ilustre la manera de diseñar procesos administrativos complejos.

Por último, es importante señalar que los procesos modelados y diseñados en ésta tesis se encuentran actualmente disponibles para ser utilizados dentro de la Facultad de Ingeniería. Esto se ha logrado gracias a la donación, por parte de la empresa Datum S.A. de un servidor y de las licencias de *Oracle Workflow*.

1. FLUJOS DE TRABAJO

1.1 Introducción a los flujos de trabajo

Un flujo de trabajo o *workflow* es la representación computarizada de un proceso de negocios, el cual especifica las diferentes actividades que forman dicho proceso, el orden en que éstas deben ejecutarse, el flujo de datos entre ellas y los múltiples agentes que colaboran para llevar a cabo el proceso completo.

Un proceso de negocios típico se encuentra formado de diferentes tareas que, en conjunto, cumplen con el objetivo final del proceso. Estas tareas son ejecutadas por distintos agentes, humanos o componentes de software, que colaboran para llevarlas a cabo y en conjunto completar el proceso.

Es evidente que el contar con un sistema de *workflow* proporciona grandes beneficios a las organizaciones que lo emplean. Estos beneficios no redundan únicamente en el ahorro de tiempo en el manejo de papeles, que en un principio era uno de los grandes problemas a resolver sino que también en la optimización y mejora continua de los procesos y posterior monitoreo. Los puntos a favor del uso de la tecnología de *workflow* son muchos. A continuación se presentan algunas razones por las cuales las organizaciones deben considerar adoptar una solución de *workflow*.

1.1.1 Ventajas de utilizar flujos de trabajo

1.1.1.1 Reducción de costos

El hecho de contar con una herramienta que permita definir gráficamente los procesos, revisarlos continuamente, determinar los tiempos que toman cada una de las actividades y la capacidad de poderlos cambiar rápidamente conduce a una reducción de costos dentro de una empresa.

1.1.1.2 Calidad de los procesos

La revisión continua de los procesos nos conduce paulatinamente a tener un mayor conocimiento de los mismos, lo que a su vez conduce a obtener una mejor calidad de estos.

1.1.1.3 Estandarización de procesos

Dentro de las necesidades de las empresas, cada vez sobresale la necesidad de relacionarse con otras organizaciones. Así por ejemplo los bancos necesitan, en el marco de un proceso interno, enviar información a la superintendencia de bancos o a la superintendencia de administración tributaria y luego, dependiendo de su respuesta regresar a su proceso interno. Esto implica que deben existir interfases compatibles entre los procesos de diferentes organizaciones lo que puede lograrse a través de la estandarización de los procesos. Por otro lado la estandarización nos conduce a lograr un modelado más rápido y exacto al trabajar con nuevos procesos ya que pueden clasificarse más fácilmente dentro del estándar.

1.1.1.4 Control de los Procesos (*process management*)

Al utilizar la tecnología de *workflow* es posible monitorear el estado actual de las tareas así como también observar como evolucionan los planes de trabajo realizados y si se cumplen los indicadores clave de desempeño. A esta capacidad se le conoce

también como control de procesos en tiempo de ejecución, lo que indica que mientras los usuarios ejecutan los procesos existe un usuario administrador que es capaz de determinar en que actividad se encuentra el proceso y los valores de sus variables.

Esto nos permite ver cuales son los “cuellos de botella” dentro del sistema, es decir aquellas tareas o decisiones que están requiriendo de tiempo no planificado y se tornan en tareas o decisiones críticas. Además nos proporciona un ambiente inmejorable para proponer cambios y evaluarlos antes de llevarlos a la práctica. Muchas herramientas de *workflow* proporcionan la capacidad de simular los procesos antes de implementarlos, por lo que se pueden evaluar diferentes escenarios, cambiar una tarea del proceso y determinar cómo se comporta el proceso completo.

1.1.1.5 Asignación de tareas

Al tener definidos los procesos y los agentes que los llevan a cabo, la asignación de tareas se simplifica. La asignación de tareas se realiza mediante la definición de roles dentro de la empresa, eliminando la tediosa tarea de asignar los trabajos caso por caso.

1.1.1.6 Asignación de recursos

Para completar algunas actividades se necesita contar con ciertos recursos como acceso a datos o archivos. El tener una aplicación que centralice la asignación de recursos, nos permite asegurarnos que contaremos con los recursos en el momento que los necesitemos. Podemos monitorear el uso de cada recurso por parte de los agentes para hacer un mejor uso de los mismos.

1.1.1.7 Diseño de procesos

Las herramientas de administración de flujos de trabajo coordinan los aspectos operacionales de la ejecución de un proceso tales como la transición correcta de las actividades, proveer acceso a datos y documentos, las notificaciones a los agentes, control de tiempos, ciclos de actividades, mecanismos de control y monitoreo. Todos estos aspectos son implementados por el software de administración de flujos de procesos por lo que no es necesario programarlo y los diseñadores sólo deben concentrarse en el proceso mismo.

1.1.2 Orquestación de procesos

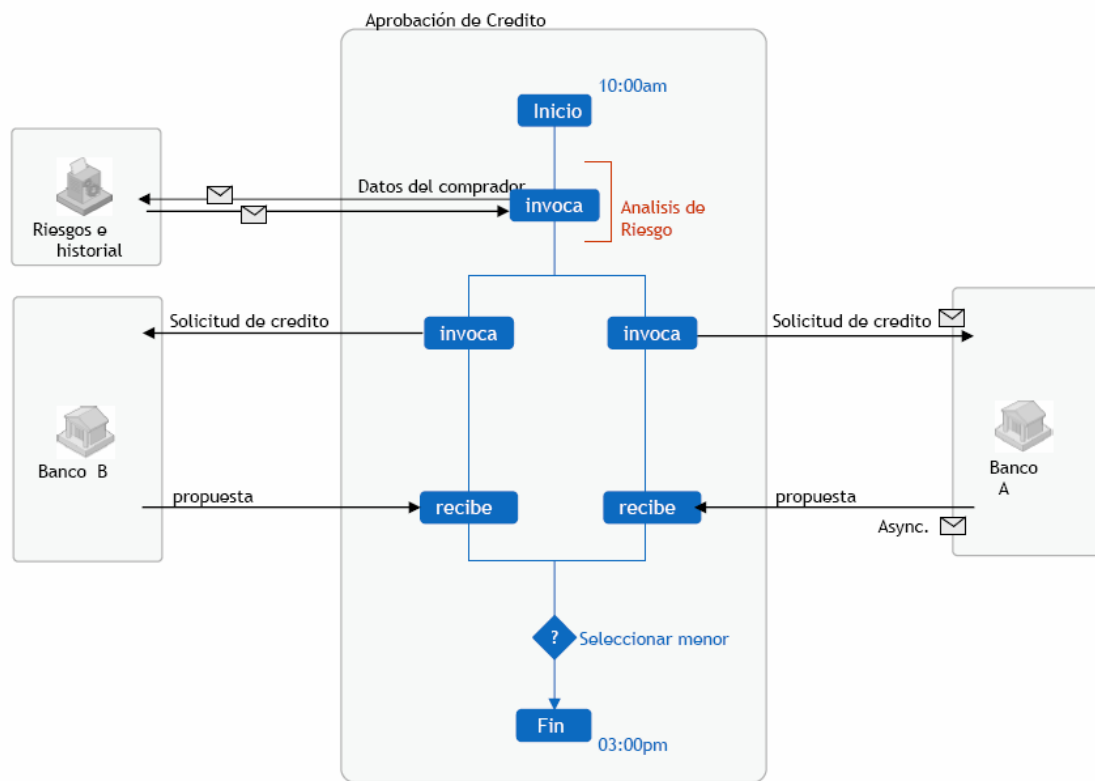
Hoy en día, la mayoría de organizaciones cuentan con “aplicaciones empaquetadas”, *legacy systems* o desarrollos internos que necesitan compartir información y funcionalidad para ofrecer una solución integral a los usuarios. Desafortunadamente, la mayoría de estos sistemas son propietarios y están escritos en distintos lenguajes de programación con diversos esquemas de datos. El escenario se vuelve aún más complejo cuando se desea integrar y compartir aplicaciones de más de una empresa.

La orquestación de procesos nos permite definir las estructuras de intercambio de información, así como las interfases de comunicación entre los procesos. Todo utilizando servicios web y contando con herramientas que nos permitan monitorear el proceso completo. Podríamos resumir la orquestación como la abstracción que permite al programador constituir múltiples servicios web en un flujo continuo de pasos y que soporta interacciones asíncronas, flujos de control y compensación de transacciones distribuidas.

Veamos por ejemplo, la compra de un automóvil nuevo en una agencia. Como la competencia es muy dura, la agencia negocia con varios bancos del sistema la

aprobación del crédito para la compra en el menor tiempo posible. Para esto los sistemas tanto de la agencia como de los bancos deben poder compartir información y la aprobación del crédito será un proceso que involucre procesos de más de una empresa. La figura 1 muestra cada una de las etapas de este proceso.

Figura 1. Flujo de trabajo de la aprobación de un crédito.



Los procesos dentro del recuadro corresponden a los que se llevan a cabo en la agencia de vehículos, así el proceso inicia a las diez de la mañana cuando un cliente desea adquirir un vehículo a crédito. El primer paso consiste en determinar el nivel de riesgo haciendo una consulta a una empresa donde nos indicarán el nivel de riesgo de cliente.

Si el nivel de riesgo es suficientemente bajo, continúa el proceso enviando en paralelo las solicitudes de crédito a dos bancos del sistema. Esta llamada se muestra como

asíncrona ya que el banco no está en línea con la agencia. El coordinador de los procesos, también llamado orquestador o director, espera a que se reciban las respuestas para continuar con el proceso. Al recibir las repuestas se elije la menor tasa y finaliza el proceso.

1.1.3 *Workflow Management Coalition (WfMC)*

Los términos que hemos utilizado hasta el momento, como proceso y actividad, poseen una acepción distinta de la que se le da en el lenguaje coloquial. Para poder tener definiciones válidas es necesario que las empresas de desarrollo de aplicaciones *workflow*, los usuarios, las empresas, universidades y otros actores involucrados en los procesos de *workflow* hayan acordado definir un estándar. La organización encargada de definir los estándares de *workflow* se denomina *Workflow Management Coalition*.

La *Workflow Management Coalition*, fundada en agosto de 1993, es una organización internacional no lucrativa que agrupa a vendedores de aplicaciones *workflow*, usuarios, analistas, universidades y grupos de investigadores.

La misión de la coalición es estimular y desarrollar el uso de *workflow* a través de establecer estándares de desarrollo, terminología, interoperabilidad y conectividad entre los diferentes productos de *workflow*. Está formado por más de 300 miembros en todo el mundo.

Muchos vendedores de software tienen productos de *workflow* enfocados en funcionalidades particulares. Sin embargo, es necesario asegurarnos que diferentes productos de *workflow* pueden, eventualmente, trabajar juntos, pues de lo contrario resultaremos teniendo islas de información incompatibles con la automatización total del proceso.

La *Workflow Management Coalition* está formada por un grupo de compañías que se han reunido para definir un estándar en el desarrollo e integración de aplicaciones de *workflow*.

Se ha determinado que todos los productos de *workflow* tienen características comunes, por lo que potencialmente se puede llegar a tener un cierto nivel de interoperabilidad utilizando un estándar común entre ellos. La *Workflow Management Coalition* ha identificado éstas áreas funcionales comunes y ha desarrollado especificaciones y estándares comunes para todos los productos de *workflow*. Esto habilita la interoperabilidad entre productos de *workflow* heterogéneos y mejora la integración de aplicaciones de *workflow* con otros servicios tecnológicos como correo electrónico y *content management systems* (CMS).

1.1.4 La necesidad de estandarizar

Se estima que actualmente los distintos productos de *workflow* que hay en el mercado sobrepasan los cien. Cada uno de ellos se enfoca sobre distintos aspectos funcionales, como son, herramientas de diseño visual, en las cuales se ofrecen ciertos diagramas para representar la realidad. Otras se enfocan en la integración de los datos con las aplicaciones. El desarrollo de estándares para la interoperabilidad de las diversas herramientas, nos permitiría la elección de los mejores productos, según el enfoque que le demos a nuestra aplicación. Por ejemplo, podríamos comprar una herramienta que nos ofrezca un entorno amigable para realizar el análisis y diseño de nuestro problema, y por otro lado comprar una componente que nos resuelva la auditoría de los datos y poder trabajar en forma integrada con las dos componentes. Una de las estrategias que siguen actualmente las empresas, es rediseñar sus procesos, esta metodología es denominada como reingeniería de los procesos de empresas o *business process re-engineering*.

Esto puede ser a causa de cambios organizacionales, legislativos, cambios en los objetivos del negocio o innovaciones. En otras ocasiones también es necesario

relacionarse con otras organizaciones. Pero para poder hacer esto debe existir la posibilidad de que los productos de un vendedor puedan comunicarse con los de otro, pues es claro que cada empresa u organización comprará los productos que crea conveniente para su caso.

Aquí se presenta entonces la necesidad de estandarizar la forma de comunicación entre los distintos componentes de un producto de *workflow*. Para poder tener flexibilidad al operar con distintos productos. Esta necesidad se justifica además por las proyecciones que se tienen actualmente, sobre la penetración de la tecnología de *workflow* en el mercado en los próximos años. Por esto se debe atacar el problema de potenciales incompatibilidades de antemano, y no cuando existan miles de productos en esta área, cada uno con sus particularidades.

1.1.5 *Workflow* adaptable

Muchos modelos y sistemas son limitados para diseñar o describir nuevos tipos de flujos de trabajo que cumplan con las necesidades de heterogeneidad y distribución de las aplicaciones. Por tal razón surge la necesidad de encontrar una forma de definir *workflows* que den propiedades de adaptabilidad a la definición de su estructura y de su ejecución. Un *workflow* adaptable es aquel que se vale de conceptos, técnicas y herramientas que lo ayudan a soportar el cambio. Un *workflow Management System* (WFMS), es un sistema que permite definir, crear una instancia del proceso y ejecutar flujos de trabajo de una manera adaptable.

Para especificar la estructura de un flujo de trabajo, deben definirse las actividades, los agentes que se encargan de ejecutarlas, el flujo de datos entre ellas, etc. Generalmente los flujos de trabajos están formados por actividades donde cada parte del proceso está representada por una actividad. La meta principal en la gestión de flujo de trabajos es

manejar estas actividades de la forma más eficiente y efectiva posible, aún en presencia de cambios.

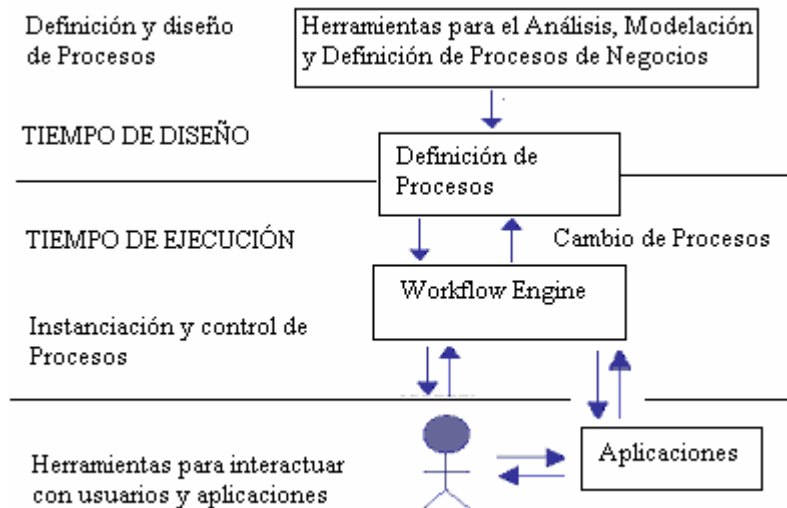
1.1.6 *Workflow Management Systems (WFMS)*

De acuerdo con la WfMC un *workflow Management System* es un conjunto de herramientas que proveen soporte a procesos de definición, almacenamiento, comportamiento, administración, ejecución y monitoreo de procesos de *workflow*. A pesar de la gran variedad de técnicas de implementación y ambientes operacionales, los WFMS presentan ciertas características en común, éstas proveen la base para el desarrollo, integración e interoperabilidad entre los diferentes productos que existen.

Un proceso de negocios individual, puede tener un ciclo de vida desde unos pocos minutos hasta meses, dependiendo de su complejidad y de la duración de las actividades que lo componen. Tales sistemas pueden ser implementados de diferentes maneras, utilizando distintas infraestructuras de comunicaciones y tecnología. El modelo de referencia de la *Workflow Management Coalition* (WfMC) describe un modelo común para la construcción de sistemas de *workflow* e identifica la manera en que están relacionadas las diferentes alternativas de enfoques.

Como se ilustra en la figura 2, al más alto nivel, todos los WFMS deben proveer soporte a tres áreas fundamentales: Tiempo de diseño, tiempo de ejecución y las interacciones entre ellos.

Figura 2. Funciones de un WFMS al más alto nivel.



1.1.6.1 Las funciones de tiempo de diseño

Se relacionan con la definición, modelado de los procesos de *workflow* y las actividades que lo conforman. Estas incluyen herramientas para facilitar el diseño de los procesos, una interfaz amigable para realizar los cambios y validar que el diseño esté completo.

1.1.6.2 Las funciones de tiempo de ejecución

Se relacionan con el manejo de los procesos en un entorno operacional y secuenciando varias actividades para manejarlas como parte de cada proceso. Estas funciones tienen que ver con los motores de *workflow* que se encargan de disparar los eventos, enviar las notificaciones, llamar a funciones automáticamente y coordinar todo el proceso. Son los intérpretes de los modelos creados en tiempo de diseño.

1.1.6.3 Las interacciones

Todas las actividades definidas dentro de los procesos, interactúan con los usuarios humanos y con aplicaciones o programas. Para esto es necesario integrar las aplicaciones ya existentes para que puedan enviar y recibir notificaciones, manejar eventos que disparen los procesos. Todo esto tiene que ser hecho a través de herramientas que faciliten crear éstas interfaces.

Tradicionalmente los WFMS son usados para apoyar procesos estáticos, es decir procesos que no cambian frecuentemente. Lo anterior ha limitado bastante el alcance de la gestión de flujo de trabajos. Nuevas tecnologías, nuevas leyes y nuevos requerimientos de mercado son los cambios que encabezan las listas de modificaciones con las que tiene que lidiar un WFMS. La inhabilidad de lidiar con estos cambios es lo que limita la aplicación de la tecnología workflow en nuestros días.

A continuación una clasificación de los cambios de los flujos de trabajo.

a. Cambios Ad-hoc

Los cambios ocurren en una sola actividad o proceso generalmente como consecuencia de un error, un evento inesperado o una demanda especial del cliente. Las excepciones con frecuencia terminan convirtiéndose en cambios ad-hoc.

b. Cambios que evolucionan

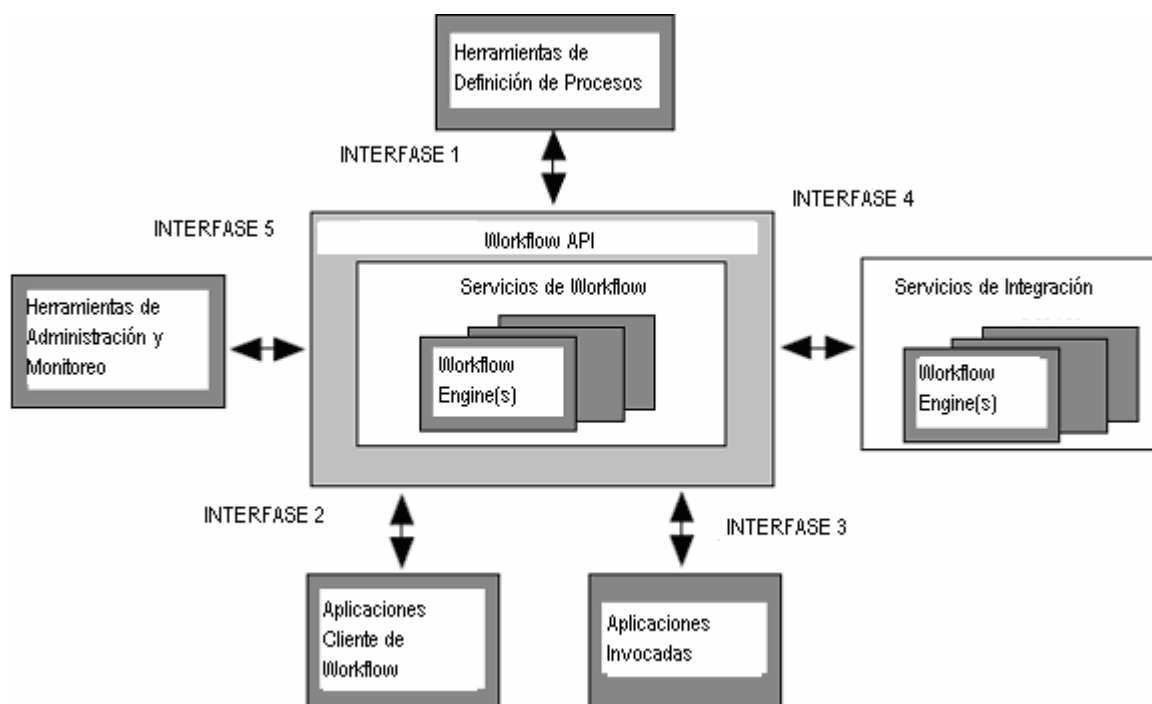
Son cambios de naturaleza estructural, por ejemplo el proceso cambia por la introducción de nuevas actividades. Este cambio es el resultado de una nueva estrategia de negocio, reingeniería o una alteración de las condiciones externas, por ejemplo la puesta en práctica de una nueva ley.

1.1.7 Arquitectura de un *workflow management system*

A pesar de la gran variedad de productos de *workflow* que se encuentran en el mercado, se puede ver que los conceptos y terminologías utilizadas no varían en gran forma. Esto permite que se tienda a realizar un modelo de implementación general. Actualmente se busca identificar los principales componentes de un sistema de *workflow*, para poder clasificarlos dentro de un mismo modelo abstracto. Es necesaria la representación formal de un modelo que permita la realización de sistemas sobre diversos escenarios, de forma tal, que distintos sistemas de *workflow* puedan interactuar entre sí.

La arquitectura general de los WfMS se desarrolló partiendo de estructuras genéricas utilizadas en aplicaciones *workflow*, con la finalidad de identificar interfases que permitan a los productos comunicarse entre los distintos niveles que existen. Todos los sistemas manejadores de *workflows* poseen componentes genéricos que interactúan entre sí de forma definida. La WfMC con el objetivo de ofrecer interoperabilidad entre los diferentes productos que existen en el mercado ha definido un conjunto de interfases y formatos que permiten el intercambio de datos entre los diversos componentes. La figura 3 ilustra la arquitectura general propuesta por la WfMC para los sistemas manejadores de *workflows*.

