

# CAPÍTULO 3

## Integración de Aplicaciones

*Patricia Bazán*

La integración de aplicaciones es un aspecto largamente abordado en la literatura y fueron varios los autores que acuñaron diversas definiciones.

Al referir el término “integración de aplicaciones” surge naturalmente la noción de EAI o *Enterprise Application Integration* como Uso de software y sistemas informáticos para integrar un conjunto de aplicaciones en una organización (Wikipedia)

*Enterprise Application Integration* emergió como una iniciativa de la industria de la computación con el advenimiento del año 2000 y la caducidad de muchos viejos sistemas no compatibles Y2K.

El objetivo de la EAI no es nuevo, comenzó ya con la distribución de datos desde los *mainframes* hacia los procesadores *front-end*, sin embargo, los datos atrapados entre distintas compañías y formatos tornan difícil la tarea [Juric M. et al., 2007].

EAI se enfoca en el desarrollo de sistemas que proveen funcionalidad de negocios fluida donde la integración está en la capacidad de intercambiar y compartir datos sin saber dónde están ubicados y qué formato tienen.

Las siguientes dos definiciones obtenidas de [Morgenthal JP., 2001] son las que mejor se adaptan a la idea de integración de aplicaciones que se quiere transmitir:

La integración fluida de procesos de negocio con el propósito de conducirlos electrónicamente

La integración es compartir y/o intercambiar datos entre los sistemas con el propósito de proveer una interfaz unificada

Este capítulo presenta la evolución de la integración de aplicaciones en términos tanto tecnológicos como metodológicos y tiene por objetivo mostrar que dicha evolución está guiada por los conceptos de reusabilidad, distribución y orquestación de servicios.

### 3.1. Evolución tecnológica de las aplicaciones

Los sistemas de información tradicionales que sustentan su modelado en un enfoque netamente dirigido a los datos han dominado la industria por largo tiempo. En los últimos tiempos

subyace, cada vez con más ímpetu, la idea de que los procesos son igualmente importantes y deben ser considerados de una manera sistemática.

Por otra parte, el paradigma de desarrollo orientado a servicios está llevando el mundo de la tecnología de información a un cambio de enfoque en la manera de resolver problemas y también a una nueva visión de integración de aplicaciones.

En la década del 70 el paradigma era el cálculo, las aplicaciones eran de estadísticas/contabilidad y balística y la herramienta los lenguajes de programación procedurales sin entornos de desarrollo.

En la década del 90 el paradigma era la construcción de los sistemas de información, aplicaciones de gestión generalmente de índole administrativa y las herramientas eran los RDBMS, TP, *Workflow* y *DataWarehouse*.

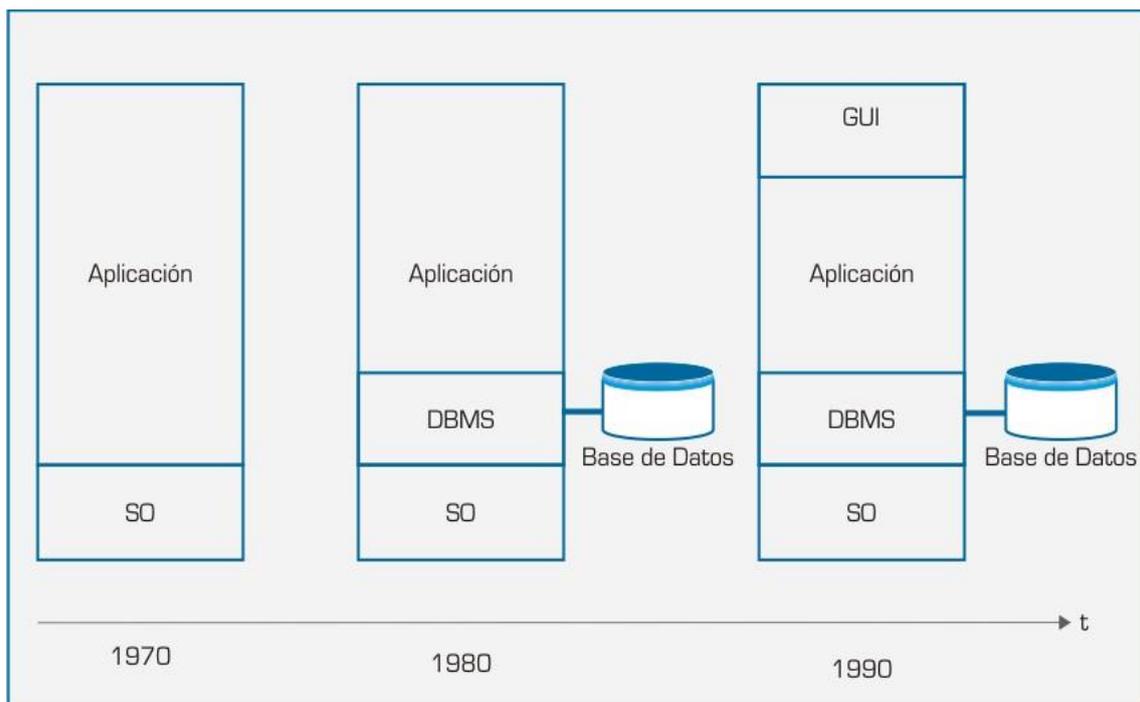
La tendencia actual es hacia un paradigma orientado a procesos cuyo enfoque se ha sustentado en los conceptos de *workflow* pero que encontró en BMP un soporte de la idea de procesos mucho más amplio incluyendo simulación, verificación y monitoreo.