Figura 3. Arquitectura general de los WFMS.



Como puede observarse la arquitectura ofrece cinco componentes, cinco interfaces y uno o más motores de ejecución, a continuación se describen cada uno de estos componentes:

1.1.7.1 Motor de ejecución del *workflow* (*workflow engine*)

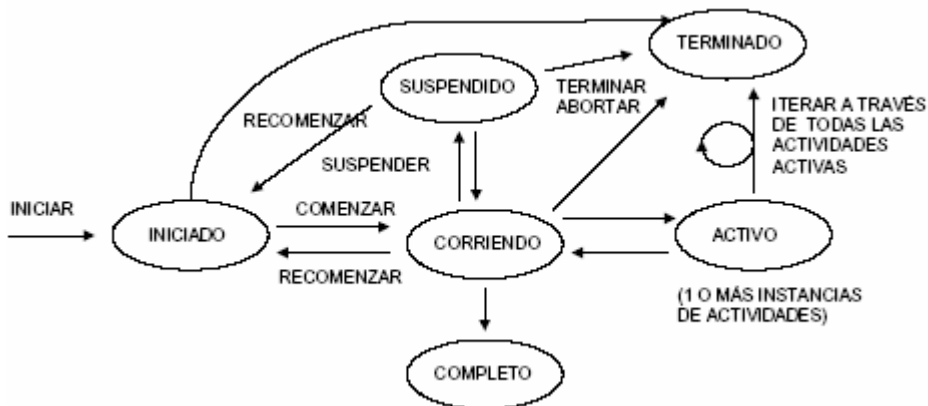
Provee facilidades para la interpretación de la definición de procesos, control de las instancias de los procesos, navegación entre las actividades, invocación de aplicaciones externas y el control del ambiente de ejecución de un *workflow*.

Típicamente dicho software provee facilidades para:

- a. Interpretación de la definición de procesos.
- b. Control de las instancias de los procesos: creación, activación, terminación, etc.
- c. Navegación entre actividades.
- d. Soporte de interacción con el usuario.
- e. Ruteo de datos al usuario o a aplicaciones.
- f. Invocación de aplicaciones externas.

El Motor de ejecución del *workflow* puede ser considerado como una máquina de estados, donde los procesos cambian de estados según eventos externos, o decisiones de control específicas, tomadas internamente por el motor de *workflow*. Los procesos están constituidos por diversas actividades. La culminación de las actividades que constituyen un proceso, implica la finalización del mismo.

Figura 4. Estados básicos dentro de un esquema de transición para un proceso



La figura 4 ilustra los estados básicos dentro de un esquema de transición para la instancia de un proceso. Las transiciones entre los distintos estados están representadas por las flechas. Cuando se publica un flujo de trabajo para su ejecución el motor de ejecución se encarga de poner el flujo en el estado iniciado y luego de acuerdo a los eventos se que

realizan con el flujo, el estado del proceso va cambiando. A continuación analizaremos con más detalle los distintos estados de los procesos y sus transiciones.

Iniciado: Ha sido creada una instancia del proceso, pero no se han dado las condiciones para su ejecución. Este es el estado normal de los procesos que se publican en un sistema administrador de flujos de procesos. Han sido compilados y están listos para su ejecución. Esta es la primera etapa de todo proceso dentro del motor de ejecución. Este proceso se encuentra cargado en memoria y está listo para ejecutarse apenas se reciba el mensaje de comenzar.

Corriendo: Al recibir el mensaje de comenzar se carga el proceso a memoria y se pone en la lista de los procesos que se encuentran en ejecución. El proceso en este estado es capaz de recibir los parámetros necesarios para su ejecución. Está a la espera de iniciar su ejecución con el mensaje de ejecución. En este estado están siempre las actividades cuando no están activas sino a la espera de un evento.

Activo: Este estado corresponde a las actividades que se encuentran en ejecución en ese preciso momento. El procesamiento de una actividad requiere que esté activa. Luego de que se procesa normalmente regresa al estado de corriendo para esperar otro evento que la lleve al estado de activo. Una o más actividades del proceso comenzaron. Normalmente la actividad que está en este estado recibe parámetros que condicionan la siguiente actividad a ejecutarse y esa actividad pasa a ser activa y la anterior a corriendo.

Suspendido: Cuando una actividad envía una notificación y depende de su respuesta para continuar su procesamiento se saca de la lista de activas o corriendo y pasa a este estado. Normalmente es cuando las actividades dependen de pasos humanos como la recepción de un correo o notificación la cual puede llevarse mucho tiempo es que se suspende la ejecución de la actividad. También se puede suspender la actividad por parte del administrador para evitar que el proceso siga su curso normal.

Completado: El proceso culminó, se realizan las acciones programadas (auditoria) y luego se elimina la instancia del proceso. Con fines administrativos los tiempos de cada actividad y las tareas que se han ejecutado siguen disponibles para los reportes de desempeño.

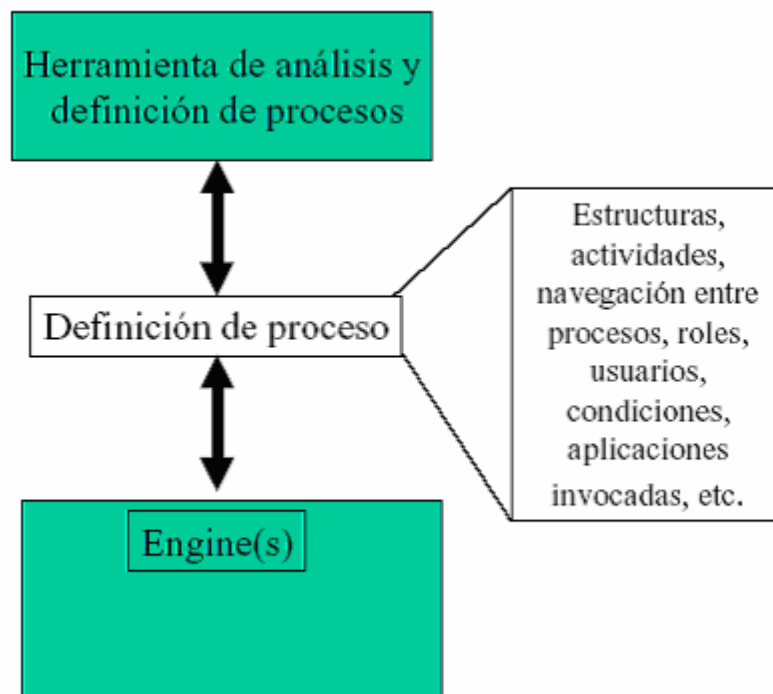
Terminado: Si ocurre algún error durante la ejecución de una actividad, el proceso termina con éste estado indicando que no se pudo terminar normalmente la ejecución del proceso.

1.1.7.2 Herramientas de definición de procesos (interfase 1)

La meta de esta interfase es realizar la especificación del proceso de tal manera que la salida de este proceso de modelado y diseño sea una definición de procesos la cual pueda ser interpretada en tiempo de ejecución por el o los motor(es) de *workflow*.

Existe gran cantidad de herramientas de definición de procesos (independientes de los productos en muchos casos), las cuales deben comunicarse con los motores de algún producto de *workflow*, lo deseable es que esta herramienta pueda comunicarse con cualquier motor, esto únicamente sería posible si se establecen ciertas normas de comunicación entre las herramientas de definición de procesos y un motor de *workflow*. Por esto la WfMC propone una interfase para esta comunicación. El objetivo de la interfase es dar un formato de intercambio y llamadas a API (*Application Programming Interface*), para soportar el intercambio de información de definición de procesos. El intercambio podría ser una completa definición de los procesos o un subconjunto de la misma. En la figura 5 se muestra la composición de la definición de procesos.

Figura 5. Herramientas de definición de procesos (interfase 1)



1.1.7.3 Aplicaciones clientes (interfase 2)

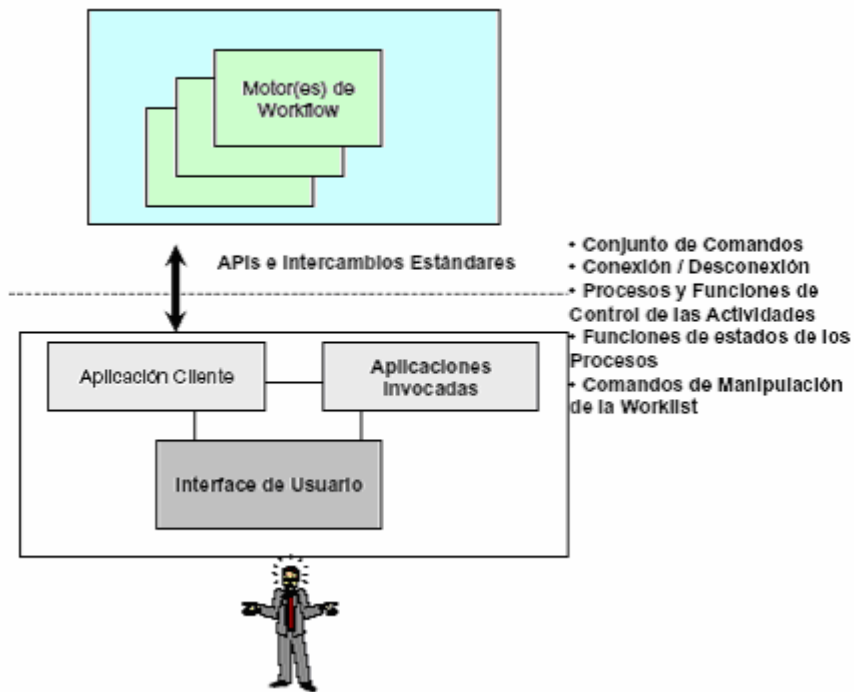
La interacción entre las aplicaciones clientes y el motor de ejecución está basada en gran parte en el concepto de lista de trabajo (*worklist*), donde se colocarán las actividades a ser ejecutadas por una aplicación. Parte de la información almacenada en la lista de trabajo es utilizada para transmitirle al manejador el nombre de las aplicaciones a invocar.

La *worklist* podría contener *items* relacionados con diferentes instancias de un proceso o *items* de diferentes procesos. El manejador de la *worklist* podría estar interactuando con diferentes motores. La interfase entre una aplicación cliente de *workflow* y el motor de *workflow* debe ser lo suficientemente flexible en los siguientes puntos:

- a. *Identificadores de procesos y actividades.*
- b. *Estructuras de datos.*
- c. *Diferentes alternativas de comunicación.*

La figura 6 es un esquema que muestra las componentes que participan en esta interfase.

Figura 6. Esquema de los componentes de las aplicaciones cliente (interfase 2)

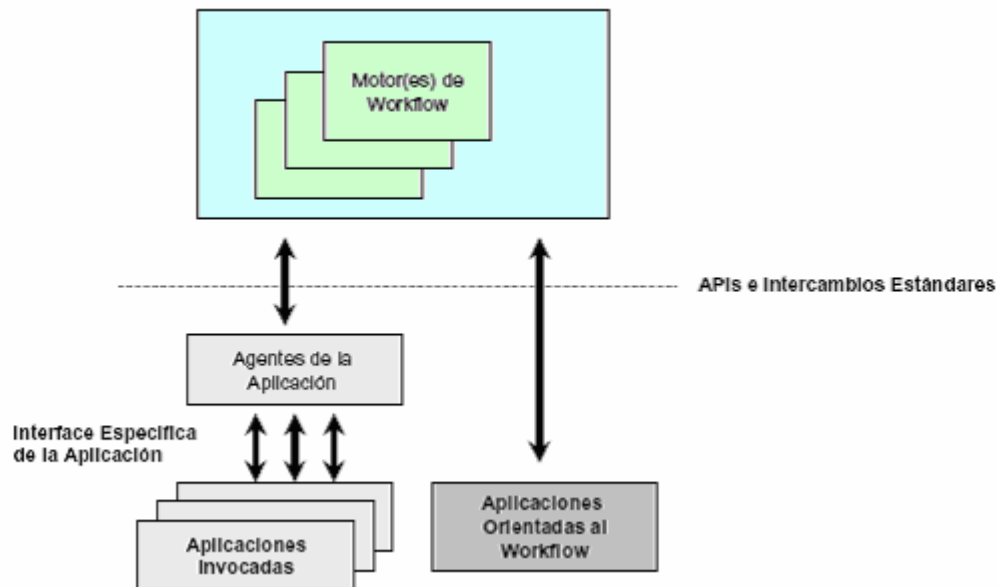


1.1.7.4 Aplicaciones invocadas (interfase 3)

Esta interfase está orientada a interactuar con agentes de una aplicación, o con toda la aplicación. Una aplicación debe estar orientada al contexto general de un sistema de *workflow*, es decir, debe poder interactuar directamente con el motor de *workflow*, usando la información suministrada en la definición del proceso para identificar la naturaleza de la actividad.

La figura 7 muestra el alcance de esta interfase, la cual está pensada para interactuar con agentes de una aplicación, o con una aplicación entera propiamente dicha. Dichas aplicaciones deben estar orientadas con el contexto general de un sistema de *workflow*, es decir, deben poder interactuar directamente con el motor de *workflow*. La aplicación invocada es manejada localmente por un motor de *workflow*, usando la aplicación que se invoca puede ser local al motor de *workflow*, o sea, residente en la misma plataforma, o estar en otra plataforma dentro de una red. En este caso la definición del proceso debe contener la información necesaria para poder encontrar la aplicación que se va a invocar (como es la dirección dentro de la red). Los detalles semánticos y sintácticos de esta interfase se pueden encontrar en el documento elaborado por la WfMC, donde se especifica en forma detallada esta interfase.

Figura 7. Esquema de los componentes de las aplicaciones invocadas (interfase 3)

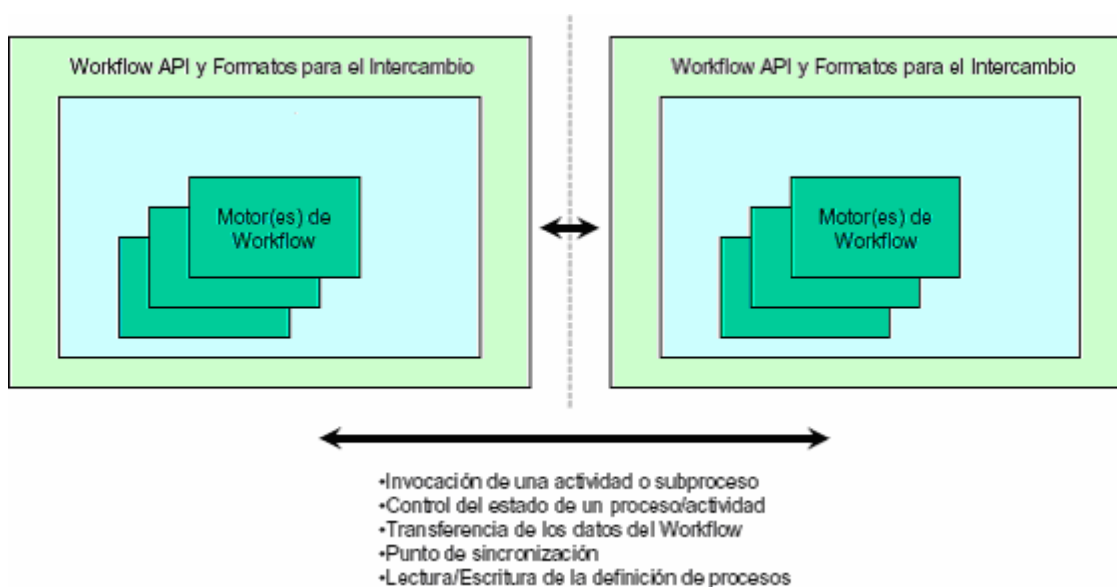


1.1.7.5 Funciones de interoperabilidad WAPI (interfase 4)

Existen dos aspectos necesarios para la interoperabilidad que deben ser satisfechos en esta interfase: la interpretación de la definición de procesos que será realizada y el soporte en tiempo de ejecución para el intercambio de información de control y transferencia de los datos relevantes del *workflow*, y/o de las aplicaciones entre los distintos servicios de representación.

La figura 8 muestra la forma general del intercambio y control de flujo entre sistemas de *workflow* heterogéneos.

Figura 8. Interfase de interoperabilidad del *workflow* (interfase 4)

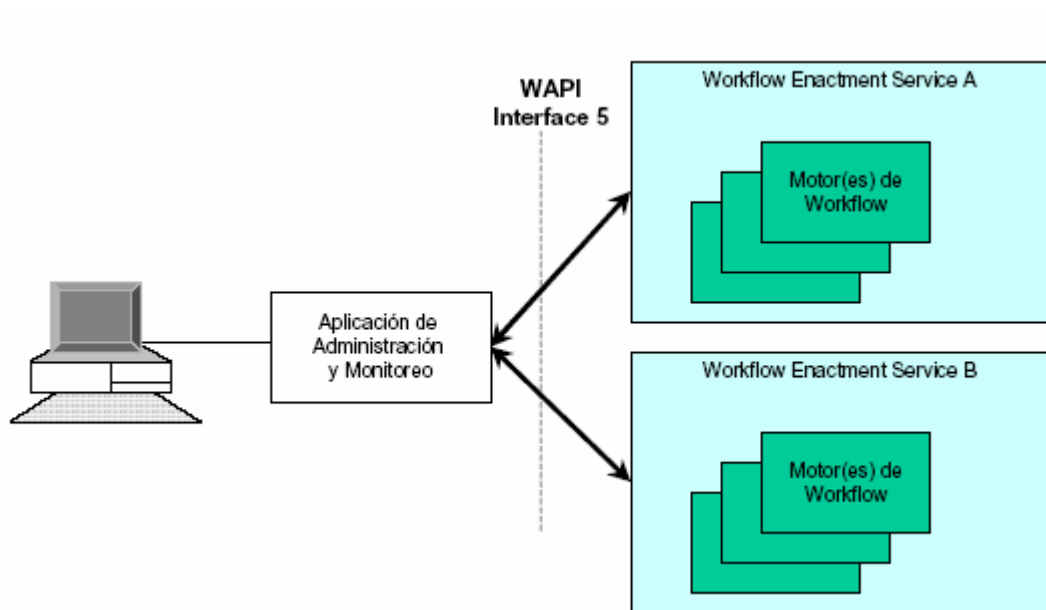


1.1.7.6 Herramientas de administración y monitoreo (interfase 5)

El propósito de esta interfase es permitir una vista completa del estado del flujo de trabajo, además de poder realizar auditorías sobre los datos del sistema. La figura 9 ilustra como una

aplicación de administración independiente, interactúa con los distintos dominios del *workflow*. Es posible implementar otros posibles escenarios, como por ejemplo tener la aplicación de administración y monitoreo dentro del mismo servicio.

Figura 9. Herramientas de administración y monitoreo (interfase 5)



1.2 TECNOLOGÍA *WORKFLOW*

1.2.1 Conceptos

Con el objetivo de comprender la tecnología *workflow* se hará un breve recorrido por algunos de sus conceptos más importantes, esto servirá para entender la importancia de esta tecnología y como el éxito o fracaso del un WFMS se basa en la claridad de la definición de sus procesos.

1.2.1.1 Actividad o Tarea

El término actividad, como se menciono anteriormente, se refiere a uno de los conceptos más importantes de la tecnología *workflow*. Por medio de la identificación de actividades o tareas es posible estructurar *workflows*.

Una actividad o tarea se define como la unidad lógica de trabajo. Es atómica y por esta razón debe ser llevada a cabo completamente. Si algo va mal durante el funcionamiento de una tarea, debe regresarse completamente al inicio de la tarea. Sin embargo, la naturaleza atómica de la actividad depende del contexto en el cual está definida. Escribir una carta, evaluar un reporte, llenar una forma y revisar los datos de un empleado, son ejemplos de actividades. Se puede diferenciar entre actividades manuales, automáticas y semiautomáticas.

Una actividad manual es aquélla que es llevada a cabo, completamente, por una o más personas, sin usar ninguna aplicación, un ejemplo de este tipo de tarea es llevar un cheque (físico) de una oficina a otra. En cambio, una tarea automática es llevada a cabo sin intervención alguna de personas. Esto significa que una aplicación, es decir un programa de computadora, va a realizar por completo la actividad basándose en datos que han sido previamente registrados. Cuando tanto una persona como un programa de computadora están involucrados en la realización de una tarea, se dice que este tipo de tarea es semiautomática; un ejemplo de este tipo de tareas es la evaluación de alguna clase de reporte por un asesor que utiliza un programa de computadora especialmente diseñado para ayudarlo a realizar esta tarea.

1.2.1.2 Personas (usuarios)

Las tareas son realizadas en un orden definido por determinadas personas (o agentes automatizados tomando el rol de las personas) basados sobre las condiciones o reglas del negocio.

1.2.1.3 Roles

Cada rol define las distintas competencias potenciales que existen en el sistema. Se definen independientemente de las personas físicas a las cuales se les van a asignar dichos roles. Una persona puede tener más de un rol.

1.2.1.4 Rutas

Una ruta define la secuencia de pasos a seguir por los documentos o información dentro de un sistema de *workflow*. La capacidad de rutiar las tareas a usuarios remotos u ocasionales es vital en una aplicación de *workflow*. Para asegurar el éxito del flujo de información y decisiones, todos los miembros del equipo deben ser capaces de tomar parte en este proceso. Se distinguen varios tipos de rutas:

a. Rutas Fijas:

En este caso los documentos siguen siempre el mismo camino. Se define de antemano cual es la próxima etapa a seguir.

b. Rutas Condicionales:

El camino a seguir depende de la evaluación de condiciones. Estas decisiones se toman en el mismo momento que se pasa por el punto donde hay que evaluar las condiciones.

c. Rutas Ad Hoc:

En este caso el usuario elige explícitamente cual es la siguiente etapa a seguir.

1.2.1.5 Procesos

Según la *Workflow Management Coalition* (WfMC) un proceso se define como: “Un conjunto de uno o más procedimientos o actividades directamente ligadas, que colectivamente realizan un objetivo del negocio, normalmente dentro del contexto de una estructura organizacional que define roles funcionales y relaciones entre los mismos.”

Un proceso describe las tareas que deben ser realizadas y el orden en que debe realizarse cada una de ellas. Un proceso puede estar compuesto de cero o más subprocesos. Cada uno de estos subprocesos está compuesto por tareas, e incluso de más subprocesos. De esta manera procesos muy complejos pueden estar organizados de forma jerárquica.

Si se observa cualquier empresa de la actualidad, podemos apreciar que la misma esta constituida por varias partes. Son estas partes quienes tienen que relacionarse fluidamente para que se cumplan el cometido de la empresa. La definición anterior plantea este contexto, en el cual para que todo funcione correctamente, se deben definir previamente la forma en que se comunicarán dichas partes. Actualmente no existe un buen grado de formalización de los procesos de empresa. Un área con gran potencial es rediseñar procesos existentes, eliminar redundancias, identificar “cuellos de botella” y entender porque es que se hace lo que se hace.

1.2.1.6 Condiciones de transición

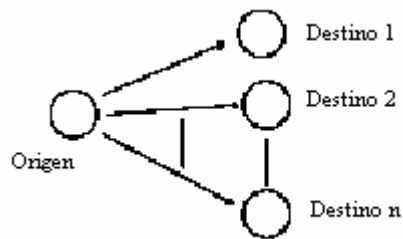
La más simple forma de orden es la ejecución secuencial de las tareas. Esto usualmente significa que hay una dependencia entre ellas. Es decir, el resultado de una de estas tareas es la entrada de la otra.

En construcciones más complejas se utilizan operadores para indicar el orden de ejecución de los procesos. En el orden en el que se ejecutan las tareas puede hacer uso de cinco operadores para formar las construcciones del flujo:

a. *AND-Split*:

A partir de un lugar fuente, los documentos son distribuidos hacia varios destinos simultáneamente. Se utiliza cuando las tareas necesitan forzosamente ser ejecutadas simultáneamente o en paralelo. Se ilustra en la figura 10.

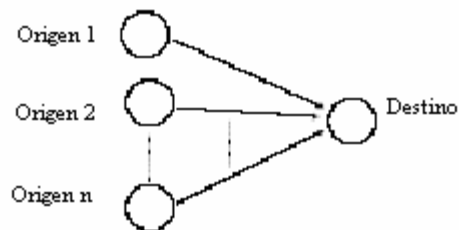
Figura 10. Construcción AND-Split



b. *AND-Join*

A partir de varios lugares fuentes, los documentos convergen, sincrónicamente, hacia un único destino. Se muestra en la figura 11

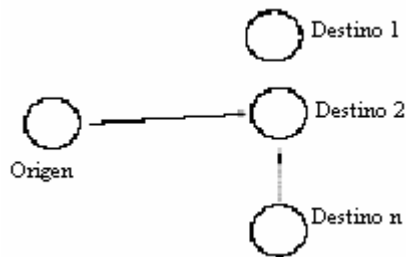
Figura 11. Construcción AND-Join



c. OR-Split

A partir de un lugar origen, los documentos toman un destino entre varios posibles. Se muestra en la figura 12.

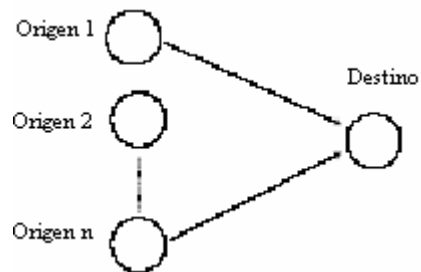
Figura 12 Construcción OR-Split



d. OR-Join

A partir de uno o más lugares de origen, dentro de varios posibles, convergen hacia un único destino (no se requiere sincronización). Se muestra en la figura 13.

Figura 13. Construcción OR-Join



e. Ciclo o iteración

En una situación ideal una tarea no se realiza más de una vez por caso. Sin embargo, algunas veces es necesario ejecutar una tarea muchas veces. A esta forma de orden se le llama iteración o ciclo. En este caso se forma un circuito cerrado dentro del camino que recorren los documentos. Se muestra en la figura 14

Figura 14. Construcción ciclo o iteración.



1.2.1.7 Agente

Los agentes en el contexto de la tecnología *workflow* se definen como entidades ya sean humanas o computacionales que llevan a cabo las actividades de un proceso. De acuerdo a la forma en la que se definen los procesos y las actividades, los agentes pueden contar con ciertas características que permitan al comportamiento ir escogiendo uno a uno para ir ejecutando cada actividad.

1.2.1.8 Comportamiento

Describe los pasos que se siguen durante la ejecución del flujo de trabajo y que caracterizan su comportamiento. Esta descripción incluye el momento de disparo de cada una de las actividades, qué acciones se deben de realizar en caso de error, el criterio para la elección de agentes.

Por ejemplo, una tarea puede ser disparada por iniciativa de un empleado. Sin embargo, existen otras formas para disparar una actividad y estas son: por un evento externo, por ejemplo la recepción de un mensaje; o porque se ha alcanzado cierto límite de tiempo, por ejemplo el procesamiento de una lista de órdenes comienza a las 6:00 en punto. En este contexto se distinguen dos tipos de disparadores de tareas: por iniciativa de un agente o por un evento externo.

1.2.1.9 Eventos

Un evento es una interrupción que contiene información, el mismo tiene un origen y uno o más destinatarios. La información contenida en el mensaje que se produjo por el evento puede ser implícita o dada por el usuario. Los eventos pueden ser disparados voluntariamente por el usuario; o en forma implícita durante un proceso según el estado de los datos o de decisiones tomadas por el usuario; o en forma automática. Por ejemplo, cuando un gerente de un banco hace una consulta sobre ciertos datos para hacer una auditoria, se dispara un evento que le devuelve la información de dicha consulta.

1.2.1.10 Plazos (*Timeouts*)

Podemos ver a los plazos como los tiempos que se le asignan a ciertos elementos. Ejemplos de plazos pueden ser: el tiempo máximo que se le asigna a una tarea para que sea terminada; el tiempo máximo para recorrer una ruta; terminar una tarea antes de cierta fecha; terminar el recorrido de una ruta antes de cierta fecha; y así podríamos seguir. A los plazos podemos asignarles eventos, de forma tal de que cuando venza determinado plazo se disparen ciertos eventos asignados por el usuario, o programados para que se disparen automáticamente.

2. ORACLE WORKFLOW

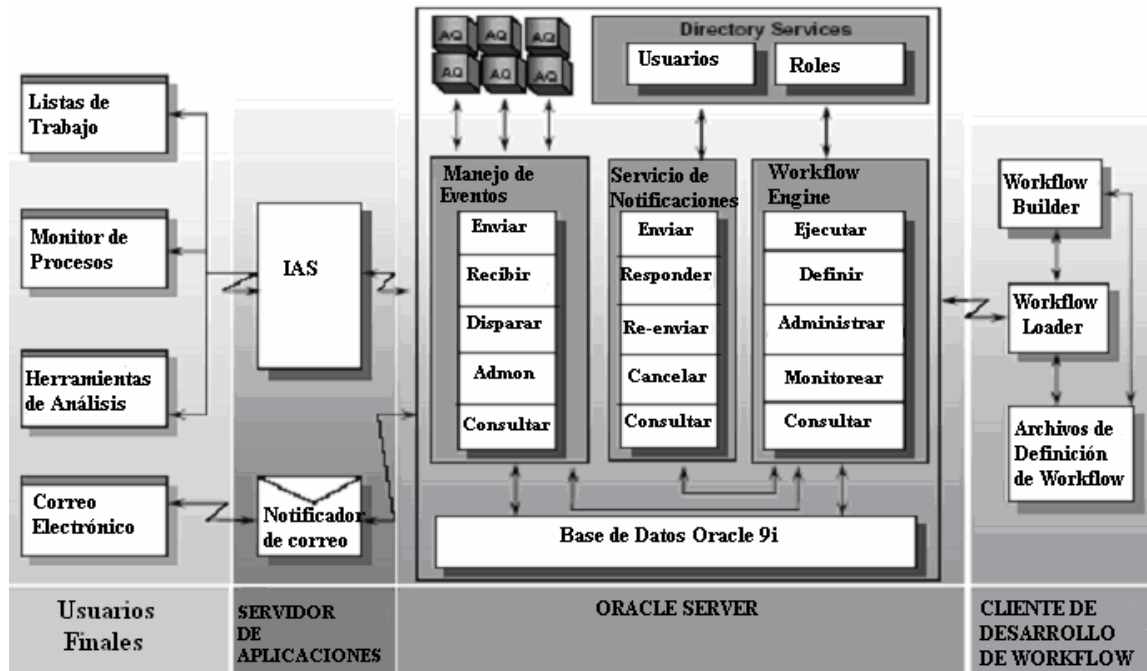
Oracle workflow es un *workflow management system* (WMS) completo que soporta el modelado, automatización, almacenamiento, ejecución y mejora continua de procesos de negocios, enrutando la información de cualquier tipo de acuerdo a reglas definidas por el usuario que pueden ser fácilmente modificables.

Oracle workflow es fácilmente extensible. Su arquitectura abierta y que cumple con los estándares hace que sea un producto que puede integrarse fácilmente a una aplicación existente. Tiene muchas características innovadoras como su herramienta gráfica de diseño *Oracle workflow builder* y la integración con *Java* cumpliendo con el estándar J2EE.

2.1 ARQUITECTURA DE ORACLE WORKFLOW

Oracle workflow está dividido en cuatro grandes componentes. Cliente de desarrollo, *oracle server*, servidor de aplicaciones y navegador del cliente. Como se puede apreciar en la figura 15.

Figura 15. Arquitectura de *oracle workflow*



2.1.1 Cliente de Desarrollo de *workflow*

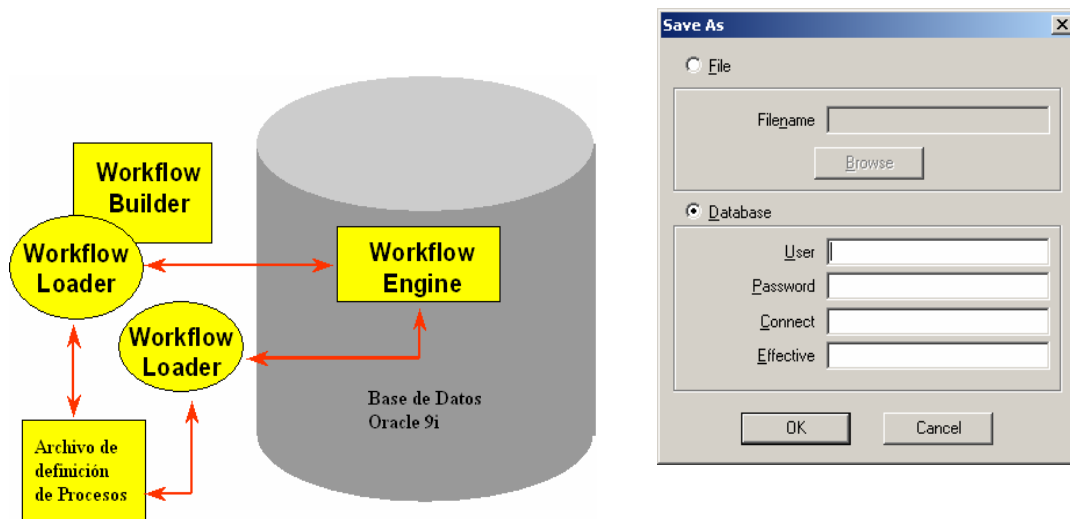
El cliente de desarrollo, llamado *oracle workflow builder* consiste en una serie de programas que se ejecutarán en una computadora personal para asistir al desarrollador a modelar, diseñar, consultar y almacenar procesos de *workflow*. Las definiciones de los flujos de trabajo se podrán almacenar en archivos de definición de *workflows* (.wtf) o bien dentro de la base de datos.

Cuando se instala el *oracle workflow builder* se instala también un utilitario denominado: *workflow loader*. Este componente sirve para leer las definiciones de los procesos de *workflow* y cargarlos a la base de datos.

En la figura 15 vemos la relación que existen entre estos dos componentes, ya que el *workflow builder* se utiliza para construir el diagrama del proceso, luego éste se almacena en un archivo local. Por último se utiliza el *workflow loader* para almacenar la definición del proceso en la base de datos. Todo esto se realiza desde el mismo entorno de desarrollo, facilitando la realización de éstas tareas. Para utilizar el *workflow loader*, se elige la opción “*save as*” del menú de “*File*”.

Las definiciones de los procesos se pueden grabar en la base de datos para que se ejecuten o bien en un archivo local con extensión wft. Esto le permite al diseñador tener una copia de seguridad de su trabajo así como una manera de modificar los flujos sin afectar lo que se encuentra en producción. Ésta relación es representada en la figura 16.

Figura 16. Componentes del cliente de desarrollo



2.1.2 Oracle Workflow Builder

Oracle workflow builder es una herramienta gráfica para la creación, consulta y modificación de definiciones de flujos de trabajo. Tiene dos componentes principales el navegador que se utiliza para definir las actividades y componentes de los procesos, y el diagramador de procesos, para estructurar las transiciones entre las instancias de las actividades.

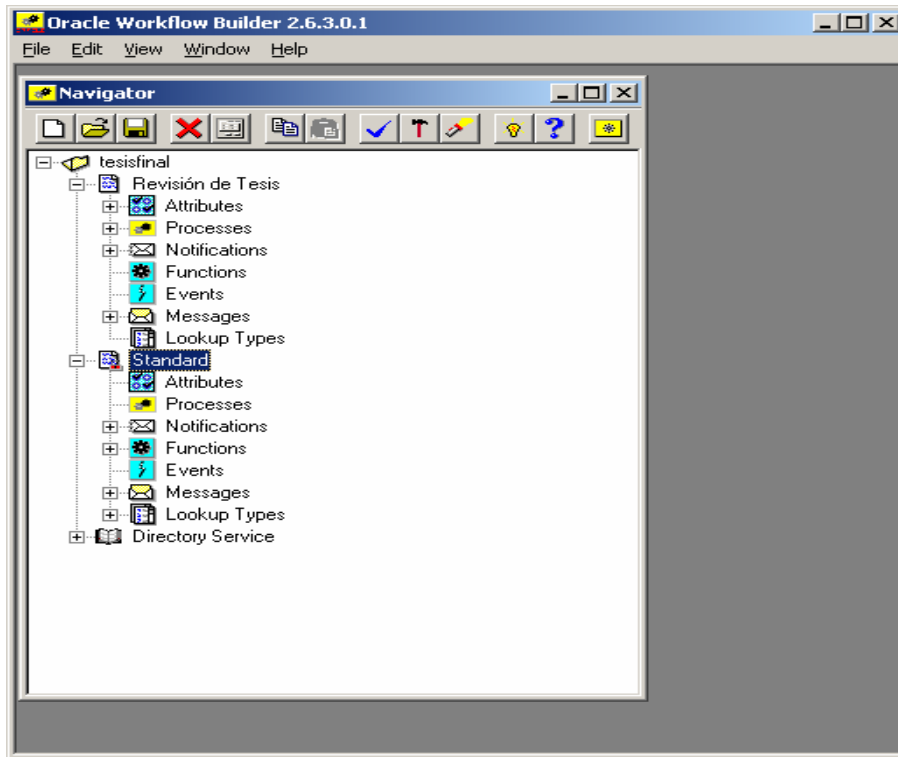
El navegador proporciona una estructura jerárquica para la definición de flujos de trabajo, siendo el nivel más alto el almacén de datos (archivo o base de datos). Luego en la jerarquía sigue el *item type*, el cuál puede agrupar uno o más flujos de trabajo que estén interrelacionados. Al nivel más bajo de la jerarquía se encuentran los atributos, procesos, notificaciones, funciones, eventos, mensajes y listas de valores. Estos componentes se analizarán con detalle más adelante.

El segundo componente es el diagramador de procesos, el cuál permite crear representaciones gráficas, es decir diagramas, que muestran las diferentes actividades involucradas en un flujo de trabajo y las transiciones entre ellas. Un diagrama se construye arrastrando los componentes del navegador hacia la ventana del diagramador.

2.1.2.1 El navegador

Proporciona una estructura de árbol donde el nivel más alto es el origen de datos. El nivel más bajo corresponde a los componentes del flujo de trabajo, por ejemplo, mensajes, notificaciones y funciones. Todos estos objetos están organizados dentro de sus respectivas categorías, como se aprecia en la Figura 17.

Figura 17. El navegador de Oracle workflow builder



Un origen de datos (primer nivel de la jerarquía) es una conexión a una base de datos o un archivo que almacena la definición del proceso de flujo de datos. Dentro de cada origen de datos hay al menos un *item type*. En la figura 2.3 el origen de datos es “tesisfinal”

Un *item type* (segundo nivel) es una clasificación de los componentes que conforman un proceso de flujo de trabajo. Se debe asociar cualquier componente que se cree para un proceso, tales como funciones o mensajes, a un *item type* en particular. En la figura 17 tenemos dos *item types*, el primero llamado “Revisión de Tesis” y el segundo “Standard”.

Los siguientes elementos (tercer nivel) aparecen por debajo del nivel del *item type*:

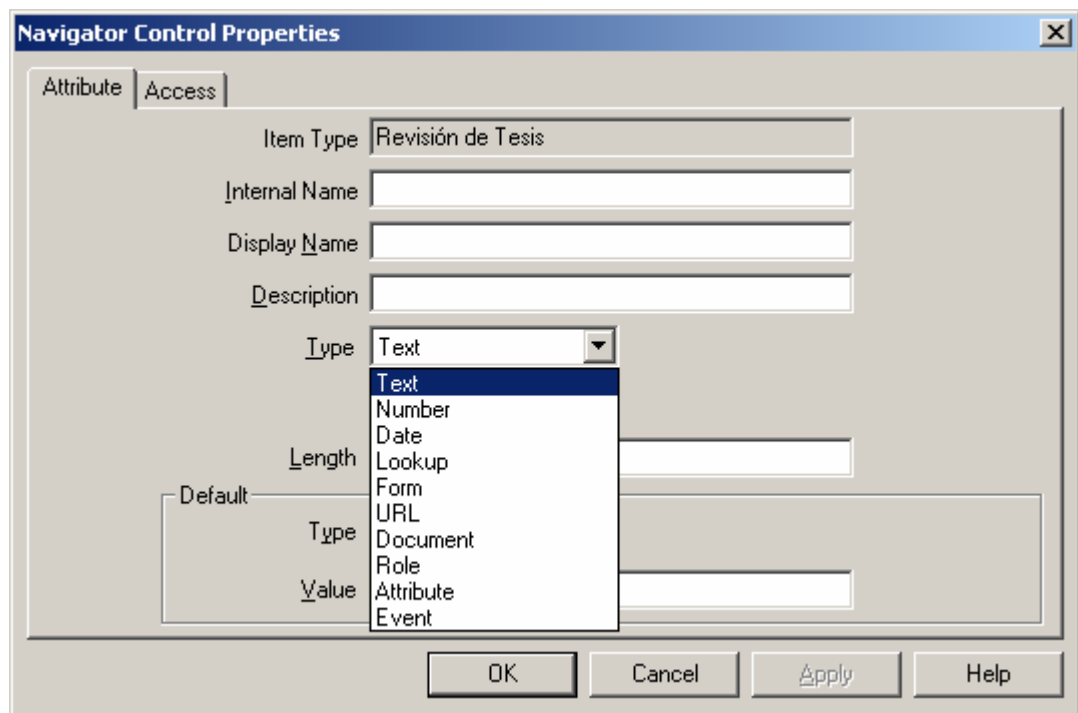
Atributos: En este nivel se listan los atributos para el *item type* actual. Los atributos describen características para un *item type*. Un atributo es una propiedad asociada con un *item type* dado. Actúa como una variable global que puede ser referenciada o actualizada por cualquier actividad dentro del proceso. Un atributo a menudo provee información que es necesaria para que el proceso del flujo de trabajo se complete.

Por ejemplo, si un *item type* es una requisición de orden de compra, entonces un atributo puede ser el monto o el número de la requisición y una actividad puede requerir el valor de éste atributo para determinar que autoridad tiene la capacidad de autorizar una solicitud por ese monto.

Aplicaciones externas también pueden referenciar y modificar un atributo utilizando el *oracle workflow engine* APIs. Se pueden definir tantos atributos como sean necesarios para un *item type*. Y se deben definir como atributos, cualquier información que sea requerida por una actividad dentro del proceso o por un mensaje dentro de una notificación.

Existen diez tipos de atributos, como se muestra a continuación. El tipo de atributo determina qué valores son aceptables y la manera en que el atributo será utilizado. Para crear un atributo existen varios métodos, uno es seleccionar el nodo *attribute* dentro del navegador y con el botón derecho del *mouse* seleccionar la opción de *new attribute*. Esto nos muestra la ventana de propiedades de los atributos. En la figura 18, podemos ver ésta ventana.

Figura 18. Propiedades de un atributo



Atributo tipo *Text*: El valor del atributo es una cadena de texto. Opcionalmente podemos definirle una longitud y un valor por defecto.

Atributo tipo *Number*: El valor del atributo es un número con una máscara de formato opcional.

Atributo tipo *Date*: El valor del atributo es una fecha con una máscara de formato opcional.

Atributo tipo *Lookup* o lista de valores: El valor del atributo es uno que exista en la lista de valores del tipo *lookup* que se haya elegido. Es decir, primero se necesita

crear el tipo *lookup* (lista de valores) y los valores de éste atributo estarán limitados a los definidos en la lista de valores.

Atributo tipo *URL*: El valor del atributo es un localizador de recursos universal que apunta a una localidad de red específica. Si se hace referencia a un atributo de tipo *URL* en el cuerpo de una notificación, entonces al ser vista, la notificación, a través de la página de notificaciones, se despliega una ancha al valor del atributo tipo *URL*. El usuario puede obtener información adicional haciendo *click* sobre el *URL*.

Atributo tipo *PL/SQL CLOB document*: El valor de este atributo es un documento anexo. El atributo es un documento representando datos de la base de datos como un *character large object* (*CLOB*), generado a partir de un procedimiento de *PL/SQL*. El *CLOB*, puede contener: texto plano, *HTML*, *PDF* o bien *RTF*.

Atributo tipo *PL/SQL BLOB document*: El valor de este atributo es un documento anexo. El atributo es un documento representando datos de la base de datos como un *binary large object* (*BLOB*), generado a partir de un procedimiento de *PL/SQL*. El *BLOB*, puede contener una imagen o cualquier otro tipo de dato almacenado como datos binarios.

Atributo tipo *Role*: El valor de éste atributo es el nombre interno de un *role*. Si un atributo de tipo *role* es incluido en el mensaje de una notificación, el atributo automáticamente resuelve el nombre de despliegue del *role*, eliminando la necesidad de tener nombres internos y de despliegues para un mismos *role*. El termino *role* en éste contexto se refiere tanto a *roles* de la base de datos como usuarios.

Actividades: Una actividad es una unidad de trabajo que contribuye a la realización de un proceso. Una actividad puede ser una notificación, una función, un evento o un proceso.

Las actividades están organizadas, en el navegador en los apartados de procesos, notificaciones, funciones y eventos. Se pueden crear, editar y borrar las definiciones de las actividades en el navegador, y arrastrar una actividad del árbol a la ventana de procesos para crear un nuevo uso de tal actividad en el diagrama de procesos.