Además, el enfoque orientado a procesos actual ordena la arquitectura de integración accidental originada por la integración punto a punto, proveyendo un mayor nivel de abstracción donde existen procesos que orquestan servicios, que a su vez, resuelven la funcionalidad.

La tecnología de Servicios Web representa el estándar de computación distribuida más adoptado de la historia de la industria y son la plataforma ideal para la Arquitectura Orientada a Servicios (SOA).

Asimismo, la orquestación y coreografía de los Servicios Web como *workflow* son partes esenciales de las definiciones de trabajo actuales.

La orquestación impone el orden y el ritmo individual a un conjunto de Servicios Web. La coreografía define el comportamiento variado entre los Servicios Web.



**Figura 20.** Evolución de tecnológica de aplicaciones

Como se muestra en la Figura 19 extraída de [Weske M., 2008], la evolución marca la tendencia de otorgarle primero una gran independencia al componente de manipulación de datos con la aparición de los DBMS. Luego de ello se dio mayor relevancia a la interfaz con el usuario, dejando la lógica de la aplicación como un elemento o componente separado y que debería tender a no mezclarse con los otros dos elementos.

### 3.2. Alternativas de integración de aplicaciones

La integración de aplicaciones presenta una evolución descrita parcialmente en [Bazán P, 2017] cuando se menciona la evolución desde los objetos distribuidos hacia los servicios. Allí se menciona la integración punto a punto como una de las primeras aproximaciones donde (existe) la misma se encontraba embebida en el código con serias dificultades para su modificación y gestión y haciendo imposible su administración y monitoreo.

Esto provocó enormes frustraciones tanto en las áreas de IT como de negocios de las organizaciones y comenzó a esbozar las definiciones de EAI enunciadas más arriba.

Surge así la EAI tradicional como primera alternativa que dará lugar a los modelos basados en *workflow* y más tarde al uso de modelos de procesos que utilizan servicios dentro de una SOA.

#### EAI tradicional

Este modelo se basa en la construcción de un nodo central y un conjunto de repetidores directamente asociados a éste, aunque los repetidores no se encuentran conectados entre sí. De esta manera, el middleware está representado por la integración centralizada de aplicaciones, y las aplicaciones que deben ser integradas son reflejadas por los repetidores. Las mismas interactúan entre ellas por medio de la integración centralizada de aplicaciones [Weske M., 2008].

Las aplicaciones pueden interactuar de formas muy diversas, desde invocaciones simples hasta interacciones complejas entre múltiples aplicaciones. Estas últimas consisten en una serie de actividades representadas por una invocación a una aplicación, además de existir restricciones de ejecución entre las mismas.

Este esquema de agentes y mensajes tiene ciertos inconvenientes. El primero de ellos es que el agente contiene cierta lógica, oculta en las reglas. La programación de éstas puede volverse una labor compleja debido a las dependencias que pueden darse entre las mismas, y el hecho de cambiar una regla puede tener implicancias en el comportamiento global del sistema.

La razón principal de estos problemas es la “pérdida” conceptual que se da en la integración de aplicaciones, ya que la integración de datos y de procesos requiere gran actividad de programación y configuración de bajo nivel, tanto de adaptadores como de los agentes de mensajes.

La integración de datos suele darse mediante actividades de mapeo, lo cual requiere un modelo de datos acordado entre todas las aplicaciones y que reside en los agentes. Este mo-

delo global suele no ser explícitamente desarrollado, pero es común encontrarlo oculto en las reglas de mapeo de datos efectuadas por los adaptadores.

### **Sistemas de Workflow y la integración de aplicaciones**

El término *workflow* consiste en la automatización de un proceso de negocio, en su totalidad o en parte, en el cual se intercambian documentos, información o tareas de un participante a otro, para provocar la acción de acuerdo a un conjunto de reglas procedimentales [Weske M., 2008].

Un *sistema de manejo de workflow* es un sistema que permite definir, crear y manejar la ejecución de flujos de trabajo a través del uso de software, que corre en uno o más motores, y que es capaz de interpretar la definición del proceso, interactuar con los participantes del *workflow* y, donde sea requerido, invocar el uso de herramientas y aplicaciones IT.

La integración de aplicaciones es efectuada por el sistema de manejo de *workflow*, usando adaptadores similares a los que se usan en un ambiente tradicional de aplicaciones empresariales.

La tecnología de *workflow* es capaz de soportar procesos de negocio dentro de un sistema dado o dentro de un conjunto de aplicaciones, lo que permite efectivamente integrar estos sistemas. Sin embargo esta tecnología posibilita también representar procesos en los que hay seres humanos activamente involucrados, y así mejorar la colaboración entre los trabajadores con conocimiento.