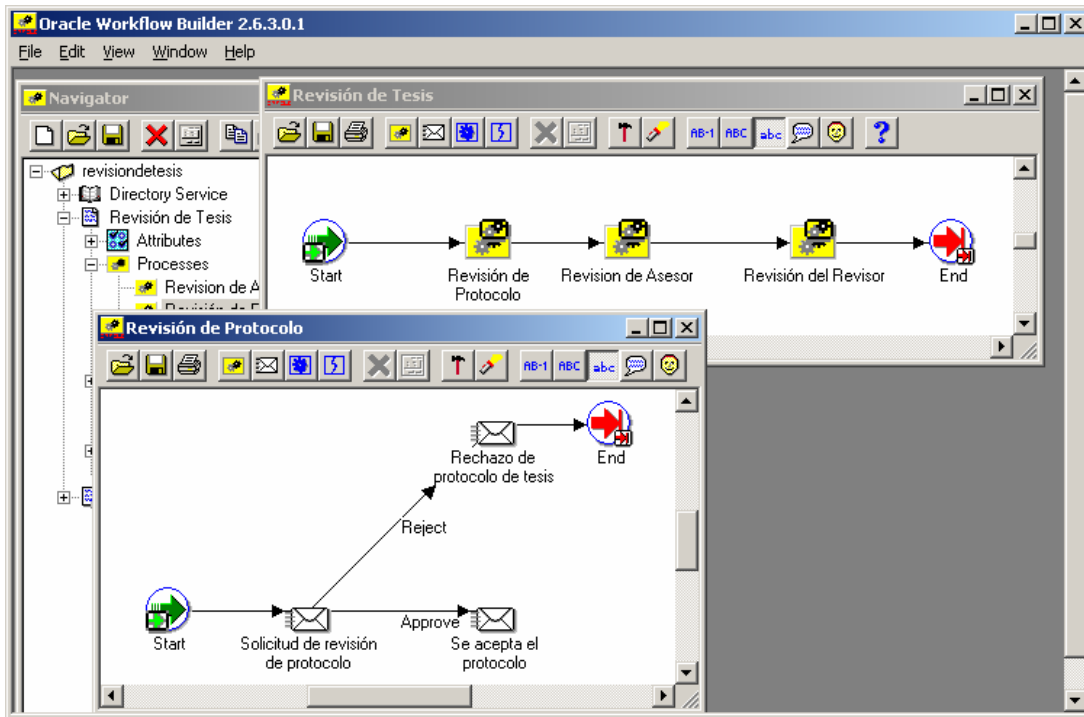
Oracle workflow proporciona un *item type* llamado *standard* que incluye actividades genéricas que se pueden utilizar en cualquier proceso que se defina. Por ejemplo, algunas de las actividades realizan funciones tales como comparar dos valores.

Oracle workflow también proporciona un *item type* llamado *system* que incluye actividades y procesos para el manejo de errores. Se puede asignar un proceso de error a una actividad, de tal manera que si un error ocurre, el proceso de error le indica a *Oracle Workflow* la manera de manejarlo.

Procesos: Son un tipo de actividad que tienen la característica de corresponder a una serie de actividades fuertemente ligadas y que con cada actividad se determina la siguiente a realizar. Un proceso es modelado utilizando el diagramador de procesos y dentro de cada uno de los procesos se puede definir otros sub-procesos para poder manejar la complejidad de esta manera se hace más fácil de entender el diagrama.

Cuando un proceso cuenta con muchas actividades puede llegar a ser muy confuso de entender, por lo que se recomienda tener varios niveles de procesos. Esto se logra creando un subproceso que contenga varias actividades. Para hacer esto debemos seleccionar el icono de proceso dentro del diagramador de procesos y colocarlo en la posición adecuada dentro de nuestro diagrama. Luego para colocar el detalle del subproceso debemos hacer doble *click* en el icono de subproceso dentro del diagrama. Esta situación la podemos apreciar en la figura 19.

Figura 19. Diagramador de procesos con subprocessos



Notificaciones: Son un tipo de actividad que se utiliza para enviar mensajes a un usuario o a un role. El mensaje puede incitar una respuesta o bien únicamente puede proveer información. Para crear una notificación se debe crear primero un mensaje y luego asociarlo a la definición de la notificación. Para crear una definición de una notificación nos colocamos en el navegador y con el botón derecho del *mouse* elegimos la opción “*new notification*”. La ventana de propiedades de la notificación se muestra en la figura 20.

Figura 20. Propiedades de una notificación

The image shows a dialog box titled "Navigator Control Properties" with a close button (X) in the top right corner. It has four tabs: "Activity", "Details", "Roles", and "Access". The "Activity" tab is selected. The dialog contains the following fields and controls:

- Internal Name: Text input field.
- Display Name: Text input field.
- Description: Text input field.
- Icon: A dropdown menu showing "NOTIFY.ICO" and a "Browse" button with an envelope icon.
- Function Name: Text input field.
- Function Type: A dropdown menu showing "PL/SQL".
- Result Type: A dropdown menu showing "<None>" and an "Edit" button.
- Message: A dropdown menu and an "Edit" button.
- Expand Roles: A checkbox.

At the bottom of the dialog are four buttons: "OK", "Cancel", "Apply", and "Help".

Al definir una notificación se debe colocar un nombre interno único, para todo el *item type*, un nombre que se desplegará en los diagramas, el “*display name*” y una descripción. Una notificación puede estar asociada con una función. En éste caso es necesario indicar el nombre de la función y cada vez que se ejecute la notificación se ejecutará la función.

Además podemos asociar a la notificación un *result type*. Estos son listas de valores que muestran los posibles resultados de una notificación. Así por ejemplo podemos tener un *result type* llamado *Approve* el cual tiene dos posibles resultados: aprobado o rechazado.

Cuando el *workflow engine* alcanza una actividad de notificación, se ejecuta una llamada *Send()*, del API del *workflow engine*, hacia el sistema de notificación, para que éste envíe un mensaje al *role* asignado en el proceso. El mensaje que se envía lo define el diseñador del proceso a través de la propiedad *Message*. Se debe especificar el destinatario de una notificación cuando se incluye un nodo de notificación dentro de un diagrama de procesos. El destinatario puede ser tanto un *role* específico o bien un atributo que dinámicamente retorna el nombre de un *role*.

El mensaje puede ser una nota informativa o bien puede solicitar al destinatario una respuesta. Cuando el destinatario responde a una notificación, el sistema de notificaciones procesa la respuesta y ejecuta una llamada *Response()* al API del *workflow engine*. La función de post-notificación (si existe) puede interpretar la respuesta y hacer un procesamiento extra, sin embargo si ésta función termina con una *exception*, la respuesta es abortada. De lo contrario se informa al *workflow engine* que la notificación se ha completado, para que el proceso continúe hacia la siguiente actividad.

El destinatario de una notificación tiene la oportunidad de hacer *forward* de ésta notificación hacia otro usuario para que sea él quien continúe con el proceso. Si se quiere deshabilitar ésta opción se debe realizar la tarea dentro de la función de post-notificación.

Las notificaciones pueden tener asociado un *time out*. En éste caso el *workflow engine* se encarga de periódicamente chequear el tiempo transcurrido en cada actividad y si se alcanza el tiempo de *time out* el proceso continuará en la transición marcada con el valor de *timeout*.

Un tipo especial de notificación es la de votación. Es una notificación que ha expandido sus roles, por lo que una copia individual del mensaje de la notificación es

enviado a cada miembro del role destino. Además tiene un mensaje con un resultado específico, que requiere que el destinatario elija de una lista de valores. Y por último tiene una función de post-notificación asociada que contiene la lógica para procesar las respuestas elegidas por los destinatarios, para generar una única respuesta que el *workflow engine* interpreta como el resultado de la actividad.

Cuando el sistema de notificaciones envía una actividad de notificación de votación, ésta es marcada con un estatus de *NOTIFIED*. El estatus de la actividad es actualizada a *WAITING* tan pronto como se reciba la primera respuesta y se mantiene mientras no se reciban todas las respuestas necesarias para satisfacer el criterio de votación.

El usuario que recibe una copia del mensaje de notificación de votación, puede responder, hacer un *forward* o transferir la notificación a alguien más.

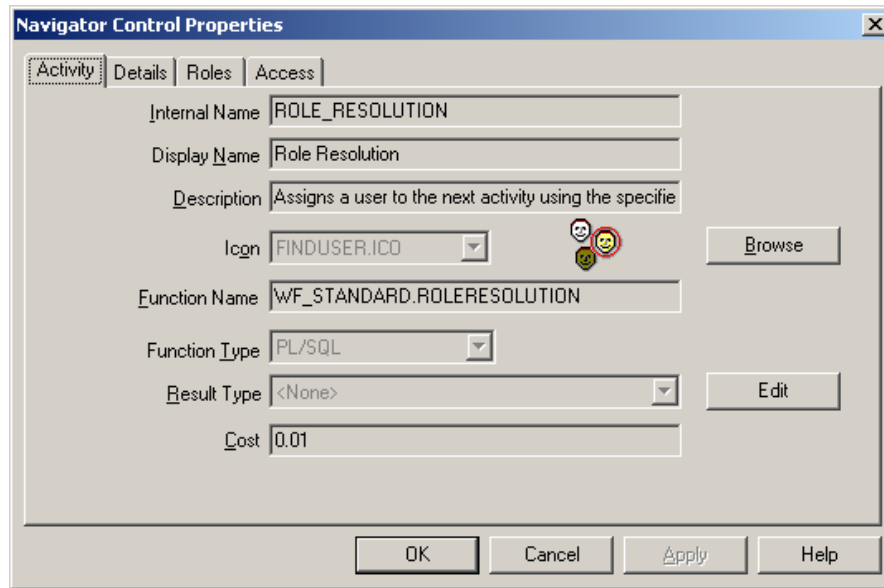
Al agregarse una notificación dentro de un proceso ésta pasa a ser una instancia de la definición guardada en el navegador.

Funciones: Son un tipo de actividad que se utiliza para representar código en procedimientos almacenados de PL/SQL que el *workflow engine* ejecuta automáticamente. Una función también tiene atributos asociados a él.

Las funciones pueden retornar un resultado que determina la siguiente transición que tomará el *workflow engine*. El resultado muchas veces se define como un código de una lista de valores o *lookup type*.

Para definir una nueva función es necesario elegir el nodo de *functions* en el navegador y luego con el botón derecho elegir *new function*. Esto muestra la ventana de propiedades de la nueva función, como se ve en la figura 21.

Figura 21. Propiedades de una función



Al definir una nueva función se debe especificar un nombre interno, un texto para mostrar en los diagramas y una descripción. Luego es necesario indicar el tipo de la función *function type*. Los valores posibles son: PL/SQL, *external Java* y *external*. Estos valores se refieren al lenguaje en el que se encuentra la función, en el caso de PL/SQL y *Java*, el código que se ejecutará deberá estar almacenado dentro de la base de datos. También se pueden ejecutar programas externos, para ello se debe elegir la opción de *external*. En cualquiera de los tipos de funciones es necesario indicar el nombre del programa que se ejecutará.

Por último se debe especificar el *result type* de la función. Esto significa los posibles valores que retornará la función, éstos valores deberán existir previamente en una lista de valores, *lookup type*, definida. También se debe indicar el costo relativo en segundos, de la función. Esto representa el tiempo que la función tarda en ejecutarse. Si el costo excede el umbral definido por el *workflow engine*, la

actividad se detendrá y su ejecución la llevará a cabo el *background engine*, hasta que termina la función.

Existen una gran variedad de funciones predefinidas que se pueden utilizar dentro del *item type* estándar. Sin embargo, los diseñadores de flujo de trabajo pueden crear sus propias funciones, el único requisito consiste en utilizar un estándar para los parámetros y las llamadas a las funciones desde el *workflow engine*, proporciona un API para poder ejecutar las funciones. En el caso de las funciones desarrolladas con PL/SQL éstas deberán cumplir con el siguiente formato estándar.

```
procedure <procedure name> (  
    itemtype in varchar2,  
    itemkey in varchar2,  
    actid in number,  
    funcmode in varchar2,  
    resultout out varchar2) is  
<declaraciones locales>  
begin  
if (funcmode='RUN') then  
    <Aquí van las instrucciones de la función>  
    resultout:='COMPLETE:<result>';  
    return;  
endif;  
if (funcmode='CANCEL') then  
    <Las instrucciones si se elige cancelar la función>  
    resultout:='COMPLETE';  
    return;  
endif;  
...
```

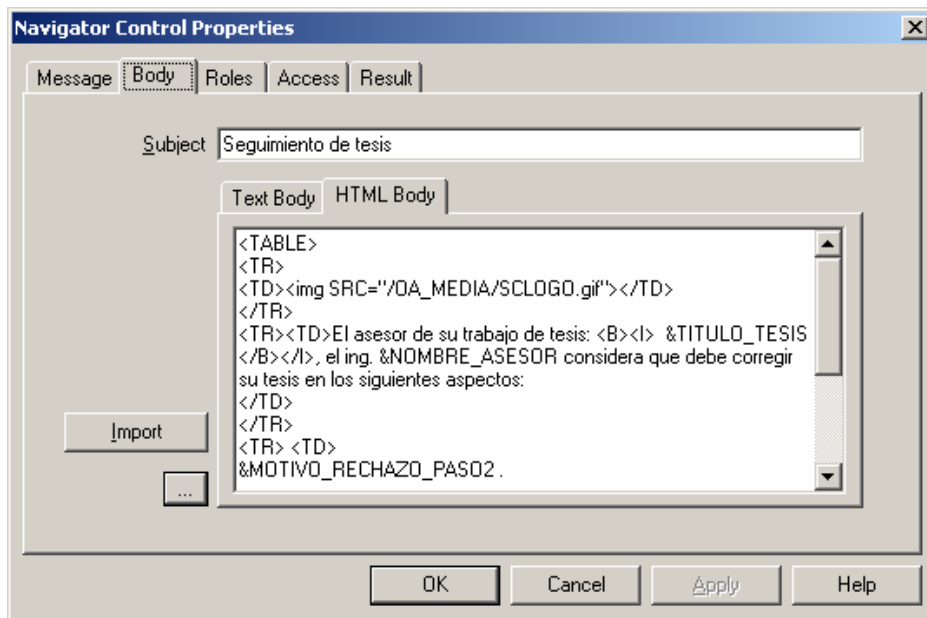
Los parámetros que recibe las funciones son: *itemtype* el cuál es el nombre interno del *item type*, definido en el *Oracle workflow builder*. *Itemkey* es un texto que representa un valor único generado por el *workflow engine* para el *item type* dado, *actid* es el código numérico de la actividad desde la que se llama a la función. *funcmode* determina el modo de ejecución de la función, el cuál puede ser *RUN* o *CANCEL*, *resultout* es el texto que retorna la función, si todo ha terminado correctamente el valor esperado es *COMPLETE*.

Mensajes: Un mensaje es el texto que se envía a través de una notificación a un usuario o a un *role*. Un mensaje está formado por dos componentes, uno estático y otro variable. Normalmente un mensaje tiene atributos asociados a él, estos atributos conforman las partes variables de un mensaje.

Las notificaciones pueden ser de texto plano puro o bien pueden tener un formato en HTML. Cuando necesitamos un mensaje para el cuerpo de una notificación nos colocamos en el nodo *messages* del navegador y elegimos la opción *new message*, esto nos mostrará las propiedades del mensaje. En la primera pestaña de la ventana de propiedades del mensaje colocamos el nombre interno del mensaje, el nombre que se mostrará en los diagramas *display name* y una descripción.

La pestaña del body será la que contenga el texto del mensaje, que puede ser texto plano o bien con formato de HTML. En la figura 22 mostramos el texto de un mensaje con formato HTML.

Figura 22. Propiedades de un mensaje



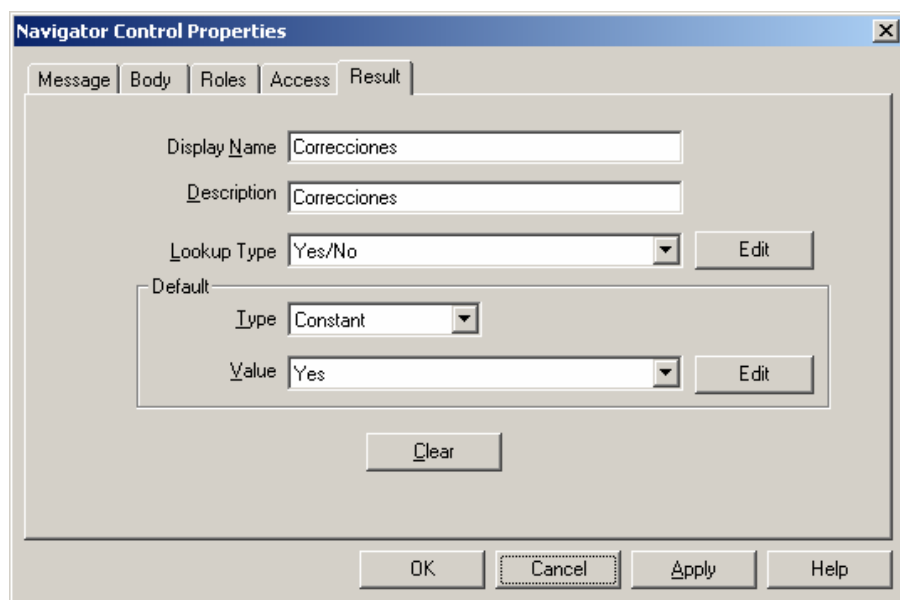
Como se aprecia en la figura 22, el cuerpo de un mensaje soporta cualquier etiqueta válida de HTML y para hacer referencia a información variable utilizamos los atributos definidos en el mensaje anteponiendo el símbolo “&” antes del nombre interno del atributo. Es importante resaltar que no podemos utilizar cualquier atributo definido en el *item type*, sólo podemos utilizar los atributos que se encuentren registrados en el mensaje.

Para registrar un atributo dentro de un mensaje debemos arrastrar el atributo desde el nodo de atributos en el navegador hasta el mensaje elegido. Otro método consiste en crear el atributo bajo el nodo del mensaje.

La única diferencia entre los atributos globales y los de mensaje es que éstos últimos tienen una propiedad nueva llamada *source*. Esta propiedad tiene dos posibles valores: *send* o *response*. En el primer caso se refieren a un atributo cuyo valor viene al mensaje para ser desplegado y no se capturará un nuevo valor. Por el otro lado un atributo *response*, es aquel cuyo valor se ingresará en el mensaje por parte del destinatario de la notificación.

Un mensaje también tiene un resultado. El resultado es la respuesta que da el receptor de una notificación al contenido del mensaje y dependiendo de ésta se determina el paso siguiente dentro de un proceso. En la figura 23 se observa la pestaña *result* en las propiedades de un mensaje.

Figura 23. Resultado de un mensaje

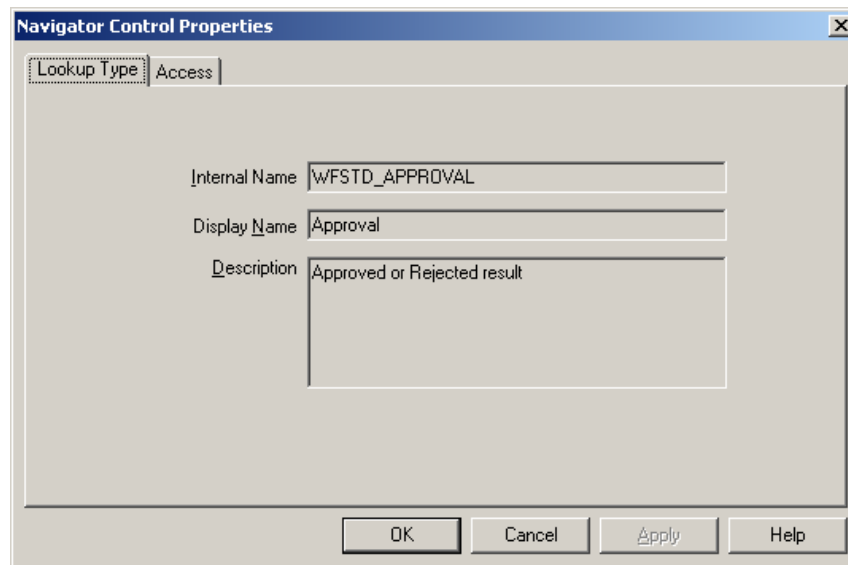


Como se aprecia en la figura el resultado de un mensaje tiene un nombre y una descripción. Además está asociado a una lista de valores *lookup type* que determina los posibles valores del resultado. También se pueden definir los valores por defecto del resultado de un mensaje. Estos valores están limitados a los existentes en la lista

de valores del resultado, los mensajes también tienen un plazo o *timeout*, es decir, una cuenta regresiva que se inicia desde que se envía la notificación hasta que el destinatario la atiende.

Listas de valores (*Lookup types*): Los *lookup types* tienen uno o más valores llamados códigos asociados a él. Una lista de valores puede ser referenciada por un mensaje, una notificación o una función, también puede ser el resultado de un proceso. Las listas de valores son vitales en todos los procesos, para crear una lista de valores se debe colocar en el nodo de *lookup types* del navegador y con el botón secundario del *mouse* elegir la opción *new*. Al definir una nueva lista de valores se debe indicar un nombre interno, un nombre para mostrar en los diagramas y una descripción. Esto se muestra en la figura 24.

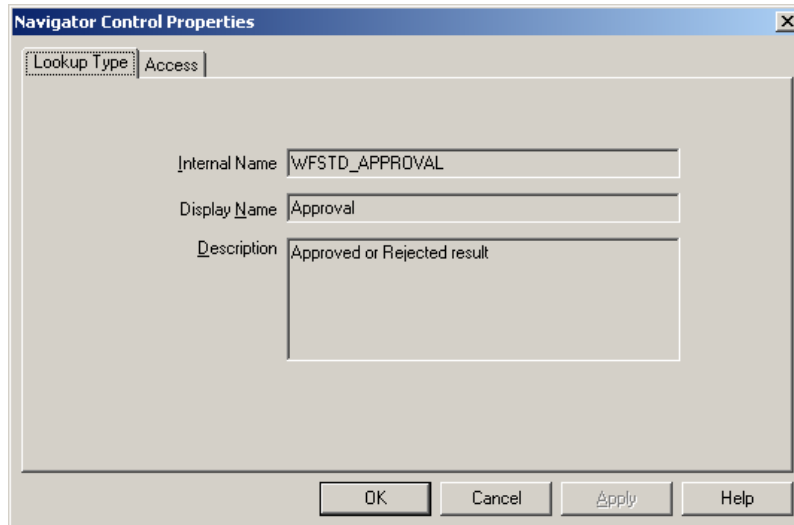
Figura 24. Propiedades de una lista de valores



En éste punto lo que tenemos es únicamente la definición de la lista de valores, hace falta definir los distintos valores válidos. Para esto es necesario definir un *lookup code*, esto es, un posible valor para nuestra lista. Al definir un nuevo código debemos indicar

el nombre interno, el texto que mostraremos en los diagramas y una descripción. Como se puede apreciar en la figura 25.