La tecnología de manejo de *workflow* puede ser utilizada para facilitar la modificación de la lógica del proceso realizado por aplicaciones. Las funciones de una aplicación son pasos en el *workflow*, y cada componente usa un modelo de *workflow* para representar las funciones. Por la modificación de la lógica del proceso especificada en los modelos de *workflow*, se puede modificar el comportamiento de las aplicaciones sin codificar.

Hoy en día, la mayor cantidad de aplicaciones empresariales, como las aplicaciones de planeamiento, poseen un componente *workflow* que facilita la adaptación flexible de los procesos de negocio dentro de estos sistemas. Observar que usamos el término “componente de *workflow*” además de “sistema de manejo de *workflow*”, debido a que un componente no es un sistema aislado sino que está embebido en la aplicación.

Podemos concluir que una aplicación de *workflow* única consiste de actividades y su correspondiente ordenamiento causal y temporal, donde las mismas son realizadas por un sistema común (tal como se muestra en la Figura 20 [¡Error! No se encuentra el origen de la referencia.] [Weske M., 2008]). Los *workflow* de aplicación múltiple contienen actividades que son realizadas por sistemas de múltiples aplicaciones, proveyendo así una integración de las mismas (tal como se muestra en la Figura 21 [Weske M., 2008]).

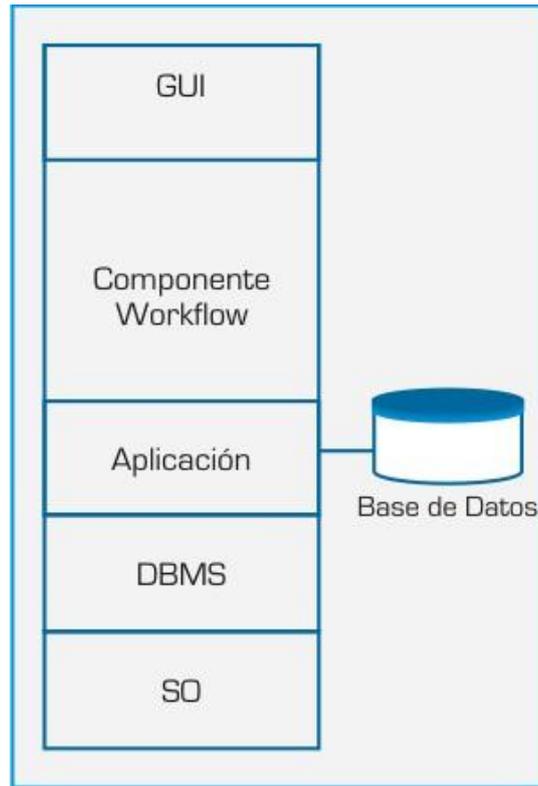


Figura 21. Componente Workflow de aplicación simple

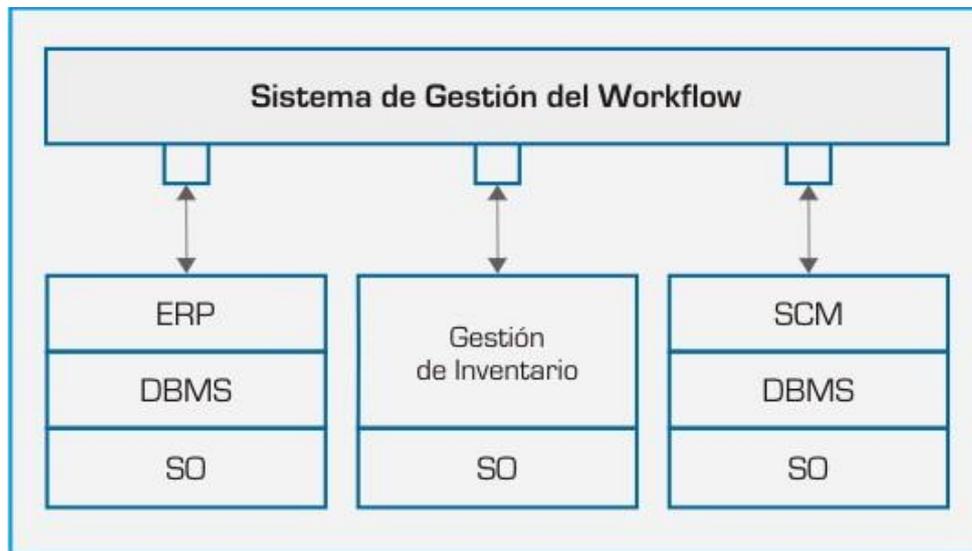


Figura 22. Componente Workflow de aplicación múltiple.

### Integración basada en SOA y BPM

El software de *workflow* utilizado para integrar aplicaciones, puede cubrir cuatro aspectos:

1. Ser un componente de aplicaciones verticales. Un software de *workflow* es ágil a la hora de adaptarse a los cambios de procesos y cambios organizacionales. Esta es una realidad en muchas aplicaciones verticales.

2. Son adecuados para utilizar como APIs. Esto es así si cuenta con soporte para Java que permite integrarse tanto para aplicaciones Web como para otras aplicaciones de IT.
3. Constituye el elemento unificador para aplicaciones colaborativas, tanto desde el punto de vista de aplicaciones que se construyen como composición de otras bajo la filosofía de *Web Services*, como del de la automatización de procesos basados en reglas.
4. Se ajustan a la implementación de *Web Services*.

Pero en una aplicación donde el proceso de negocio sea realmente un conjunto de tareas cuyos participantes son Servicios Web, provoca inevitablemente un desorden dentro del *workflow* y surge la necesidad de interoperar y describir procesos ejecutables.

La interoperabilidad se logra a través de la adopción de estándares como XML y WSDL, mientras que la descripción de procesos ejecutables se puede llevar a cabo a través de BPEL [Kettinger et al., 1997] [Yourdon, E et al. 1988].

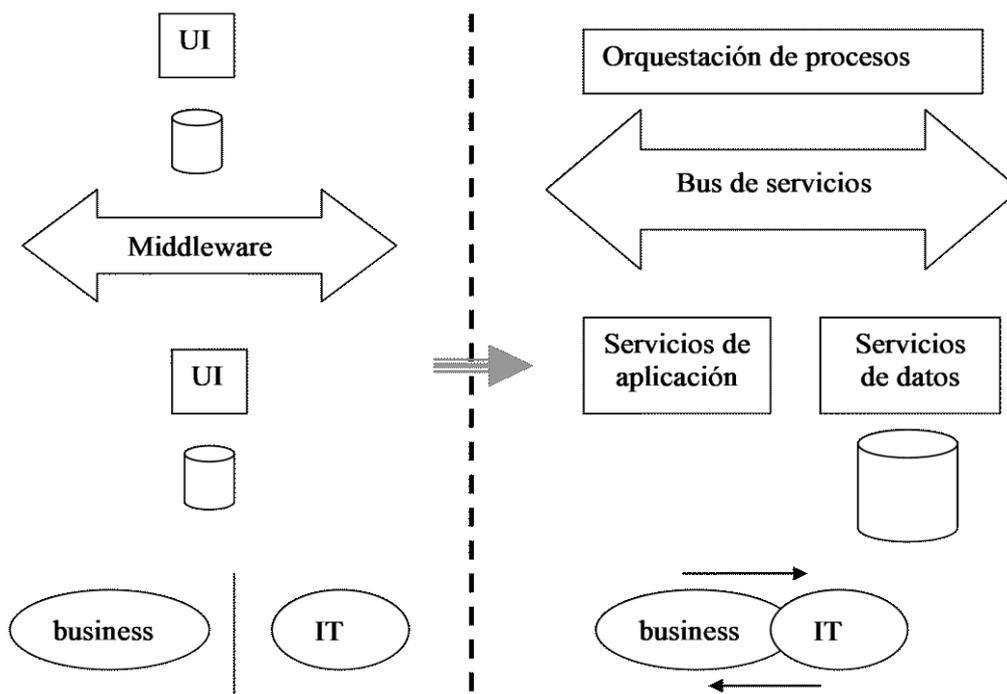
Los desarrollos en arquitectura de software empresarial y en BPM están relacionados con el manejo de *workflow*. El logro principal que se busca alcanzar aquí es la representación explícita de las estructuras de los procesos a través de modelos, y la representación controlada de los procesos basada en los modelos creados con anterioridad [Weske M., 2008].

Para la realización de aplicaciones de composición en un ambiente de orientación a servicios, se utilizan técnicas de composición de servicios.

El *middleware* de integración de aplicaciones en general, y el *middleware* de bus de servicios empresariales en particular, proveen una base técnica aceptable para realizar composición de servicios, debido a que proveen interfaces estándar que pueden ser utilizadas en desarrollos de composición. El *middleware* típico para la integración de aplicaciones empresariales presenta un componente de *workflow* de sistema, que puede o bien usar un código propietario o bien usar código BPEL (Lenguaje de ejecución de procesos de negocio para *Web Services*).

La composición de servicios es una idea de especial interés para el desarrollo de nuevas aplicaciones, basándose en funcionalidad ya existente. Así, la composición describe la forma en que se relacionan los distintos servicios, es decir que se están describiendo estructuras de proceso. Como resultado, una composición posee un conjunto de servicios, cada uno de los cuales realiza una actividad.

La Figura 22 muestra la analogía entre aplicaciones tradicionales, que conllevan un mecanismo de integración tradicional, con las aplicaciones basadas en servicios [Oracle, 2007] [Scott McKorkle, 2007].