Figura 25. Propiedades de un código de una lista de valores



2.1.2.2 Diagramador de procesos.

El diagrama del flujo de trabajo se construye arrastrando los objetos del navegador hacia el diagrama de procesos. La ventana que contiene el diagrama de procesos y herramientas para su modificación se denomina Diagramador de procesos este representa gráficamente las actividades (íconos) y las transiciones (flechas) para un proceso en particular. Cada actividad es un nodo, un paso lógico que contribuye a la finalización del proceso. Es importante resaltar que los nodos son instancias de las definiciones que se tienen en el navegador, así por ejemplo, si se tiene una notificación en el navegador es posible tener varias instancias de ésta notificación en los diagramas.

Se pueden arrastrar actividades desde la ventana del navegador hacia el diagramador de procesos o bien crear las actividades directamente en la ventana del diagramador de procesos. Las propiedades para un nodo pueden ser consultadas o editadas haciendo doble *click* sobre el nodo en la ventana de procesos. Las transiciones se definen dibujando líneas de un nodo al siguiente utilizando el botón secundario del *mouse*.

Las notificaciones, funciones, eventos y subprocessos conforman los nodos de un proceso. No existen restricciones sobre la profundidad de ésta jerarquía. Para desplegar el diagrama de un subprocesso en la ventana del diagramador de procesos, se debe hacer doble *click* sobre el nodo de la actividad que representa el subprocesso en el proceso que lo contiene.

Transiciones: Las transiciones aparecen dentro del diagrama como flechas y representan la finalización de una actividad y el inicio de otra. Para una actividad que termina sin un resultado, cualquier transición que se dibuje desde ella simplemente aparece como una flecha hacia la próxima actividad, indicando que tan pronto como la actividad original acabe, se pasa a la siguiente actividad.

Para una actividad que tiene un resultado, se debe asociar la flecha de transición que se ha creado con uno de los posibles resultados de una actividad. Los resultados que una actividad puede retornar cuando se termina determinan cuál es la siguiente actividad en el proceso. Por ejemplo una actividad llamada “solicitud” que termine con el resultado “rechazado” pasa a la actividad “solicitud rechazada”.

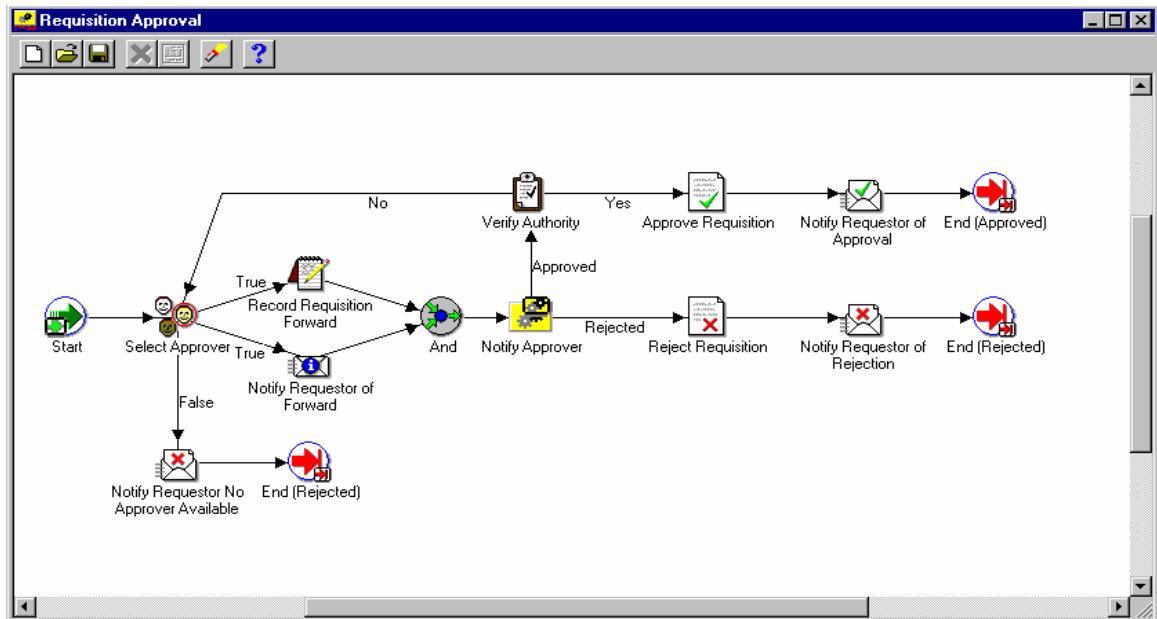
También se pueden crear transiciones *default*, *any* o *timeout* para una actividad que tiene un resultado. El *workflow engine* sigue una transición *default* si no hay otra transición que haga juego con el resultado de la actividad o bien un flujo marcado como *any* independientemente del resultado que retorne la actividad. Esto nos permite incluir una actividad genérica en nuestro proceso que el *engine* ejecute en paralelo, por ejemplo llevar una bitácora.

Por último puede seguir una transición marcada como *timeout* si el tiempo definido para que ocurra una notificación se ha alcanzado y la actividad aún no ha terminado. Normalmente se trazan transiciones *timeout* de una actividad de tipo notificación hacia otra actividad para forzar que el proceso continúe a la otra actividad si la notificación no se ha completado en un período de tiempo especificado.

Las actividades pueden tener múltiples transiciones para un mismo resultado dando por resultado bifurcaciones en paralelo. Un componente indispensable en cualquier diagrama de procesos son las funciones. Estas representan procedimientos almacenados en la base de datos en PL/SQL. Por lo que son las encargadas de incluir la lógica de programación dentro de un diagrama de procesos.

En el diagrama de procesos se representa el orden en que se ejecutan los diversos pasos y algunos pasos son automatizados completamente, por ejemplo, el envío de notificaciones. Sin embargo en cada paso puede ejecutarse un código, ya sea insertar datos en una tabla, consultar información de una tabla, ejecutar un ciclo, evaluar una condición compleja, etc. Por lo que se necesita ejecutar un programa. Esta labor se lleva a cabo a través de las funciones.

Figura 26. Diagrama de procesos



En la figura 26, el proceso inicia con la actividad *start*, el *workflow engine* se encarga de localizar ésta actividad e iniciar el proceso en ella. Luego hay una transición que no tiene un resultado, por lo que únicamente indica el orden de ejecución. Este orden nos lleva a la actividad *select approver*. Esta es una función, es decir, un procedimiento almacenado dentro de la base de datos. Al llegar aquí el *workflow engine* ejecutará el procedimiento almacenado y esperará una respuesta que puede ser: verdadero o falso.

En éste punto nuestro proceso tiene varias transiciones. Dos de ellas marcadas con el resultado *true* y una con *false*. Esto significa que si el resultado de la función *selected approver* es verdadero el *workflow engine* ejecutará en paralelo las dos actividades siguientes: *record requisition forward* y *notify requestor of forward*. Y si el resultado es falso, entonces sólo ejecutará la actividad *notify requestor no approver available* y al final de proceso con un resultado de rechazo, a este tipo de transiciones se les conoce como flujos en paralelo.

Las actividades en paralelo pueden tener diferentes duraciones por lo que cualquiera de ellas puede terminar primero, sin embargo ambas tienen transiciones “*default*” hacia una actividad *and* lo que implica que para continuar, ambas deben terminar. La actividad *record requisition forward* es una función que guarda información de la solicitud en tablas de la base de datos.

Al terminar las dos actividades en paralelo y completarse, de ésta manera, la actividad *and* se ejecuta un subproceso llamado *notify approver*. Esta actividad es realmente un subproceso, es decir está formado por múltiples actividades, pero se puede tomar como una actividad que tiene una salida.

Esto es muy común en los procesos complejos, ya que si se detallara en un mismo nivel todas las actividades se perdería la claridad del diagrama. Por lo que se crean subprocesos para simplificar el diagrama. Luego en otro diagrama se completa el subproceso. La salida del subproceso puede ser: *rejected* o bien *accept* por lo que se crean transiciones para cada posible valor.

En el caso que el resultado del subproceso sea *accept* la transición con ese valor nos lleva a una función llamada: *verify authority* quien se encargará, de acuerdo con un código desarrollado, de verificar que quien emitió la autorización esté en la capacidad de hacerlo. Este código se ejecutará y podrá tener uno de dos resultados posibles: *yes* o *no*.

En caso que el resultado sea *yes* se ejecutará una función llamada *approve requisition* en la que se podrá llevar, por ejemplo, un contador de requisiciones aprobadas o cualquier otro código necesario. Y finalizará con una notificación al solicitante original indicando que su solicitud fue aprobada.

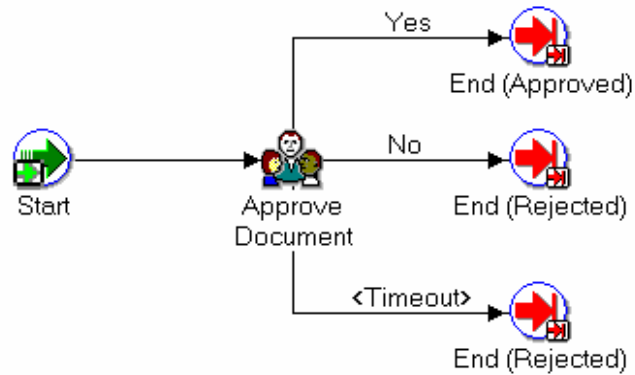
En el caso que el resultado sea negativo, se volverá a ejecutar el código de la función *select approver* para generar a un nuevo aprobador de la requisición. En este caso ocurre un flujo denominado cíclico ya que es como un ciclo en programación donde dada una condición se retorna a un paso anterior y pueden ejecutarse varias veces estos pasos, a cada ejecución de éste ciclo se le conoce como iteración.

El subproceso *notiify approved* puede también generar una salida negativa produciendo como resultado un *rejected* por lo que se ejecutará la transición con éste valor de resultado y se ejecutará luego la función: *reject requisition* donde se podrá llevar el control de las solicitudes rechazadas. Y por último la notificación de rechazo para el solicitante.

Del análisis del proceso del ejemplo se observa que las funciones son las que juegan el papel más importante ya que tienen la lógica del proceso. Las transiciones únicamente indican el orden en el que se debe llevar a cabo ésta lógica y los casos en los que se puede ejecutar. Y la comunicación interactiva entre los usuarios finales se lleva a cabo a través de las notificaciones.

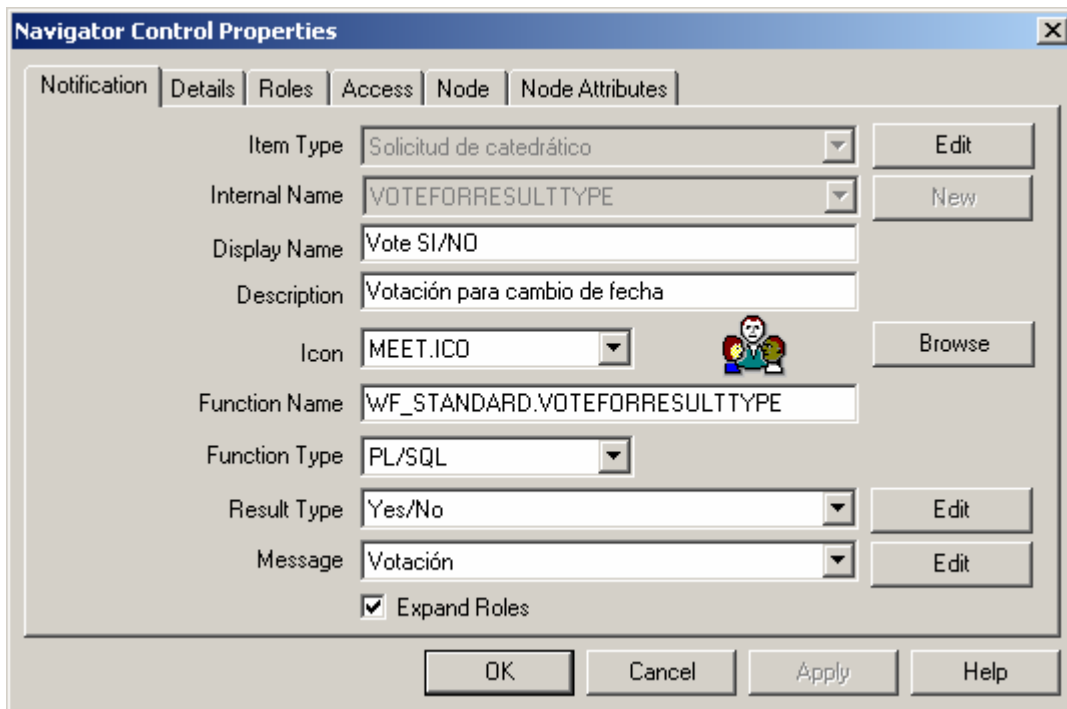
Es por esto se debe colocar mayor atención en la forma en que se codifican las funciones y la manera en que las podemos integrar en nuestro proceso.

Figura 27. Diagrama de proceso de una votación



En la figura 27, el proceso inicia en la actividad *start* y luego hacia la notificación *approve document*. Esta notificación está asociada a una función de PL/SQL la que puede tener varios resultados: *yes*, *no* y *timeout*. En el caso de una notificación normal, al recibirse la primera respuesta con un valor, sea éste *yes*, *no* o *timeout*, se continúa con el flujo. En el caso de una votación el proceso es diferente. En éste caso se debe esperar a que todos los destinatarios contesten a la notificación y luego determinar cuál fue la respuesta que tuvo una mayoría y dependiendo de esto se continúa con el flujo.

Figura 28. Propiedades de una notificación para votación



En la figura 28, se puede apreciar las propiedades de la notificación “solicitud de catedrático” la cuál será de tipo votación. Para lograr esto se debe asociar la notificación a una función que realizará el conteo de los votos y determinará la salida de la función. En éste caso la notificación tiene seleccionada una función predefinida llamada: *WF_STANDARD.VOTEFORRESULTTYPE*.

La función recibe de parámetros el proceso que se modeló (*itemtype*), la instancia de éste proceso (*itemkey*) y además el valor del atributo *voting option*. Este atributo puede tener uno de tres posibles valores:

WAIT_FOR_ALL_VOTES: Evalúa la votación después que todos los votos han sido emitidos o bien una condición de *timeout* cierra la votación. Cuando ocurre un *Timeout*, el porcentaje es calculado como un porcentaje de los votos emitidos.

REQUIRE_ALL_VOTES: Evalúa la votación después de que todos los votos han sido emitidos. Si ocurre un *timeout* y no se han emitido todos los votos, entonces el proceso continúa en la transición de *timeout* estándar. Los votos son calculados como un porcentaje de los usuarios notificados para votar.

TALLY_ON_EVERY_VOTE: Evalúa la votación después de cada voto o bien cuando una condición de *timeout* cierra la votación. El porcentaje de votos es calculado como un porcentaje de los usuarios notificados. Si ocurre un *timeout* el porcentaje se calcula a partir de los votos emitidos.

2.1.3 Oracle server

El núcleo de todo *workflow* management system es el *workflow engine*. En el caso del *oracle workflow* ésta capa se implementa dentro de una instancia de base de datos, a ésta instancia se le llama *oracle server*. Dentro del *oracle server* se encuentran diferentes componentes como el servicio de directorio, el manejo de eventos, el servicio de notificaciones y el *workflow engine* como se ilustra en la figura 15.

2.1.3.1 Servicio de directorio.

Oracle workflow no mantiene su propio repositorio de usuarios, en lugar de esto el servicio de directorios consiste en una serie de vistas que mapean al repositorio de usuarios y roles elegido. El servicio de directorio puede mapearse a un servidor de directorio que cumpla con el estándar de *lightweight directory access protocol*

(LDAP) como es el *oracle internet directory* o bien al repositorio de usuario de la base de datos.

Para realizar éste mapeo se crean una serie de vistas que contienen la información necesaria para identificar a cada usuario y role del sistema. Las principales de éstas vistas son: *WF_USERS*, *WF_ROLES* y *WF_USER_ROLES*.

La vista *WF_USERS* contiene información sobre los usuarios del sistema, ésta información incluye: Nombre interno, nombre visible, preferencias de notificación, lenguaje, dirección de correo, el sistema origen del usuario (directorio LDAP o base de datos), estatus (activo o inactivo) y fecha de expiración de cuenta.

Por otro lado la vista *WF_ROLES* representa los roles o grupos de usuarios. Contienen las columnas: nombre interno, nombre visible, preferencias de notificación, lenguaje, dirección de correo, el sistema origen del usuario (Directorio LDAP o base de datos), estatus (activo o inactivo) y fecha de expiración de cuenta.

Además existe una vista que determina la relación entre los grupos o roles y los usuarios, ésta vista es: *WF_USER_ROLES*, la cuál tiene: nombre interno del usuario, sistema origen del usuario, nombre interno del role, sistema origen del role.

Además de éstas vistas el sistema de directorios cuenta con un API para poder comunicarse con él. El paquete se llama *WF_DIRECTORY* y tiene las siguientes funciones y procedimientos definidos:

CreateAdHocUser()

CreateAdHocRole()

AddUsersToAddHocRole()

GetRoleDisplayName()

SetAdHocUserStatus()

SetAdHocRoleStatus()

SetAdHocUserExpiration()

SetAdHocRoleExpiration()

SetAdHocUserAttr()

SetAdHocRoleAttr()

RemoveUsersFromAdHocRole()

Los nombres de las funciones son lo bastante descriptivos para darnos una idea de la funcionalidad que proporcionan.

2.1.3.2 Sistema de manejo de eventos

Un *workflow management system* debe proporcionar una interfase para poder comunicarse con otros componentes de software. Es por esto que existe el sistema de manejo de eventos que implementa, a través de *advanced queue* (AQ) de la base de datos, el sistema de colas para manejar la notificación de eventos.

Oracle workflow permite establecer suscripciones a eventos de negocios, los cuales activarán el flujo de trabajo o habilitarán mensajes para que se propaguen de un sistema a otro cuando un evento de negocios ocurra.

Se pueden comunicar eventos pertenecientes al mismo sistema, es decir, dentro de la misma base de datos o con sistemas externos. De esta manera, se pueden implementar la integración de mensajes punto a punto o bien escenarios de integración más complejos. Se puede modelar procesos de negocios que incluyan enrutamientos complejos y procesamiento de reglas para manejar evento con flexibilidad.

Cuando *oracle workflow* necesita notificar un evento a una aplicación externa, coloca un mensaje en la cola de salida (“*outbound*”). Esta cola puede ser consumida por agentes externos que soporten *Java* y *Common Object Request Broker (CORBA)* o bien procesos a nivel de sistema operativo.

Cuando el proceso externo necesita notificar de un evento al *oracle workflow* coloca un mensaje en la cola de entrada (“*inbound*”) la cual es consultada continuamente por el background *engine*.

Estas colas de entrada y de salida se implementan utilizando la opción de *advanced queue* de la base de datos. *oracle workflow* presenta además un API para poder interactuar directamente con las colas para poder, programáticamente, modificarlas. Dentro de las funciones del API podemos mencionar las siguientes:

EnqueueInbound()
DequeueOutbound()
DequeueEventDetail()
PurgeEvent()
PurgeItemType()
ProcessInboundQueue()
GetMessageHandle()
Deferred_queue()
Inbound_queue()
Outbound_queue()
ClearMsgStack()
CreateMsg()
WriteMsg()

SetMsgAttr()

SetMsgResult()

2.1.3.3 Servicio de notificaciones

El servicio de notificaciones es un conjunto de paquetes de PL/SQL que se encargan de enviar las notificaciones hacia los diferentes destinatarios. Cuando los destinatarios reciben las notificaciones, normalmente tienen que tomar alguna acción al respecto, que podría ser rechazar, aceptarla o re-enviarla a alguien más.

Cuando el destinatario recibe la notificación, ésta cuenta con links para las diferentes acciones que se hayan definido para la notificación. Cuando el destinatario lo considere adecuado toma la acción necesaria. En ese momento el servicio de notificaciones recibe la acción y efectúa el procesamiento que se tenga definido para la acción y realiza las notificaciones necesarias.

Oracle workflow extiende el límite de las notificaciones automáticas de los procesos de flujo de trabajo, para incluir usuarios externos que tengan una dirección de correo electrónico o usuarios de Internet.

Oracle workflow permite a las personas recibir notificaciones de procesos que espera una acción o respuesta, vía correo electrónico o el sistema de notificaciones interno de *Oracle workflow*. Las notificaciones permiten tener una lista de notificaciones pendientes que representan procesos en espera por una respuesta.

2.1.4 Servidor de aplicaciones

En ésta capa se encuentra el servidor de aplicaciones donde se publicarán las aplicaciones que iniciarán los flujos de trabajo y pueden servir también para recibir las notificaciones. El servidor de aplicaciones que utiliza *oracle workflow* es el *oracle application server 10g* (AS 10g) que cumple con el estándar de *Java 2 enterprise edition* (J2EE).

Este servidor tiene varios módulos, a continuación se describe únicamente los que tienen relación con la atención de solicitudes provenientes de *Oracle workflow*

2.1.4.1 Http server

El servidor *web* de *Oracle* está basado en el servidor *web* Apache que es un estándar *defacto* en los servidores *web*. Por su arquitectura modular, sus funcionalidades pueden extenderse fácilmente agregando módulos, de ésta manera se agrega, entre otros, el módulo de *mod_plsql*.

El servidor *web* es el punto de contacto entre los usuarios finales y el *oracle workflow*. Este vínculo se realiza a través de lo que se denomina un *database access descriptor* (DAD). Esto no es más que un path virtual que se define en el servidor *web* y que es manejado a través del módulo de *mod_plsql*, de tal manera que las solicitudes que se hagan a éste url serán atendidas por un paquete en la base de datos que generará el HTML dinámicamente.

Para poder generar el HTML se necesita tener instalado el *PL/SQL Web toolkit* que está formado por una serie de librerías de PL/SQL que generarán dinámicamente la página *web*.

2.1.4.2 Oracle Content Management SDK (CMSDK)

El *oracle content management* SDK permite leer, editar, copiar, bloquear y compartir el contenido almacenado en el servidor de aplicaciones a través de un gran rango de protocolos como: FTP, SMB, WebDAV, NFS, AFP, IMAP4, and SMTP.

Oracle content management SDK está completamente integrado con *oracle text*, *oracle workflow*, *oracle intermedia* y *oracle advanced queueing*. Permitiendo de ésta manera comunicarse a través de un API con éstos componentes. En el caso del *Oracle workflow* esto se logra a través de la definición de eventos.

2.1.4.3 Notificador de Correo

Oracle workflow puede integrarse con cualquier servidor de correo electrónico que soporte IMAP4 y SMTP. También puede utilizarse el servicio de notificaciones de *Oracle workflow* desarrollado en PL/SQL.