**Figura 23.** SOA alinea negocios con IT, llevando las aplicaciones tradicionales hacia las que usan servicios

BPM como la nueva metodología para satisfacer los objetivos de una organización a través de la gestión de procesos de negocio, no plantea una visión completamente renovadora y de empezar de cero, sino que es necesario contemplar todo el trabajo realizado con anterioridad dentro de la empresa, de manera de apuntar a la integración. Aquí es donde se insertan los conceptos de SOA, *Web Services* y BPEL, de manera tal de lograr construir una aplicación real, integrada e insertada en un entorno B2B, y por sobre todo, orientada a procesos.

### 3.3. Elementos de un modelo de integración

El modelo de integración tiene por objetivo lograr una integración completa, segura y confiable de un conjunto de sistemas de software existentes, maximizando la reutilización de código, manteniendo un bajo acoplamiento y favoreciendo el mantenimiento ágil y a bajo costo.

Los elementos de este modelo de integración llevan a analizar los tipos de integración posible, los métodos aplicados para llevar a cabo la integración, los componentes de infraestructura requerida y los actores que participan. Este análisis realizado tiene por finalidad aportar criterios a la hora de decidir cuáles de todos los elementos se elegirán para componer un modelo de integración.

#### Tipos de integración

Si bien existen diversas taxonomías acerca de los posibles tipos de integración existentes [Juric M. et al., 2007], los más frecuentes, teniendo en cuenta el enfoque que se abordará, se refieren a la integración desde dos aspectos posibles:

- Integración a nivel de datos: se enfoca en el movimiento de datos entre aplicaciones con el objetivo de compartirlos. Es una integración relativamente simple si se comparan formatos y estructuras, de lo contrario se establecen protocolos o acuerdos entre las partes para poder realizar la integración. (Ejemplo: integración por XML)
- Integración a nivel de aplicaciones: se basa fundamentalmente en compartir funcionalidad. Es una integración basada en APIs que exponen su funcionalidad a través del uso de interfaces que serán tanto más portables dependiendo del lenguaje utilizado para definirse. (Ej: IDL de CORBA o WSDL de los *Web Services*)

La integración de datos dirigida por procesos [Bazán P.,2009] ayuda a enriquecer los servicios de negocios SOA y los procesos BPM a través de una secuencia de servicios de datos combinados de manera reusable que incorpora la intervención de tareas humanas transformando la información en exacta, consistente y oportuna.

La integración de aplicaciones tiene por objetivo entender y usar las interfaces para acceder a la funcionalidad requerida y enmascarar u ocultar las diferencias tecnológicas usadas por cada interfaz en su acceso. Esto último se lleva a cabo con servicios que exponen sus interfaces [Juric M. et al., 2007].

### **Escenarios de integración de datos dirigida por procesos**

Entre las particularidades que presenta un escenario de integración dirigido por procesos podemos encontrar: mantenimiento de la integridad transaccional, manejo de tareas de larga duración o múltiples pasos, colaboración entre las personas y las actividades de integración y el soporte para integración de datos analítica.

- *Manejo del rechazo o datos inválidos, para mantener la integridad transaccional.* Los procesos o aplicaciones deben tener en cuenta cómo responden los procesos de integración de datos ante entradas erróneas o respuestas recibidas fuera de tiempo. Sin duda el manejo de excepciones consume cerca del 80% del tiempo en la construcción de los procesos de negocios.
- *Manejo de tareas de integración de datos de larga duración y múltiples pasos.* La integración no puede hacerse en un único paso y necesita ser modelada como un workflow orquestado.
- *Promover una mejor colaboración entre las personas y las actividades de integración de datos.* La integración de datos basada en procesos puede ayudar a coordinar esfuerzos a través de múltiples grupos. La orquestación junto con el workflow humano puede asistir con la revisión y aprobación de ciertos datos que caen bajo la responsabilidad de varias personas.

- *Soporte para BI e integración de datos analítica.* Cuando un KPI que se monitorea en una consola de BI atraviesa ciertos umbrales, se requiere la intervención de humanos que tomen acciones correctivas.

### Método para la integración basada en servicios

La orientación a servicios es una filosofía de diseño que hace que los recursos de IT estén disponibles sobre una red en forma independiente de su ubicación [Oracle, 2007]. Es posible delinear dos maneras de introducir esta filosofía de orientación a servicios en una organización.

La orientación top-down o dirigida por procesos, parte a partir organización y va refinando hacia sus áreas funcionales identificando actividades, quién las realiza y bajo qué restricciones.

La orientación bottom-up o dirigida por aplicaciones, se inicia a partir de las aplicaciones para identificar componentes funcionales que puedan combinarse para dar respuesta procesos de grano más grueso.

La Tabla 7 enuncia las actividades de cada método y la Figura 23 muestra un ejemplo acerca de cómo asociar servicios a actividades, procesos y sub-procesos.

**Tabla 5. Integración top-down y bottom-up**

Método	Actividades
Top - down o dirigida por procesos, la organización se modela a sí misma desde lo general a lo particular.	Identificar procesos de negocio y eventos.
	Reunir requerimientos de negocio de cada proceso.
	Descomponer procesos hasta un grado de detalle atómico (actividades y servicios).
	Coordinar la descomposición dentro de un flujo de procesos transversales a la organización.
Bottom-up o dirigida por aplicaciones, a partir de ellas se detectan áreas debajo acoplamiento y reusabilidad para luego componer servicios de grano más grueso	Identificar la funcionalidad dentro de las aplicaciones para ser publicados como servicio.
	Exponer y publicar los servicios.
	Orquestar los servicios expuestos de modo que logren alcanzar la funcionalidad de las aplicaciones existentes.
	Reevaluar la cartera de aplicaciones para eliminar redundancias.

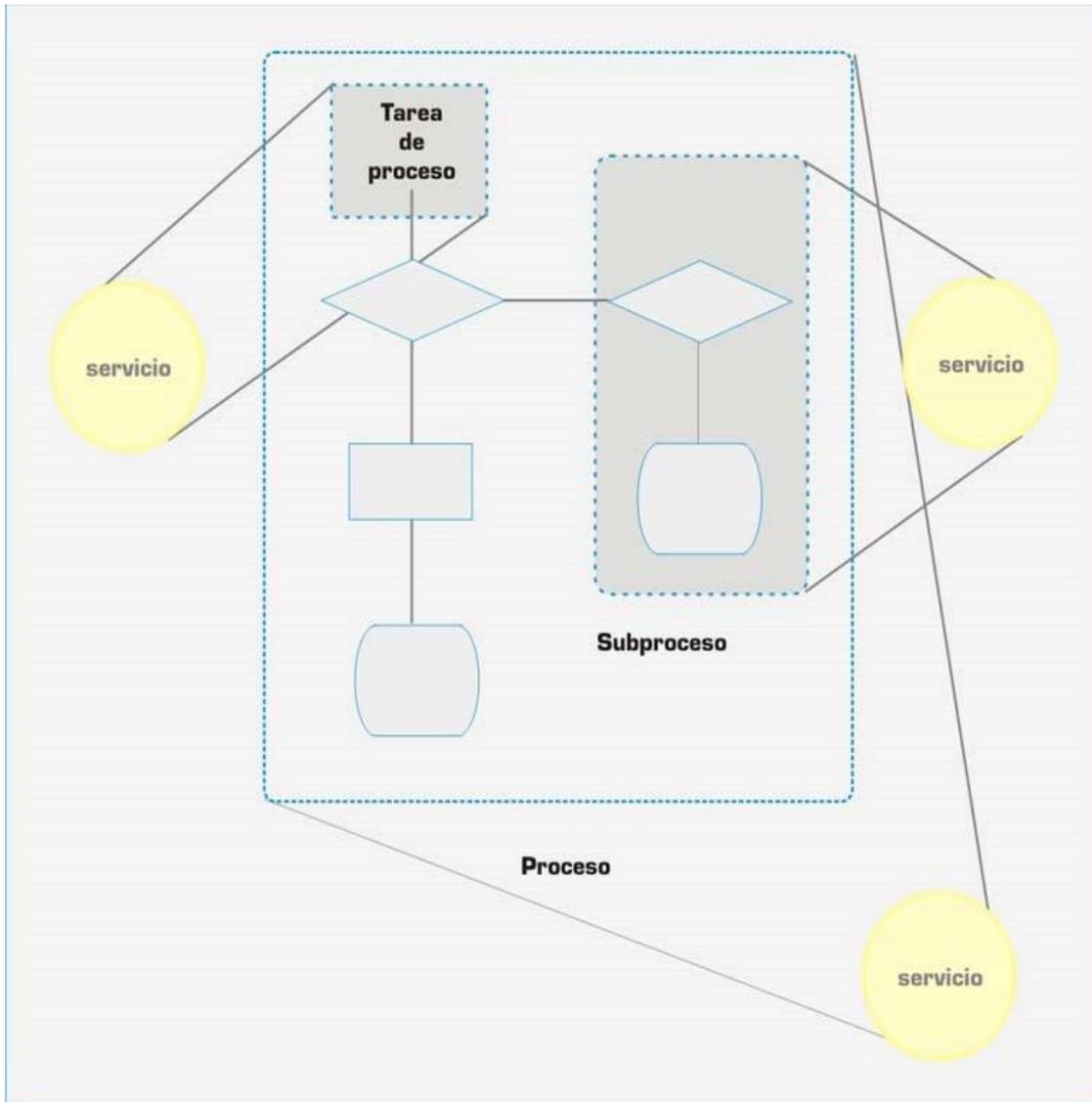


Figura 24. Asociación de servicios a actividades, procesos y sub-procesos

### Componentes de la infraestructura para la integración

Otro de los elementos de un modelo de integración de aplicaciones, aplicables cada uno al escenario o enfoque de integración abordado, se enuncian en esta sección como los elementos que componen la infraestructura de la integración. Entendiendo por infraestructura al “conjunto de medios técnicos e instalaciones necesarias para llevar a cabo un actividad”.

### Comunicación

La comunicación asegura a los desarrolladores un nivel de abstracción tal, que pueden independizarse de los detalles de bajo nivel asegurando que proveedor y consumidor de servicio puedan encontrarse.