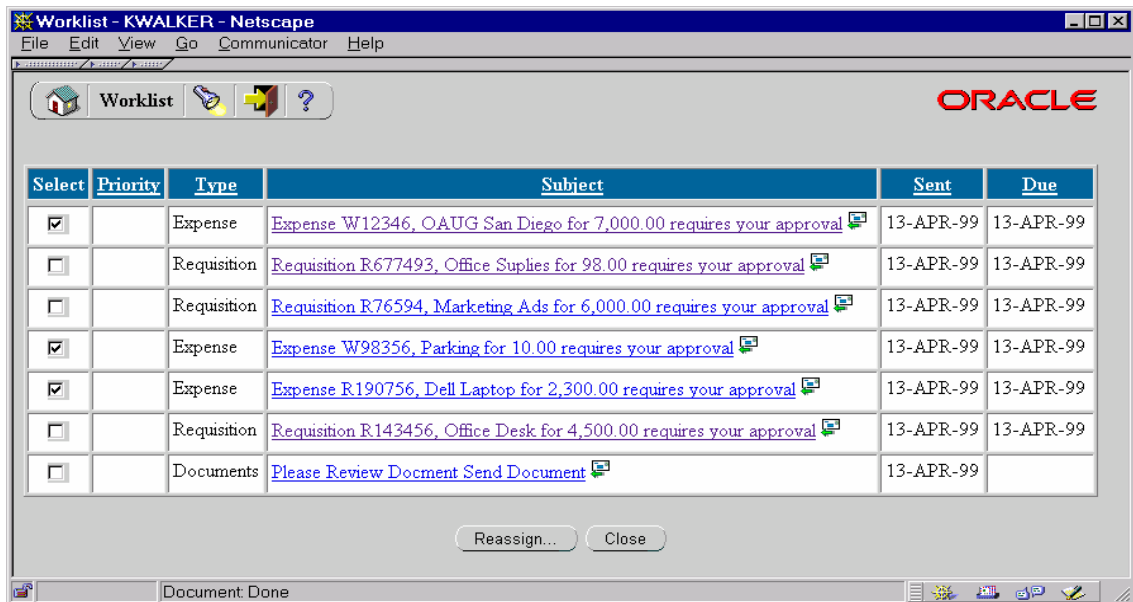
2.1.5 Usuarios finales

Los clientes finales se conectarán al servidor de aplicaciones tanto para iniciar un flujo de trabajo como para aprobar, rechazar o votar por las notificaciones. También puede recibir las notificaciones a través de un cliente de correo. Los administradores contarán con herramientas de monitoreo y análisis de los procesos.

2.1.5.1 Listas de Trabajo

Las listas de trabajo muestran una lista de las notificaciones abiertas para un usuario en particular. Las listas de trabajo pueden consultarse a través de una página *web* o a través de un cliente de correo electrónico.

Figura 29. Lista de Trabajo

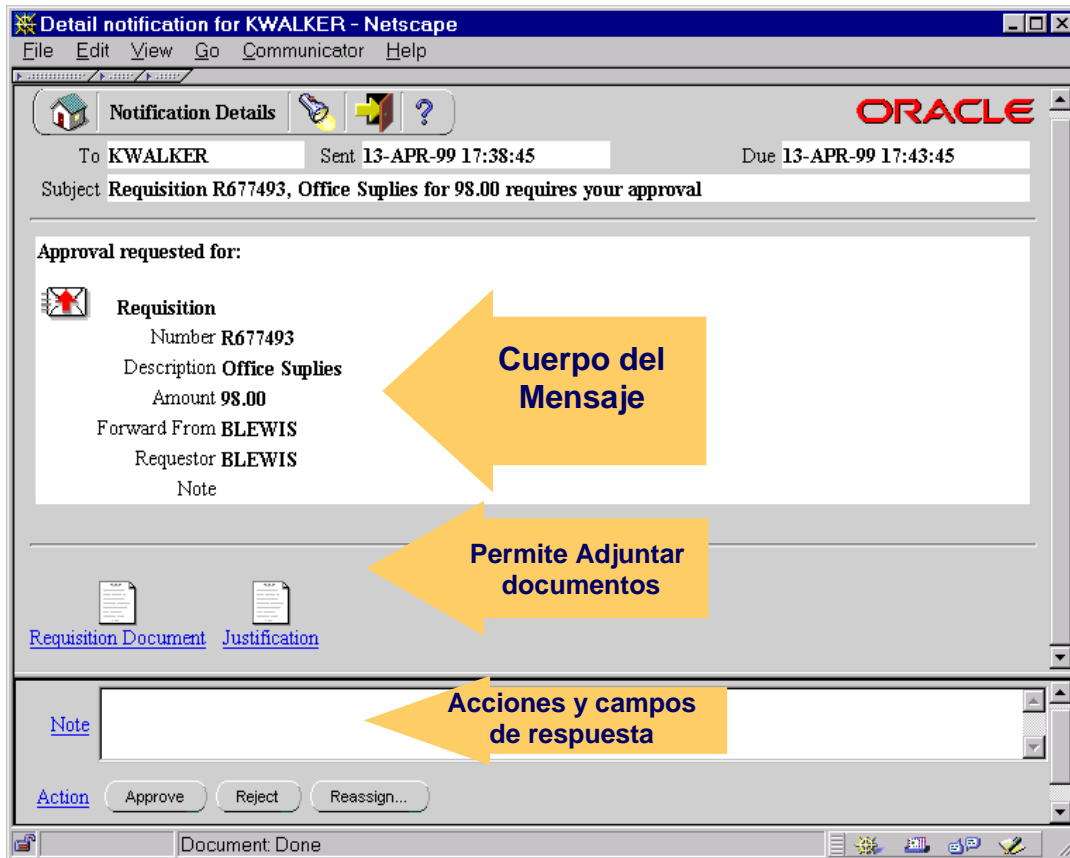


Select	Priority	Type	Subject	Sent	Due
<input checked="" type="checkbox"/>		Expense	Expense W12346, OAUG San Diego for 7,000.00 requires your approval	13-APR-99	13-APR-99
<input type="checkbox"/>		Requisition	Requisition R677493, Office Suplies for 98.00 requires your approval	13-APR-99	13-APR-99
<input type="checkbox"/>		Requisition	Requisition R76594, Marketing Ads for 6,000.00 requires your approval	13-APR-99	13-APR-99
<input checked="" type="checkbox"/>		Expense	Expense W98356, Parking for 10.00 requires your approval	13-APR-99	13-APR-99
<input checked="" type="checkbox"/>		Expense	Expense R190756, Dell Laptop for 2,300.00 requires your approval	13-APR-99	13-APR-99
<input type="checkbox"/>		Requisition	Requisition R143456, Office Desk for 4,500.00 requires your approval	13-APR-99	13-APR-99
<input type="checkbox"/>		Documents	Please Review Document Send Document	13-APR-99	

Como se muestra en la figura 29, al consultar el detalle de una notificación se puede observar el cuerpo de la notificación el cuál sustituye las variables que se hayan definido en el *workflow* por los valores apropiados. También se tiene la opción de adjuntar documentos que serán links hacia el manejador de contenido del servidor de aplicaciones.

Un ejemplo del detalle de una notificación puede verse en la figura 30.

Figura 30. Detalle de una Notificación



En el cuerpo del mensaje se puede incluir el cuerpo de la notificación que incluye texto fijo que constituye la plantilla de la notificación. Los valores de los atributos han sido referenciados con el símbolo ampersand (&) para efectos de despliegue.

2.1.5.2 Monitor de Procesos

El monitor de procesos es una herramienta desarrollada en *java* que permite observar y administrar el progreso de una instancia de un proceso de *workflow*. Dentro de las opciones del monitor, la herramienta presenta un diagrama del proceso el cuál muestra, en una ventana, el proceso creado en el *workflow builder*. También provee atributos visuales sobre el status de los procesos y sus actividades.

Una actividad puede estar resaltada con un marco de colores que indican un estado distintivo. Rojo significa que la actividad está en un estado de error. Verde indica que la actividad está en progreso y amarillo cuando la actividad está suspendida. Una flecha de transición puede tener una línea gruesa verde que indica que el flujo del proceso ha utilizado ésta transición o bien una línea negra que indica que no se ha utilizado. El administrador también puede modificar a la instancia del proceso que se está observando. Para esto cuenta con los siguientes botones:

Abort Process: Llama a la función *WF_ENGINE.AbortProcess* para abortar el proceso seleccionado y cancelar cualquier notificación pendiente.

Suspend Process: Llama a la función *Calls WF_ENGINE.SuspendProcess* para suspender el proceso seleccionado.

Resume Process: Llama a la función *WF_ENGINE.ResumeProcess* para reanudar el procesos seleccionado de un estatus suspendido a ejecución normal.

Reassign: Llama a la función *WF_ENGINE.AssignActivity* para reasignar la actividad de notificación seleccionada a un diferente usuario.

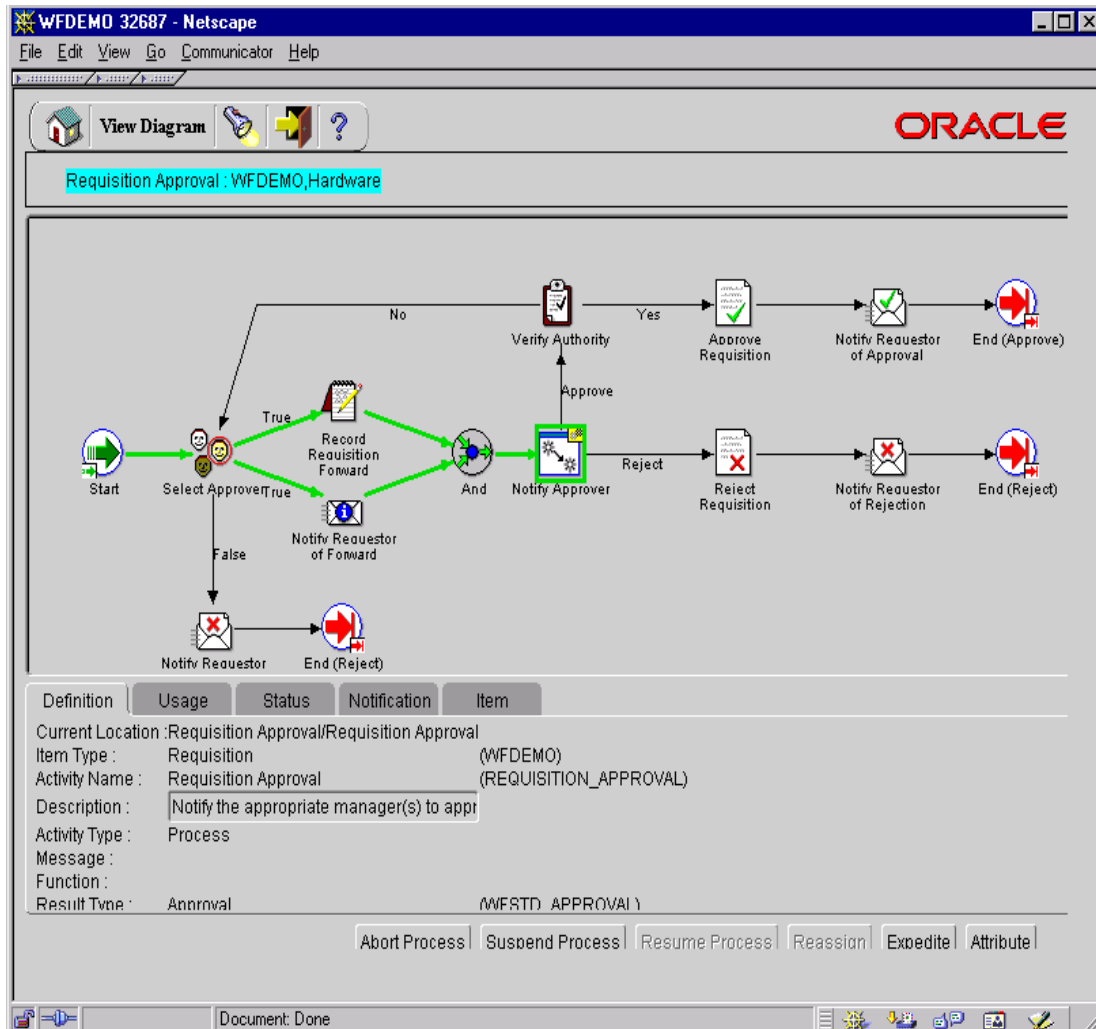
Expedite: Llama a la función *WF_ENGINE.HandleError* para alterar el estado de una actividad en estado de error, o para deshacer la actividad seleccionada y las subsiguientes.

Para poder utilizar el monitor de procesos es necesario que el navegador soporte *Java* 1.1.4. El url para acceder al monitor de procesos debe ser:

```
http://host:puerto/pls/wf/wf_monitor.html?x_item_type=<item_type>&x_item_key
=<item_key>&x_access_key=<access_key>
```

Donde *<item_type>* es el nombre del proceso que se desea monitorear, y *<item_key>* y *<access_key>* son los valores numéricos que identifican de forma única una instancia del proceso de otra instancia del mismo proceso. Este URL está protegido por autenticación, por lo que si el usuario no tiene los privilegios adecuados no podrá monitorear el proceso. El monitor de instancias de procesos se muestra en la figura 31.

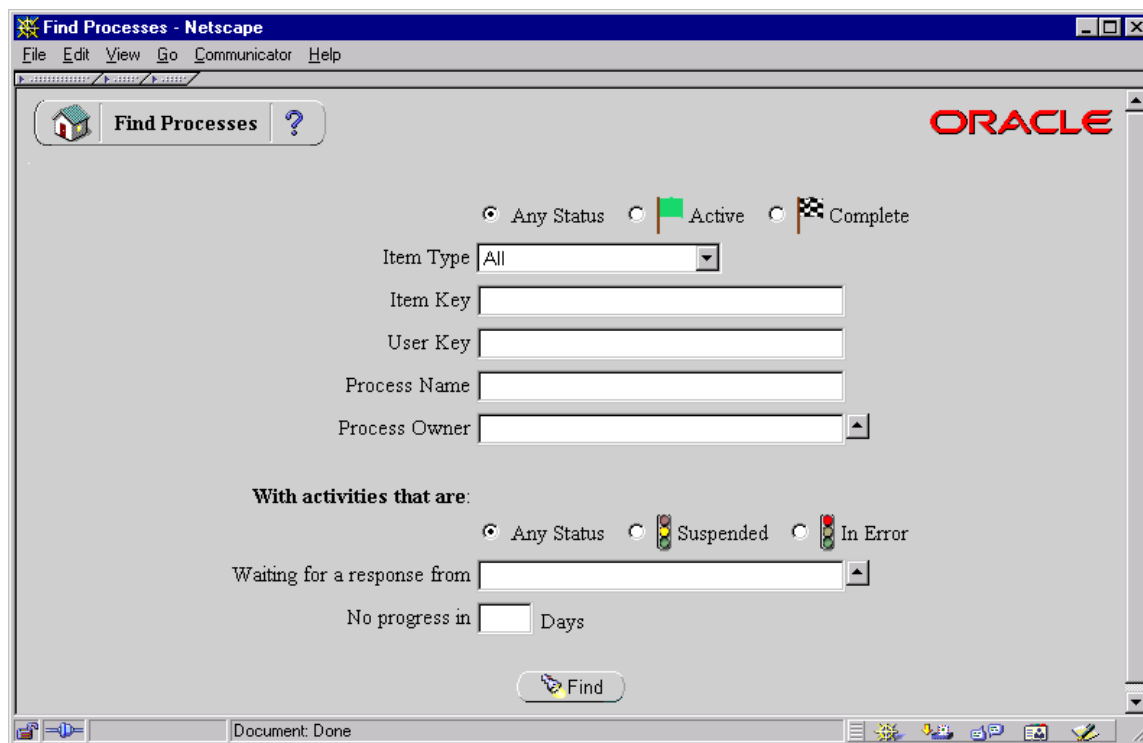
Figura 31. El Monitor de instancias de procesos



Si se desconocen los valores de `<item_type>` y `<access_key>` se puede realizar una búsqueda de las instancias de un proceso a través del URL http://host:puerto/pls/wf/wf_monitor.find_instance.

En ésta página se puede filtrar la búsqueda de una instancia, por el estado del proceso, *Item type*, *Item Key*, *User Key*, Nombre del proceso o bien por el dueño del proceso. Además de restringir por el estado de las actividades del proceso o si alguna está a la espera de una respuesta. Esta página se muestra en la figura 32.

Figura 32. Búsqueda de instancias de procesos



3. CASO PRÁCTICO: PROCESOS ADMINISTRATIVOS DE LA FACULTAD DE INGENIERÍA DE LA UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

Antes de utilizar el *oracle workflow builder* para diagramar un proceso es necesario planificar qué es lo que el proceso necesita llevar a cabo. Determinar, en particular, qué actividades deben ocurrir, el orden en que deben ejecutarse, los resultados de éstas y los diferentes caminos que puede tomar el proceso, quién necesita ser informado y que cosas necesita saber. A ésta etapa se le conoce como análisis de requerimientos.

Luego de completar el análisis de los requerimientos es necesario realizar el diseño de cada uno de los procesos de acuerdo con el estándar definido por la *workflow management coalition* (WfMC). Esta etapa incluye la diagramación de cada proceso de acuerdo con el estándar, utilizando el *oracle workflow builder*, así como la definición de los roles y los usuarios para las notificaciones y el manejo de los documentos.

Por último debemos publicar los procesos en un *workflow management system* (WFMS), en éste caso utilizando el *oracle workflow* y programar las funcionalidades específicas que requieran una integración con otros sistemas o bien que requieran un cálculo o procesamiento adicional. En ésta etapa deben realizarse, también, las pruebas de funcionalidad de los procesos.

Para ilustrar mejor los conceptos en la definición y publicación de procesos de *workflow* se realizó un caso práctico en el que se modelarán y publicarán tres procesos administrativos de la facultad de ingeniería de la Universidad de San Carlos de

Guatemala. Este caso práctico además de ejemplificar el uso de flujos de trabajo en casos concretos, esperamos que ayudará a los catedráticos, estudiantes y al Director de Escuela a realizar los procesos de revisión de tesis, consulta a los catedráticos y consulta a los alumnos de una forma más automática, segura y controlada.

3.1 Metodología utilizada

La workflow management coalition (WfMC) no tiene un estándar definido para el análisis de requerimientos y deja a cada desarrollador libre para definir los requerimientos de la manera que crea más conveniente. Sin embargo, estamos convencidos que ésta etapa es una de las más importantes ya que es fácil cometer errores y definir erróneamente los requerimientos de cada proceso. Para facilitar a los desarrolladores el completar ésta etapa sin errores se definió una metodología así como una serie de instrumentos que la complementan.

Esta metodología consiste en tres etapas bien definidas: En primer lugar tenemos el análisis de requerimientos que incluye la identificación y descripción de las necesidades de intercambio de información así como la definición de usuarios y roles involucrados. Este paso es independiente del *workflow management system (WFMS)* que se elija para ejecutar los procesos.

El segundo paso es el diseño de los procesos, cumpliendo con el estándar definido por la *workflow management coalition (WfMC)* que se mostraron en el primer capítulo de ésta tesis utilizando como herramienta el *oracle workflow builder*.

Y el tercer paso consiste en la implementación funcional de los procesos en un *workflow management system (WFMS)* que en nuestro caso será *oracle workflow* y la programación de las funciones específicas utilizando para el efecto paquetes de PL/SQL almacenados en la base de datos *oracle*.

3.1.1 Análisis de requerimientos

3.1.1.1 Identificación y descripción de las necesidades de información

Este es el primer paso de nuestra metodología y en él se deben identificar qué necesidades existen en la empresa para poder luego modelarlas y crear procesos de flujos de trabajo para satisfacerlas. La manera en que se realizará éste paso consiste proporcionar una descripción sucinta del proceso.

En nuestro caso práctico tenemos tres necesidades de intercambio de información que, conforme a nuestra metodología describiremos.

Revisión de examen: En éste proceso los estudiantes podrán solicitar revisión de la nota de un examen al catedrático del curso. En éste proceso el estudiante debe indicar de que curso y sección es la nota, y una pequeña descripción del porque cree que la nota no es correcta. La lista de cursos de los cuales un estudiante puede solicitar revisión está limitada a los cursos que tenga asignados el estudiante. Automáticamente se determinará quien es el catedrático del curso y se le solicitará la revisión.

Si al examinar la descripción, el catedrático encuentra razones de peso para hacer la revisión podrá calcular la nueva nota y enviarla junto con un mensaje de aprobación y el valor de la nueva nota al estudiante. Si por el contrario no hay suficientes argumentos para cambiar la nota el catedrático o bien al realizar la revisión no se ha amerita a cambiar la nota, se le enviará al estudiante un mensaje de rechazo.

Si durante 10 días el catedrático no ha rechazado ni aprobado la solicitud, ésta será enviada automáticamente al director de escuela, quien podrá solicitar al catedrático que le de seguimiento a la revisión.

El director de escuela también será capaz de revisar el status de las notificaciones y determinar los tiempos de respuestas más adecuados, para poder modificar el flujo de trabajo y alargar o acortar los tiempos. También será capaz de guardar un histórico de las decisiones de tal manera que queda una constancia de la solicitud y de la respuesta que se ha dado, para que en casos similares se tenga un precedente.

Revisión de tesis: Este proceso ayudará a darle un mejor seguimiento al proceso de revisión de tesis y mandar los mensajes de forma automática a quien corresponda.

Este proceso inicia cuando un estudiante toma el curso propedéutico de tesis, en ese momento ya está capacitado para poder dar inicio al proceso. Sólo aquellos estudiantes que cumplan con éste requisito podrán acceder a éste proceso.

Como primer paso, el estudiante debe elegir un tema de su interés y que cumpla con los requisitos de un trabajo de graduación. Luego debe realizar un protocolo de trabajo de graduación, el cuál debe presentarse, junto a una carta de aceptación de su asesor a la comisión de revisión de privados y tesis.

La comisión evalúa la solicitud y revisa el protocolo y determina si cumple con los requisitos necesarios para ser aceptado y si el tema que desea tratar no ha sido ya tomado por otro estudiante. Opcionalmente puede trasladar al estudiante los cambios que sean necesarios para incrementar el valor académico y/o beneficio que se pueda obtener al finalizar el trabajo. El proceso continúa en aprobación o rechazo del protocolo de trabajo de graduación. Si se rechaza el protocolo, el proceso finaliza y se debe iniciar de nuevo.

Cuando el protocolo cumple con todos los requisitos necesarios la comisión le envía una notificación de la aprobación de su protocolo y debe reunirse con su asesor para planificar las revisiones periódicas de su trabajo de graduación.