Pueden utilizarse sistemas de comunicación asincrónicos basados en mensajería o bien sincrónicos vía un broker de objetos o bien un ESB.

Estos *middleware* usan protocolos estándar como SOAP (*Simple Object Access Protocol*), HTTP, TCP/IP, IIOP (*Internet Inter-ORB Protocol*) [Kettinger et al., 1997].

### **Ruteo y mediación**

El ruteo y mediación es un nivel que adapta el nivel de comunicación entre aplicaciones de tal modo que las mismas puedan interoperar [Juric M. et al., 2007]. Entre las responsabilidades que tiene este nivel está la de lograr que datos provenientes de distintas fuentes representen un concepto de negocio.

### **Transformación**

La transformación de las estructuras de datos ha sido siempre un problema resuelto de manera puntual, escribiendo código ad-hoc que leía y transformaba al formato destino de manera puntual.

La aparición de los lenguajes de marcado y en particular el XML como estándar de facto para el intercambio de datos, otorgaron un mayor nivel de madurez a la transformación de los datos [Juric M. et al., 2007].

La transformación hoy puede considerarse como un servicio provisto por las máquinas de transformación basadas en XSLT (*EXtensible Stylesheet Language for Transformations*) que producen transformaciones independientes del lenguaje y la plataforma.

### **Coordinación y Orquestación**

En un enfoque dirigido por procesos es preciso sincronizar las actividades de integración definiendo formalmente los servicios que requieren los procesos y las aplicaciones, secuenciándolos en una orquestación.

En este sentido, una infraestructura de integración debe contar con algún mecanismo de coordinación inter-procesos o intra-procesos que ordene los pasos a seguir para conducir los servicios.

La orquestación ha evolucionado desde el enfoque manual al automatizado. En el primer caso, se reducía a código inyectado en las aplicaciones que resolvía integración punto a punto y que tenía embebida la lógica del *workflow*. Esto resultaba difícil de mantener y de modificar. El enfoque automático provee una mayor flexibilidad, desacoplando procesos de datos y produciendo un servicio de datos independiente que inicia una tarea de integración de datos de bajo nivel como parte del proceso de *workflow*.

Este último enfoque, el automático, es solamente posible si los servicios están construidos de modo adecuado como módulos autocontenidos sin las dependencias típicas de la programación procedural.

### **Transacción**

Uno de los principales aportes al definir una infraestructura de integración es la de proveer un mecanismo para llevar a cabo las operaciones de modo transaccional, esto es, la capacidad de invocar operaciones sobre diferentes sistemas soportando la semántica del modelo transaccional bajo las propiedades ACID (*Atomicity, Consistency, Isolation, Durability*) [Juric M. et al., 2007].

Estas propiedades garantizan la preservación de la consistencia del sistema, aíslan las operaciones entre sí, otorgan persistencia a los cambios y además son atómicas.

### Seguridad

Desde el punto de vista de la integración, la seguridad conlleva los mismos conceptos tradicionales: autorización, autenticación y auditoría. En este sentido, la infraestructura de la integración debe proporcionar los medios para limitar el acceso al sistema, hacerlo de una manera unificada y dejar rastros de dichos accesos.

### Actores que participan en la integración

Los actores que participan en un escenario de integración de aplicaciones, tienen la finalidad de instanciar los componentes del modelo, elegir un tipo de integración posible, identificar el escenario y aplicar el método elegido.

Los actores pueden o no estar representados por seres humanos, pudiendo reemplazarse actores no humanos por humanos cuando la solución tecnológica del modelo de integración así lo requieran. La Tabla 8 describe los diferentes actores o roles y su función.

**Tabla 6. Actores/Roles participantes en la integración de aplicaciones**

Actor/Rol	Función
Analista de negocio	Identificar los procesos que deben llevarse a cabo para alcanzar el objetivo de integrar sistemas de software en el marco de una visión de procesos de la organización. Eventualmente, modela los procesos en alguna herramienta de diseño basada en BPMN.
Arquitecto del sistema	Identificar funcionalidades existentes y transformarlas en insumos para los procesos a cubrir. Refinar los modelos realizados por el analista mejorando los detalles que se requieren para identificar componentes.
Coordinador de servicios	Aplicar los flujos y condiciones que deben cumplirse para que los servicios y componentes identificados por el arquitecto trabajen en forma coordinada (orquestada) para alcanzar el objetivo del negocio. Este actor puede ser reemplazado por un software.
Desarrollador de componentes y servicios	Codificar las interfaces necesarias y desarrollar los componentes o servicios que se requieran según lo han delineado los procesos identificados y descritos dentro del modelo de integración.

Como se muestra en la Figura 24, los actores del modelo de integración trabajan de manera colaborativa en cada nivel.

Los Analistas de Negocios definen el modelo de procesos y sus instancias, en el contexto de la composición de servicios y transformaciones de datos.

Los Arquitectos del Sistema trabajan sobre los servicios a nivel de composición y luego articulan con el Desarrollador de Componentes al momento de realizar la orquestación BPEL y definir el transporte y publicación de los servicios.

El Coordinador de Servicios realiza y da soporte al despliegue a nivel de Operación y Gestión y también trabaja junto al Desarrollador de Componentes a nivel de orquestación y transformación XML.

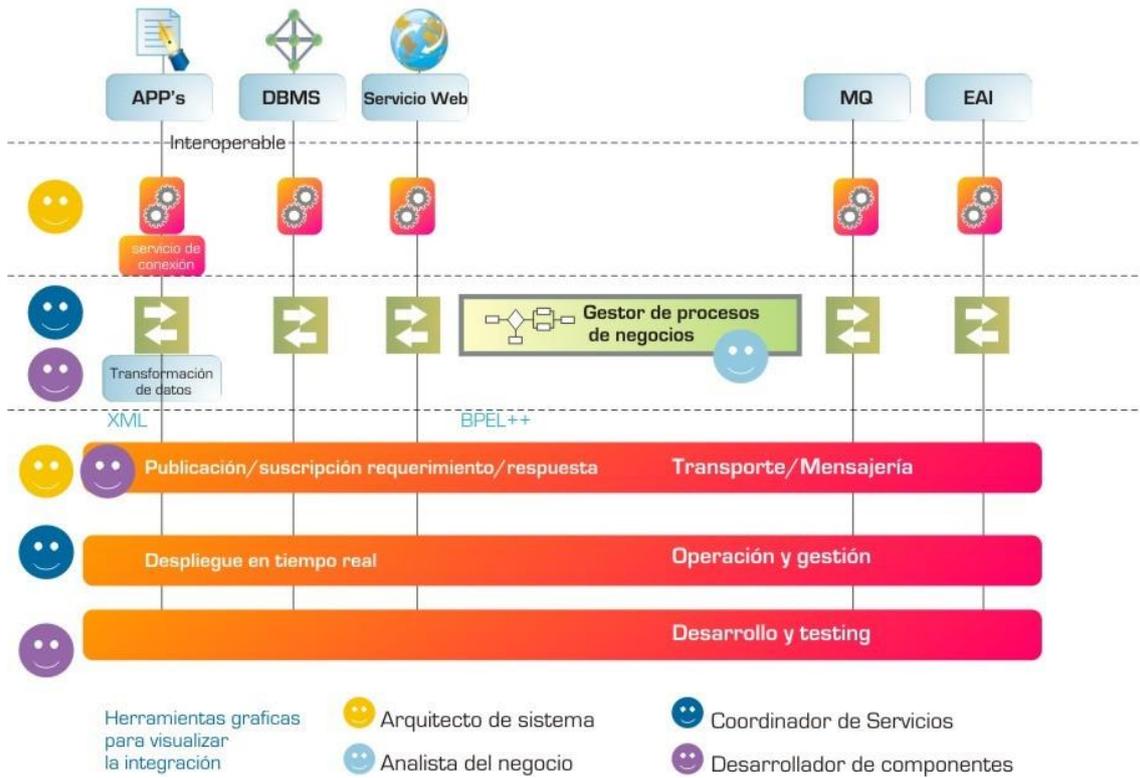


Figura 25. Escenario de un modelo de integración y sus actores

### 3.4 Conclusiones

En escenarios típicos de integración de aplicaciones, la funcionalidad de los sistemas de integración está organizada en forma de un proceso. Este proceso consiste en la ejecución de un conjunto de actividades con restricciones de ejecución, apuntando a la obtención de un objetivo. En estos escenarios la representación de los procesos está embebida en las reglas que los agentes manejan, y parece ser más apropiado alcanzar una representación explícita del mismo.

La integración por línea de base siempre es posible, pero hacer una verdadera integración significa reconocer que la orquestación de servicios y la gestión de procesos de negocios no son lo mismo y que usar adaptadores para integrar a modo de APIs directamente desde el modelo de procesos, es diferente a usar servicios de negocios gestionados por tecnología SOA, como lo es un bus de servicios o la registración de servicios. También significa reconocer

que la tarea humana juega un importante rol en muchos procesos de negocios y que cualquier plataforma de BPM debe incluir soporte para *workflow* e interacción humana.

El desarrollo de aplicaciones con enfoque orientado a procesos comparte la mayoría de los conceptos vertidos en este modelo de integración, pero aplicado a la definición, especificación, desarrollo, puesta en marcha y seguimiento de proyectos, cuyo objetivo es modelar una organización en términos de los procesos que ejecuta.

Un modelo con enfoque a procesos y a servicios cuenta con el valor agregado de que sus conceptos se pueden utilizar para el desarrollo de soluciones computacionales nuevas o bien como mecanismos de integración de aplicaciones existentes, además de permitir capturar el conocimiento subyacente en la manera en que se ejecutan los procesos de negocio.

De acuerdo a investigaciones de Gartner, la integración de aplicaciones representa un 35% del costo total en diseño, desarrollo y mantenimiento de software en una organización típica, sea esta grande, mediana o pequeña. Por lo tanto, el aspecto crucial ya no es si se debe integrar sino directamente cómo hacerlo.