Al recibir el estudiante la notificación de que su protocolo ha sido aceptado, se debe iniciar la elaboración de su trabajo de graduación de acuerdo a las especificaciones indicadas en las guías de elaboración de trabajos de graduación.

En éste punto el proceso entra en un ciclo de revisiones, por parte del asesor, para ir mejorando continuamente el trabajo de graduación. Por lo tanto el estudiante deberá enviar el documento a su asesor para su revisión. El asesor evalúa el trabajo y cuando ya se han efectuado todas sus recomendaciones da su aprobación.

Al contar el estudiante con la aprobación del asesor se le envía el trabajo de graduación a la comisión de revisión de tesis. Quienes nombran a un profesional, para que haga una revisión final del trabajo de graduación. Este profesional hace sus recomendaciones para que el estudiante corrija su trabajo de graduación y cuando ya están efectuadas todas las recomendaciones, el profesional da su aprobación.

Cuando el encargado de la revisión está de acuerdo se entrega un mensaje de aprobación y anexo una ficha de seguimiento y una carta, con lo que se finaliza el proceso.

El proceso de revisión de tesis, tiene más pasos, como son la aprobación del decano y lingüística pero éstas no serán modeladas en éste ejemplo.

Solicitud del Catedrático a los estudiantes: En algunos casos el catedrático de un curso es quien necesita solicitar a los estudiantes algún cambio, que puede ser de horario, de salón o de la fecha entrega de una tarea.

En éstos casos se solicitará a los estudiantes que aprueben la solicitud del catedrático, y si la mayoría de los estudiantes del curso está de acuerdo en el cambio se enviará una notificación al catedrático que su solicitud ha sido aprobada, de lo contrario se rechazará.

Los procesos que se modelarán tienen cada uno formas de notificación diferentes, por niveles (en el caso de la revisión de examen), con varios sub-procesos (revisión de tesis) y con una aprobación por votación (solicitud de catedrático a los estudiantes) por lo que son representativos de los tipos más importantes de procesos que podemos modelar.

3.1.1.2 Identificación de los usuarios y roles involucrados

El siguiente paso es la completa identificación de los usuarios y roles involucrados en cada uno de los procesos. Es muy importante identificar claramente los roles para evitar duplicidad o ambigüedad en los roles. Realizaremos ésta definición en cada uno de los procesos modelados.

Para facilitar ésta tarea se creó un documento útil para definir los roles y los usuarios al que llamaremos tabla de roles y usuarios. En ésta tabla definiremos los roles que intervienen en un proceso de manera que fácilmente podamos indicar a qué roles pertenece un usuario nuevo o uno ya existente. La idea es que éstas definiciones no queden únicamente en la mente del analista sino que quede una constancia escrita de las definiciones de cada rol y usuario en el sistema.

A continuación mostramos las tablas de roles y usuarios para los tres procesos administrativos de la Facultad de Ingeniería de la Universidad de San Carlos.

Cada tabla contiene tres columnas la primera es role bajo la cuál colocaremos los nombre que le daremos a los diferentes grupos de usuarios de nuestros procesos. La segunda es usuario en ella colocaremos los nombres de los usuarios específicos de nuestros sistema. Al decir usuarios específicos nos referimos a usuarios con nombre y apellido o con algún otro atributo que sirva para identificarlos de forma única. Esta columna puede estar vacía en ese caso nos referimos a la definición de un role. En la tercera columna colocaremos la descripción, lo más ajustada a la realidad para definir tanto a los grupos de usuarios como a los usuarios individuales.

En nuestro ejemplo no identificaremos a nuestros usuarios con nombre y apellido o con otro atributo, sino lo colocaremos entre los símbolos de mayor que y menor que (< >). Esto es debido a que se trata de un ejemplo, sin embargo el instrumento está ideado para que sirva para identificar a los usuarios individuales plenamente.

Tabla I. Roles del proceso de revisión de examen

Role	Usuario	Descripción
Catedráticos		Grupo de usuarios que imparten, como titulares, al menos un curso en la carrera de Ciencias y Sistemas en el presente semestre.
Estudiantes		Grupo de usuarios inscritos en la carrera de ciencias y sistemas que tienen asignado al menos un curso en el presente semestre.
dirección		Grupo de usuarios que cumplen roles administrativos de dirección en la dirección de escuela. Incluye al director de escuela.
catedráticos	<catedrático>	Usuario específico que pertenece al grupo de catedráticos que está vinculado a un curso y una sección en particular.
estudiantes	<estudiante>	Usuario específico que pertenece al grupo de estudiantes.
dirección	<director>	Usuario específico que pertenece al grupo de dirección. Tiene

		funciones de administración y monitoreo del proceso.
--	--	--

Tabla II. Roles del proceso de revisión de tesis

Role	Usuario	Descripción
estudiantes		Grupo de usuarios inscritos en la carrera de ciencias y sistemas que han aprobado el curso propedéutico.
comisión		Grupo de usuarios que pertenecen a la Comisión de Revisión de Privados y Tesis.
asesores		Grupo de usuarios profesionales graduados que cumplen con todos los requisitos para poder ser asesores de tesis.
revisores		Grupo de usuarios profesionales graduados que pertenecen al grupo de revisores de tesis de la comisión de Revisión de Privados y Tesis.
estudiantes	<estudiante>	Usuario específico que pertenece al grupo de estudiantes.
comisión	<comisionado>	Usuario específico que pertenece al grupo de comisión.
asesor	<asesor>	Usuario específico que pertenece al grupo de asesores de tesis.
Coordinador	<coordinador>	Usuario específico quien tendrá a cargo el monitoreo de todo el proceso.
revisor	<revisor>	Usuario específico que pertenece al grupo de revisión de tesis.

Tabla III. Roles del proceso de solicitud del catedrático a los estudiantes

Role	Usuario	Descripción
catedráticos		Grupo de usuarios que imparten, como titulares, al menos un curso en la carrera de Ciencias y Sistemas en el presente semestre.
estudiantes		Grupo de usuarios inscritos en la carrera de ciencias y sistemas que tienen asignado al menos un curso en el presente semestre.
catedráticos	<catedrático>	Usuario específico que pertenece al grupo de catedráticos que está vinculado a un curso y una sección en particular.
estudiantes	<estudiante>	Usuario específico que pertenece al grupo de estudiantes.

Al obtener éstas tablas de Roles y Usuarios, podemos crear éstos en el *workflow management system* (WFMS) elegido para implementar nuestros procesos. En el caso del *oracle workflow* podemos realizar ésta tarea utilizando los roles y usuarios de una base de datos *oracle* o bien utilizando un servidor de directorio que cumpla con el estándar LDAP versión 3 como el *oracle internet directory*.

3.1.1.3 Tablas de mapeo

Para resumir la etapa de análisis de requerimientos se diseñó un instrumento que denominamos tabla de mapeo. Este instrumento nos permite mostrar en una tabla los distintos usuarios y roles involucrados en cada proceso, así como la información que fluye en cada paso. La tabla también nos muestra el orden en que las actividades se llevan a cabo.

La información que se muestra en las tablas de mapeo será la base para que se puedan realizar los diagramas de procesos en la siguiente etapa.

Tabla IV. Mapeo del proceso de revisión de examen

Orden	Descripción de la actividad	Quién debe dar la información.	Quién necesita estar informado	Qué información es la que se transmite	Posibles acciones a tomar por parte del receptor	Quién podrá monitorear el proceso
1.	Solicitud de Revisión	Estudiante que solicita la revisión	Catedrático del curso y sección	Curso Sección Motivo	Aceptar (3) Rechazar (2)	Director de Escuela
2.	Rechazo del catedrático	Catedrático del curso y sección	Estudiante	Motivo del rechazo	Finalizar rechazo. (5)	Director de Escuela
3.	Aceptación por parte del	Catedrático del curso y	Estudiante	Nueva Nota	Finalizar aceptación (4)	Director de Escuela

	catedrático	sección				
4.	Finalizar aceptación					Director de Escuela
5.	Finalizar rechazo					Director de Escuela

Tabla V. Mapeo del proceso de revisión de tesis

Orden	Descripción de la actividad	Quién debe dar la información	Quién necesita estar informado	Qué información es la que se transmite	Posibles acciones a tomar por parte del receptor	Quién podrá monitorear el proceso
1.	Presentación de protocolo	Estudiante	Comisión	Documento de Protocolo. Datos del estudiante. Datos del asesor	Aceptar (4) Rechazar (2) Modificar (3)	Director de Escuela. Comisión.
2.	Rechazo de protocolo	Comisión	Estudiante	Motivo del rechazo	Finalizar rechazo. (18)	Director de Escuela. Comisión
3.	Modificar Protocolo	Comisión	Estudiante	Aspectos a agregar y/o eliminar del protocolo	Presentación de protocolo (1)	Director de Escuela
4.	Aceptación de protocolo	Comisión	Estudiante	Carta de aceptación	Enviar tesis al asesor (5)	Director de Escuela. Comisión
5.	Enviar tesis al asesor	Estudiante	Asesor	Documento de tesis	Aceptar (8) Correcciones (6)	Director de Escuela. Comisión
6.	Correcciones del asesor	Asesor	Estudiante	Documento de correcciones	Reenviar(7)	Director de Escuela. Comisión

7.	Reenviar tesis al asesor	Estudiante	Asesor	Documento de tesis	Aceptar (8) Correcciones (6)	Director de Escuela. Comisión
8.	Aceptar Tesis Por parte del asesor	Asesor	Estudiante	Carta del asesor	Enviar a comisión(9)	Director de Escuela. Comisión
9.	Enviar a comisión	Estudiante	Comisión	Documento de Tesis Carta del Asesor	Asignar revisor(10)	Director de Escuela. Comisión
10.	Asignar Revisor	Comisión	Estudiante Revisor	Datos del Revisor	Enviar tesis al revisor (11)	Director de Escuela. Comisión
11.	Enviar tesis al Revisor	Estudiante	Revisor	Documento de Tesis	Correcciones (13) Rechazar (12) Aceptar (15)	Director de Escuela. Comisión
12.	Rechazar	Revisor	Estudiante Comisión	Carta de rechazo	Finalizar Rechazo (17)	Director de Escuela. Comisión
13.	Correcciones	Estudiante	Revisor	Correcciones	Reenviar(14)	Director de Escuela. Comisión
14.	Reenviar tesis al revisor	Estudiante	Revisor	Documento de tesis	Correcciones (13) Rechazar (12) Aceptar (15)	Director de Escuela. Comisión
15.	Aceptar	Revisor	Estudiante Comisión	Carta de aceptación	Finalizar aceptación (16)	Director de Escuela. Comisión
16.	Finalizar aceptación					Director de Escuela. Comisión
17.	Finalizar					Director de

	Rechazo					Escuela. Comisión
--	---------	--	--	--	--	----------------------

Tabla VI. Mapeo del proceso de solicitud del catedrático a los estudiantes

Orden	Descripción de la actividad	Quién debe dar la información.	Quién necesita estar informado	Qué información es la que se transmite	Posibles acciones a tomar por parte del receptor	Quién podrá monitorear el proceso
1.	Solicitud del catedrático	Catedrático	Estudiantes	Texto de la solicitud	Votar si (2) Votar no (3) Abstenerse (4)	Director de Escuela.
2.	Votar si	Estudiantes	Catedrático	SI	Procesar votos (5)	Director de Escuela.
3.	Votar no	Estudiantes	Catedrático	NO	Procesar votos (5)	Director de Escuela.
4.	Abstenerse	Estudiantes	Catedrático	Nulo	Procesar votos (6)	Director de Escuela.
5.	Procesar votos	Catedrático	Estudiantes	Total de votos si Total de votos no Total de abstenciones	Mostrar Resultado(6)	Director de Escuela.
6.	Mostrar Resultado	Catedrático	Estudiantes	Resultado de la votación.	Fin de proceso	Director de Escuela.

3.1.2 Diseño de los procesos.

Luego de realizar el análisis de los requerimientos de los procesos y obtener la tabla de mapeo de cada uno de ellos, debemos diseñar los procesos, cumpliendo con el estándar definido por la *workflow management coalition* (WfMC).

En nuestro caso utilizaremos el diagramador de procesos del *oracle workflow builder* para diseñar cada uno de los procesos.

3.1.2.1 Diseño del proceso: Revisión de Examen

Para poder diseñar adecuadamente nuestro flujo de trabajo, utilizaremos como base las tablas de mapeos que construimos en la etapa de análisis de requerimientos.

Luego definimos los atributos que vamos a necesitar. Estos atributos consisten en la información que fluye en cada uno de los pasos de nuestro flujo y por lo tanto pueden ser consideradas como variables dentro de nuestros procesos. Estos atributos pueden ser globales a todo nuestro proceso o bien pueden ser vistos únicamente en algunos pasos de nuestros flujos. Los atributos globales se definen en el nodo de atributos Así que definiremos nuestros atributos a continuación.

a. Definición de atributos del proceso de revisión de examen

A lo largo del proceso se necesita comunicar información de la solicitud del estudiante hacia el catedrático. Toda la información que se necesite intercambiar debe ser definida como un atributo. En el proceso de Revisión de Examen se definieron los siguientes atributos:

Tabla VII. Atributos del proceso de revisión de examen

Nombre del atributo	Tipo de dato
Nombre del curso	Texto
Código del curso	Numérico
Nombre del catedrático	Texto
Código del catedrático	Texto
Carné del estudiante	Texto
Nombre del Estudiante	Texto
Motivo de la revisión	Texto

Explicación del catedrático	Texto
Días	Numérico
Nueva nota	Numérico

b. Definición de los Mensajes del proceso de revisión de examen

Los mensajes son los contenedores de los textos que se envían dentro de las notificaciones y contienen la información para el destinatario de la actividad. El mensaje puede incitar una respuesta o bien únicamente puede proveer información.

Los mensajes están formados tanto por información estática como dinámica. En éste último caso ésta información se obtiene de los atributos definidos previamente y referenciados dentro del cuerpo del mensaje anteponiéndole el símbolo ampersand (&) lo que indica que al generarse el mensaje por el motor de *oracle workflow*, éste sustituirá la referencia de la variable, por su valor en tiempo de ejecución. Otro aspecto importante es definir la prioridad del mensaje y el formato que puede ser texto plano o html. Para mejorar la apariencia de nuestros mensajes, éstos fueron definidos en HTML.

En el caso del proceso de revisión de examen se han definido cuatro mensajes que se encuentran en el mismo número de notificaciones. El contenido de éstos mensajes se muestra en la siguiente tabla.

Tabla VIII. Mensajes del proceso de revisión de examen

Mensaje	Descripción	cuerpo del mensaje
Mensaje de revisión de examen	Este es el primer mensajes que envía el estudiante solicitando al catedrático	<TABLE> <TR> <TD></TD></TR>

	del curso que se revise su nota. El estudiante debe justificar apropiadamente la solicitud, utilizando el atributo de Motivo de la revisión.	<pre> <TR><TD>Se le solicita realizar una revisión del examen del curso &CURSO_NOMBRE (&CURSO_CODIGO) del alumno &ESTUDIANTE_NOMBRE con número de carné &CARNE las razones por las que se solicita ésta revisión son las siguientes:</TD></TR> <TR><TD>&MOTIVO_REVISIÓN </TD></TR></TABLE> </pre>
Notificación al director de escuela	Si el mensaje del estudiante tiene más de X días y aún no ha sido atendido por el catedrático del curso. Se envía éste mensaje al Director de Escuela para mantenerlo al tanto. Este mensaje se repite cada X días.	<pre> <TABLE> <TR> <TD></TD></TR> <TR><TD>El catedratico &CATEDRATICO_NOMBRE ha recibido una solicitud de revisión de examen para el curso &CURSO_NOMBRE (&CURSO_CODIGO) por parte del estudiante &ESTUDIANTE_NOMBRE (&CARNE) desde hace ya &días y no la ha aprobado o rechazado. </TD></TR><TR><TD> Por favor darle seguimiento</TD></TR> </TABLE> </pre>
Rechazo de cambio de nota	Si el catedrático no considera que se justifique el cambiar la nota al estudiante, se le envía el siguiente mensaje indicándole las razones por las que se rechaza la solicitud.	<pre> <TABLE> <TR> <TD> </TD></TR><TR><TD> Se le informa que su solicitud de revisión de la nota del curso &CURSO_NOMBRE (&CURSO_CODIGO) ha sido rechazada por el catedratico &CATEDRATICO_NOMBRE debido a las siguientes razones: </TD></TR> <TR><TD>&COMENTARIO_CATEDRATICO</ TD></TR></TABLE> </pre>
Cambio de nota	En el caso de que la solicitud del estudiante sea aprobada se envía el siguiente mensaje indicando la nueva nota del estudiante.	<pre> <TABLE> <TR> <TD> </TD></TR><TR><TD> Se le informa que su solicitud de revisión de la nota del curso &CURSO_NOMBRE (&CURSO_CODIGO) ha sido aprobada por el catedratico &CATEDRATICO_NOMBRE de acuerdo a los siguientes criterios: </TD></TR> <TR><TD>&COMENTARIO_CATEDRATICO</ </pre>

		<pre> TD></TR> <TR><TD>Su nueva nota es por lo tanto de &NUEVA_NOTA puntos.</TD></TR> </TABLE> </pre>
--	--	--

c. Definición de las notificaciones del proceso de revisión de examen.

Estas actividades se utilizan para enviar mensajes a un usuario o a un role. Para crear una notificación se debe crear primero un mensaje y luego asociarlo a la definición de la notificación. Dentro de la notificaciones se especifica qué usuario o role será el que reciba la notificación, el tiempo de espera de la respuesta, el resultado y si está asociada a una función entonces el nombre del procedimiento que se ejecutará cuando se reciba la notificación.

Tabla IX. Notificaciones del proceso de revisión de examen

Notificación	Destinatario	Tiempo de espera	Tipo de resultado del mensaje	Mensaje
Solicitud de revisión de examen	Catedrático	10 días	Aprobación	Mensaje de revisión de examen
Notificación al director que no se ha atendido solicitud	Director de Escuela	Nulo	Nulo	Notificación al director de escuela
Se rechaza el cambio de la nota	Estudiante	Nulo	Nulo	Rechazo de cambio de nota
Cambio de nota	Estudiante	Nulo	Nulo	Cambio de nota

d. Definición del proceso de revisión de examen.

El último paso consiste en diagramar las notificaciones y sus transiciones. Cada notificación lleva asociado un mensaje por lo que no es necesario incluir éstos dentro de nuestro diagrama.

Por otro lado los atributos que utilizamos están asociados a los mensajes, por lo que no es necesario incluirlos tampoco en el diagrama.

El diagrama de procesos se crea arrastrando los componentes del navegador hacia el área de diagramación con lo que se crea un nodo de la notificación, ya que una notificación puede estar en múltiples diagramas.

Luego con el botón derecho se crean las transiciones. A cada notificación dentro del diagrama se le debe asignar un destinatario y un tipo de respuesta dentro de sus propiedades.

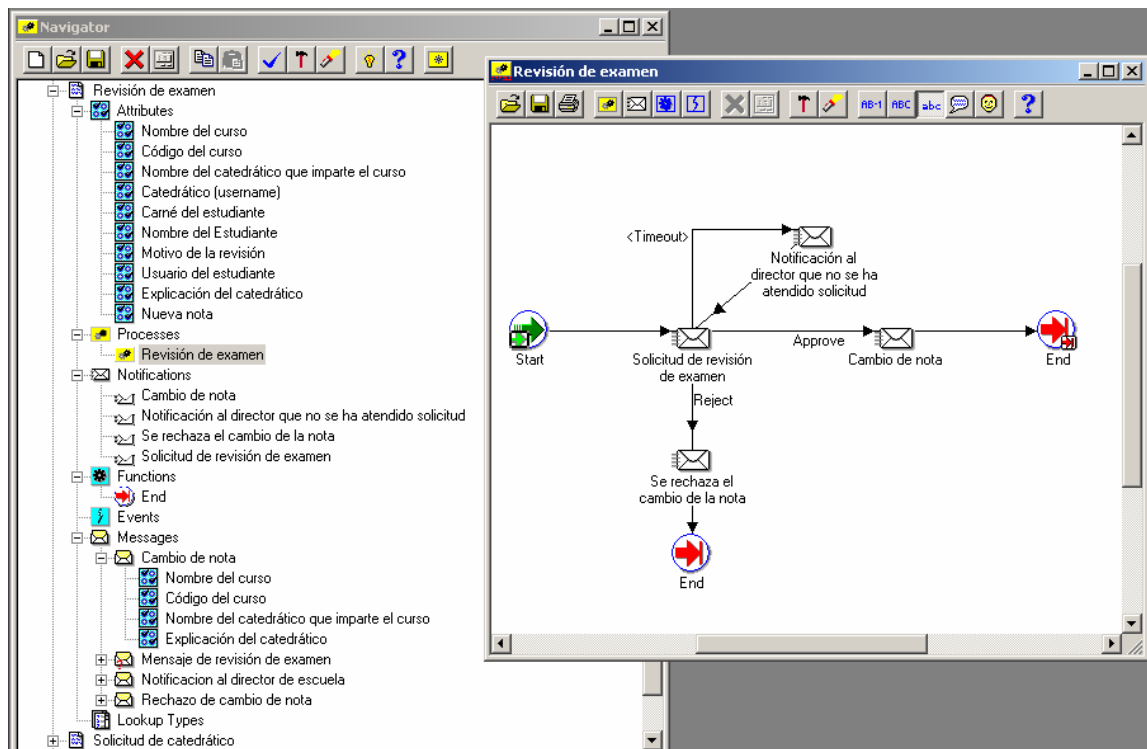
En éste diagrama pueden apreciarse las notificaciones que se han definido para el proceso, así como las diferentes transiciones que tiene cada una de ellas. El diagrama tiene la ventaja de presentar el proceso mucho más entendible a cualquier persona, que las tablas de mapeo de los pasos anteriores.

En el diagrama de la revisión del catedrático se pueden observar que inicia el proceso en la actividad start, luego se envía la notificación de Solicitud de revisión de examen al catedrático del curso. Esta notificación tiene asociado una transición de tipo *timeout*, es decir si en el período de tiempo estipulado no se ha atendido la solicitud, se envía una notificación al director de escuela. Esta notificación se repetirá tantas veces como se cumpla el *timeout*.

Si el catedrático atiende la solicitud tiene dos opciones: La primera es rechazarla, si elige ésta opción se enviará una notificación al estudiante que solicitó la revisión notificándole que su solicitud ha sido rechazada y el motivo. Esta notificación es únicamente informativa, por lo que el estudiante al aceptarla finaliza el proceso.

La segunda opción del catedrático es aceptar la solicitud, con lo que se envía al estudiante que hizo la solicitud una notificación indicándole la nueva nota que le corresponde. Al recibir la notificación informativa el estudiante únicamente tendrá la opción de aceptarla, con lo que se finaliza el proceso.

Figura 36. Diagrama del proceso de revisión de examen



3.1.2.2 Diseño del proceso: Revisión de Tesis

a. Definición de atributos del proceso de revisión tesis.

A lo largo del proceso se necesita comunicar información sobre el proceso de revisión de tesis entre el estudiante, el asesor, el revisor, la comisión de tesis y el coordinador. Toda la información que se necesite intercambiar debe ser definida como un atributo. En el proceso de revisión de tesis se definieron los siguientes atributos:

Tabla X. Atributos del proceso de revisión de tesis

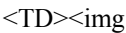
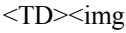
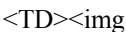
Nombre del atributo	Tipo de dato
Titulo de la tesis	Texto
Carné del estudiante	Texto
Nombre del estudiante	Texto
Nombre del asesor	Texto
Documento de tesis	Blob
Motivo de rechazo del protocolo	Texto
Motivo de rechazo por parte del asesor	Texto
Motivo de rechazo por parte del revisor	Texto
Usuario del estudiante	Texto
Usuario del asesor	Texto
Usuario del revisor	Texto

b. Definición de los mensajes del proceso de revisión de tesis.

En el caso del proceso de revisión de tesis se han definido nueve mensajes con los que se intercambia información en el proceso de revisión de tesis. El contenido de estos mensajes se muestra en la siguiente tabla.

Tabla XI. Mensajes del proceso de revisión de tesis

Mensaje	Descripción	cuerpo del mensaje
Solicitud de revisión de protocolo	Mensajes que envía el estudiante para proponer su protocolo a revisión.	<pre> <TABLE> <TR> <TD></TD></TR> <TR><TD>El estudiante &NOMBRE_ESTUDIANTE, con carné: &CARNE le solicita revisar el protocolo de su tesis: <I>&TITULO_TESIS </I>cuyo asesor es &NOMBRE_ASESOR y cuya carta de aceptación también se incluye.</TD></TR> <TR><TD>Si el contenido del protocolo cumple con todos los requisitos para ser aceptado, por favor aprobar ésta solicitud, de lo contrario indicar las razones del rechazo. </TD></TR> <TR><TD> En el caso de aprobar el protocolo debe indicar el usuario del asesor, que ha aceptado revisar el trabajo de graduación, en el campo <I>Usuario del asesor de Tesis </I>. </TD></TR> </TABLE> </pre>
Rechazo del protocolo de tesis	Se rechaza el protocolo de tesis y se indica el motivo.	<pre> <TABLE> <TR> <TD></TD></TR> <TR><TD>Lamentamos informarle que el protocolo de su </pre>

		<p>tesis: <I>&TITULO_TESIS </I>cuyo asesor es &NOMBRE_ASESOR ha sido rechazada debido a:</p> <p>&MOTIVO_RECHAZO_PASO1</p>
Se acepta el protocolo	Se acepta el protocolo y se asigna un asesor	<p></p> <p>Se le notifica que el protocolo de su tesis titulada: <i>&TITULO_TESIS</i> cuyo asesor es &NOMBRE_ASESOR ha sido aprobado</p> <p>Por favor mande el documento de progreso de su tesis para su revisión.</p>
Se envía tesis al asesor para su revisión	El estudiante envía la tesis al asesor para su revisión.	<p></p> <p>Estimado ing. &NOMBRE_ASESOR el estudiante &NOMBRE_ESTUDIANTE, con carne &CARNE le solicita revisar su tesis, si tiene alguna corrección, por favor presione el boton de <I>reject</I> e indique los cambios que se sugieren.</p> <p>Si el trabajo cumple con todos los requisitos técnicos y de forma, por favor acepte el documento de la tesis, presionando en el boton de "accept".</p>
Correcciones	El asesor envía al estudiante las observaciones correspondientes.	<p></p>

		<p><TR><TD>El asesor de su trabajo de tesis: <I> &TITULO_TESIS </I>, el ing. &NOMBRE_ASESOR considera que debe corregir su tesis en los siguientes aspectos:</p> <p></TD></p> <p></TR></p> <p><TR> <TD></p> <p>&MOTIVO_RECHAZO_PASO2 .</p> <p></TD></TR></p> <p><TR><TD></p> <p>Por favor realizar las correcciones sugeridas y cuando ya tenga las correcciones hechas, enviar de nuevo el documento.</p> <p></TD></TR></p> <p></TABLE></p>
Revisión de tesis finalizada	El asesor está de acuerdo con todo el contenido de la tesis	<p><TABLE></p> <p><TR><TD></TD></TR></p> <p><TR><TD></p> <p>El ing. &NOMBRE_ASESOR ha aprobado completamente el trabajo de tesis:</p> <p><I> &TITULO_TESIS </I> del alumno &NOMBRE_ESTUDIANTE (&CARNE) por lo que se solicita</p> <p>que se asigne a un profesional para que realice la revisión final.</TD></TR></p> <p><TR><TD>Cuando se haya elegido a éste profesional, por favor ingrese el usuario en el siguiente campo para que se le notifique.</TD></TR></p> <p></TABLE></p>
Se envía tesis al revisor para supervisión final	Se envía una copia de la tesis al revisor para que de el visto bueno.	<p><TABLE></p> <p><TR></p> <p><TD></TD></TR></p> <p><TR><TD></p>

		<p>Estimado ingeniero. La comisión de revisión de Tesis lo ha nombrado para que realice la revisión final del trabajo de tesis: <I> &TITULO_TESIS </I> del alumno &NOMBRE_ESTUDIANTE (&CARNE). Esta tesis ya ha sido aprobada por el asesor del estudiante. Si a su criterio el trabajo cumple con todos los requisitos, por favor presiones "accept", de lo contrario indique los problemas que encuentra para que el alumno los corrija. </TD></TR></p>
Correcciones finales	Las correcciones que crea convenientes el revisor	<p><TABLE></p> <p><TR><TD></TD></TR></p> <p><TR> <TD> La comisión de revisión de tesis ha nombrado a un revisor para evaluar su trabajo de graduación. Este revisor final ha considerado que su trabajo de tesis: <I> &TITULO_TESIS </I>, se debe corregir en los siguientes aspectos:</p> <p></TD></TR></p> <p><TR><TD> &MOTIVO_RECHAZO3 </TD> </TR></p> <p><TR><TD></p> <p>Por favor realice las correcciones sugeridas y cuando ya tenga las correcciones hechas, notifique al revisor presionando el botón de "yes". </TD></TR></p> <p></TABLE></p>
Aprobación final del revisor	El revisor está de acuerdo con todo el contenido de la tesis	<p><TABLE></p> <p><TR></p> <p><TD></TD></p> <p></TR></p> <p><TR></p> <p><TD>Se le notifica que el revisor final de su tesis &TITULO_TESIS la ha aprobado de forma definitiva.</TD></p>

		</TR> <TR> <TD>Se le envía una carta del revisor, con la que debe presentarse en la Dirección de Escuela. </TD> </TR> <TR><TD>FELICITACIONES</TD></TR> </TABLE>
--	--	--

c. *Definición de las notificaciones del proceso de revisión de tesis.*

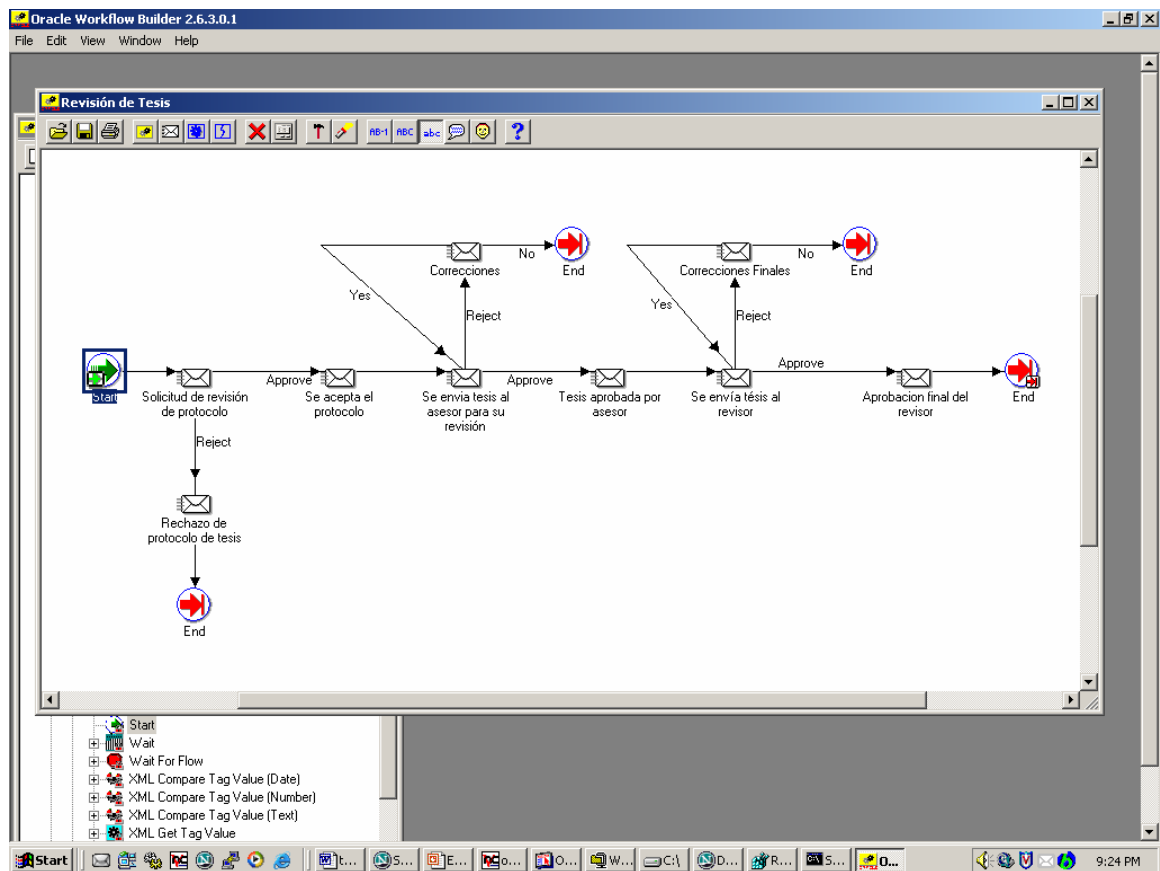
Tabla XII. Notificaciones del proceso de revisión de tesis

Notificación	Destinatario	Tiempo de espera	Tipo de resultado del mensaje	Mensaje
Solicitud de revisión de protocolo	Comisión de tesis	10 días	Aprobación	Solicitud de revisión de protocolo
Rechazo del protocolo de tesis	Estudiante	nulo	Informativo	Rechazo del protocolo de tesis
Se acepta el protocolo	Estudiante	Nulo	Informativo	Se acepta el protocolo
Se envía tesis al asesor para su revisión	Asesor	15 días	Aprobación	Se envía tesis al asesor para su revisión
Correcciones	Estudiante	Nulo	Informativo	Correcciones
Tesis aprobada por asesor	Estudiante	Nulo	Informativo	Revisión de tesis finalizada
Se envía tesis al revisor para supervisión final	Revisor	15 días	Aprobación	Se envía tesis al revisor para supervisión final
Correcciones finales	Estudiante	Nulo	Informativo	Correcciones finales
Aprobación final del revisor	Estudiante	Nulo	Informativo	Aprobación final del revisor

Es importante distinguir entre los dos tipos de notificaciones que se han utilizado para éste proceso. El primero llamado aprobación ya lo hemos utilizado y significa que el destinatario que reciba la notificación podrá optar por una de dos alternativas: aprobar la notificación o rechazarla. Y el segundo llamado informativo, en el que el destinatario tendrá un botón de aceptar que no tendrá ningún efecto en la siguiente transición.

d. Definición del proceso de revisión de tesis.

Figura 37. Diagrama del proceso de revisión de tesis



3.1.2.3 Diseño del proceso de solicitud de catedrático

a. Definición de atributos del proceso de solicitud de catedrático.

Para la solicitud por parte del catedrático se han definido los atributos: Motivo de la votación, catedrático solicitante, porcentaje de votos no, porcentaje de votos si y tipo de votación.

El resto de atributos corresponde únicamente a la votación misma, por lo que su valor no debe ser utilizado fuera de ésta actividad. Es por esto que se definen dentro los mensajes y no globalmente.

El tipo de votación se define como una lista de valores ya que sus opciones están limitadas. Esta lista de valores es necesario crearla anteriormente en el navegador.

Tabla XIII. Atributos del proceso de revisión de solicitud del catedrático a los estudiantes

Nombre del atributo	Tipo de Dato
Motivo de la votación	Texto
Catedrático Solicitante	Texto
Porcentaje de Votos No	Numérico
Tipo de Votación	Lista de valores (Lookup)
Porcentaje de Votos Sí	Numérico

b. Definición de los mensajes del proceso de solicitud de catedrático

En el caso del proceso de solicitud del catedrático a los estudiantes se han definido tres mensajes con los que se intercambia información en el proceso. El contenido de estos mensajes se muestra en la siguiente tabla.

Tabla XIV. Mensajes del proceso de solicitud del catedrático a los estudiantes

Mensaje	Descripción	cuerpo del mensaje
Votación para elegir fecha de examen	Este es el primer mensaje que envía el catedrático sometiendo a votación su propuesta de cambio de fecha de examen.	<pre> <TABLE> <TR> <TD> </TD></TR> <TR><TD>Estimados estudiantes. Estamos sometiendo a votación la siguiente pregunta: </TD></TR><TR><TD> &MOTIVO</TD></TR> <TR><TD> Si está usted de acuerdo. Por favor presione <i>SI</i>, de lo contrario presione <i>NO</i>. </TD></TR><TR><TD> Gracias por su participación. </TD></TR> </TABLE> </pre>
Propuesta Rechazada	Mensaje que se envía al catedrático cuando se determina que la mayoría de estudiantes votaron en contra de la propuesta.	<pre> <TABLE> <TR> <TD></TD></TR> <TR><TD> Estimado Ingeniero: Su invitación a votación sobre la pregunta: </TD></TR>&MOTIVO <TR><TD>Ha sido RECHAZADA por la mayoría de los estudiantes.</TABLE> </pre>
Solicitud	Mensaje que se envía al	<pre> <TABLE> <TR> </pre>

Aceptada	catedrático cuando se determina que la mayoría de estudiantes votaron a favor de la propuesta.	<pre> <TD></TD></TR> <TR><TD> Estimado Ingeniero: Su invitación a votación sobre la pregunta: </TD></TR>&MOTIVO<TR><TD> Ha tenido un resultado en la votación, favorable a la ACEPTACIÓN.</TABLE> </pre>
----------	--	--

c. Definición de las notificaciones del proceso de solicitud de catedrático.

En éste proceso resalta la notificación de votación, ya que se encuentra asociada a una función. Esto significa que cuando se envíe y responda la notificación se ejecutará código en PL/SQL. Esto se realiza ya que se hará un procesamiento más complejo, en ésta caso se necesita contar el número de votos que obtenga cada opción de la votación y luego dependiendo del resultado se definirá la transición que corresponda.

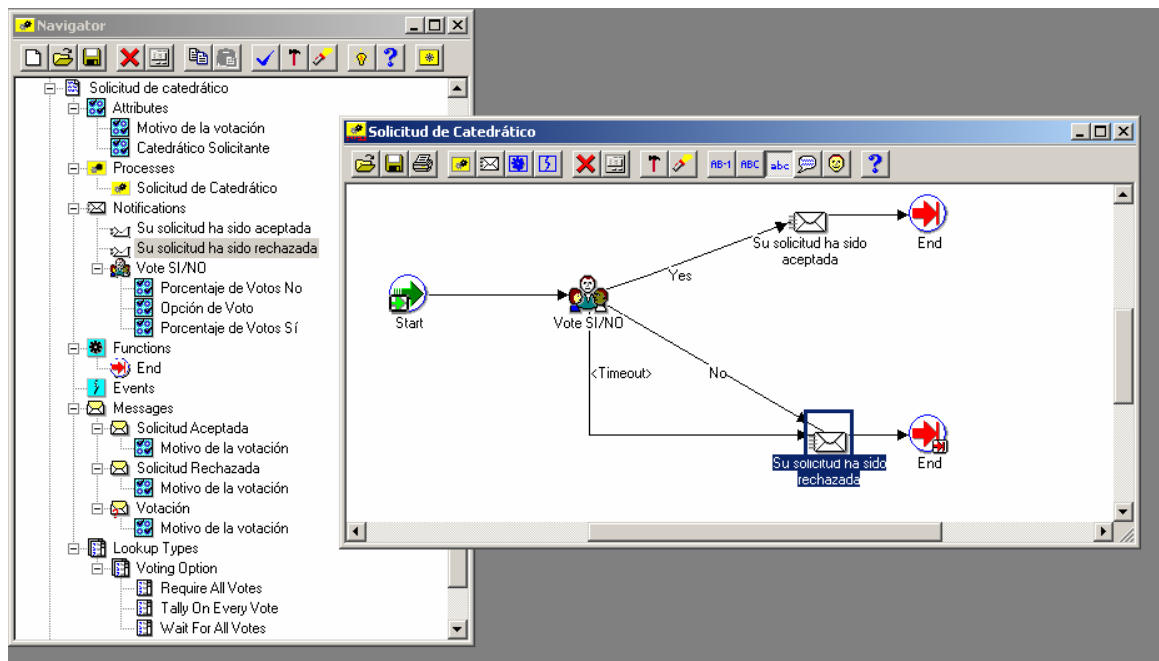
Tabla XV. Notificaciones del proceso de solicitud del catedrático a los estudiantes

Notificación	Destinatario	Tiempo de espera	Tipo de resultado del mensaje	Mensaje
Vote SI/NO	Estudiantes	10 días	Lista de valores (SI/NO)	Votación
Su solicitud ha sido aceptada	Catedrático	Nulo	Nulo	Solicitud aceptada
Su solicitud ha sido rechazada	Catedrático	Nulo	Nulo	Solicitud rechazada

d. *Definición de las notificaciones del proceso de solicitud de catedrático.*

El diagrama del proceso de solicitud de catedrático puede verse en la siguiente gráfica.

Figura 3.4 Diagrama del proceso de Solicitud de Catedrático



3.2 Resultados esperados

3.2.1 Resultados de la automatización del proceso de revisión de examen.

Actualmente la solicitud de revisión de exámenes es una tarea compleja y con muchos retardos ya que es necesario que el estudiante hable con el profesor exponga su caso y le dé seguimiento. Muchos estudiantes prefieren no realizar el proceso debido a que trabajan y no pueden darle el seguimiento adecuado.

A través del uso del sistema de administración de flujos de trabajo, los estudiantes podrán fácilmente solicitar a los catedráticos la revisión de sus exámenes de una manera rápida y eficiente y podrán estar enterados de las actividades que se han realizado.

Además las autoridades podrán verificar rápidamente el estado de cada una de las solicitudes. Por otro lado se podrán obtener estadísticas de atención para retroalimentar a los catedráticos.

3.2.2 Resultados de la automatización del proceso de revisión de tesis.

Actualmente muchos de los pasos que involucran la revisión de una tesis son manuales. Así el estudiante imprime su trabajo y se pone de acuerdo con su asesor para entregárselo, luego el asesor lo revisa y debe coordinarse con el estudiante para entregarle las correcciones. Aún en el caso de utilizar correo electrónico no se cuenta con una manera de determinar el tiempo que le lleva tanto al catedrático como al estudiante cada una de las correcciones.

El sistema proporciona un entorno muy conveniente donde todos los involucrados pueden recibir notificaciones cuando sea necesaria su participación. Así sustituimos la necesidad de buscar la información por la notificación automática.

Las autoridades también se ven beneficiadas ya que pueden determinar los tiempos de atención para cada uno de los casos y pueden fácilmente monitorear el progreso de un proceso cualquiera.

3.2.3 Resultados de la automatización del proceso de revisión de tesis.

Los catedráticos que necesiten someter a consideración de todos los alumnos de su curso un cambio para que sea aprobado se verán beneficiados con el nuevo sistema de votación a través de la automatización del proceso de solicitud de catedrático a los estudiantes.

Cambios tan comunes como un nuevo horario de clases o de laboratorio, el cambio en la hora o fecha de un examen, etc. Son algunos ejemplos de procesos comunes que ahora pueden ejecutarse desde el workflow de la facultad de Ingeniería.

CONCLUSIONES

1. La automatización de procesos a través de la implementación de flujos de trabajo nos permite reducir los costos, mejorar la calidad de los procesos, estandarizar los procesos, monitorear el progreso de cada actividad dentro del proceso y mejorar la asignación de recursos y tareas.
2. Modelar y diseñar procesos con herramientas de software facilitan la tarea de realizar cambios y mejoras en los procesos, así como realizar las pruebas de impacto que estos cambios tendrán dentro de la organización.
3. Utilizar herramientas de software que cumplen los estándares de diseño y modelación de procesos nos permiten, posteriormente, integrar nuestros procesos con los de otras organizaciones.
4. En el caso de los procesos administrativos de la Facultad de Ingeniería que se han modelado se estima que se logrará una mejora significativa en los tiempos de atención y calidad de servicio.

RECOMENDACIONES

1. Definir una política de revisión continua de procesos administrativos dentro de la Facultad de Ingeniería, dentro de la cuál se recomiende la utilización de una herramienta de modelación, diseño y ejecución de flujos de trabajo.
2. Implementar nuevos procesos administrativos utilizando la herramienta elegida y someterlos a revisiones periódicas para mejorarlos continuamente.
3. Realizar un estudio de capacidad de atención para determinar si se cuenta con la infraestructura de hardware y red adecuado para atender a todos los estudiantes de la Facultad pensando en el crecimiento de la demanda que tendrá el sistema.

BIBLIOGRAFÍA

1. Khoshafian, Setrag y Buckiewicz, Marek. *Introduction to Groupware, Workflow and WorkGroup Computing*. Editorial Wiley, 1995. ISBN: 0-471-02946-7.
2. Coleman, David. *Groupware Collaborative Strategies for Corporate and Intranets*. Editorial Prentice Hall, 1997. ISBN: 0-13-727728-8.
3. Manganelli, Raymond y Klein, Mark. *Como Hacer Reingeniería*. Grupo Editorial Norma. ISBN: 958-0430-25-x.
4. Steffen, Hermann. *Workflow. Una Herramienta para la Gestión de Procesos de Empresa*.
5. Oracle Corporation. *Oracle Workflow Release 2.6 Introducción*. Doc. Reference 14467GC11

REFERENCIAS ELECTRÓNICAS

1. *Workflow Management Coalition. The Workflow Reference Model. TC00-1003*
<http://www.wfmc.org/standards/docs/>
2. *Workflow Management Coalition. Workflow Management Coalition Terminology & Glossary. Reference Model. TC-1011.*
<http://www.wfmc.org/standards/docs/>