



Negocio Electrónico

Tema 2. Protocolos, tecnologías y lenguajes para negocio electrónico

Profesorado

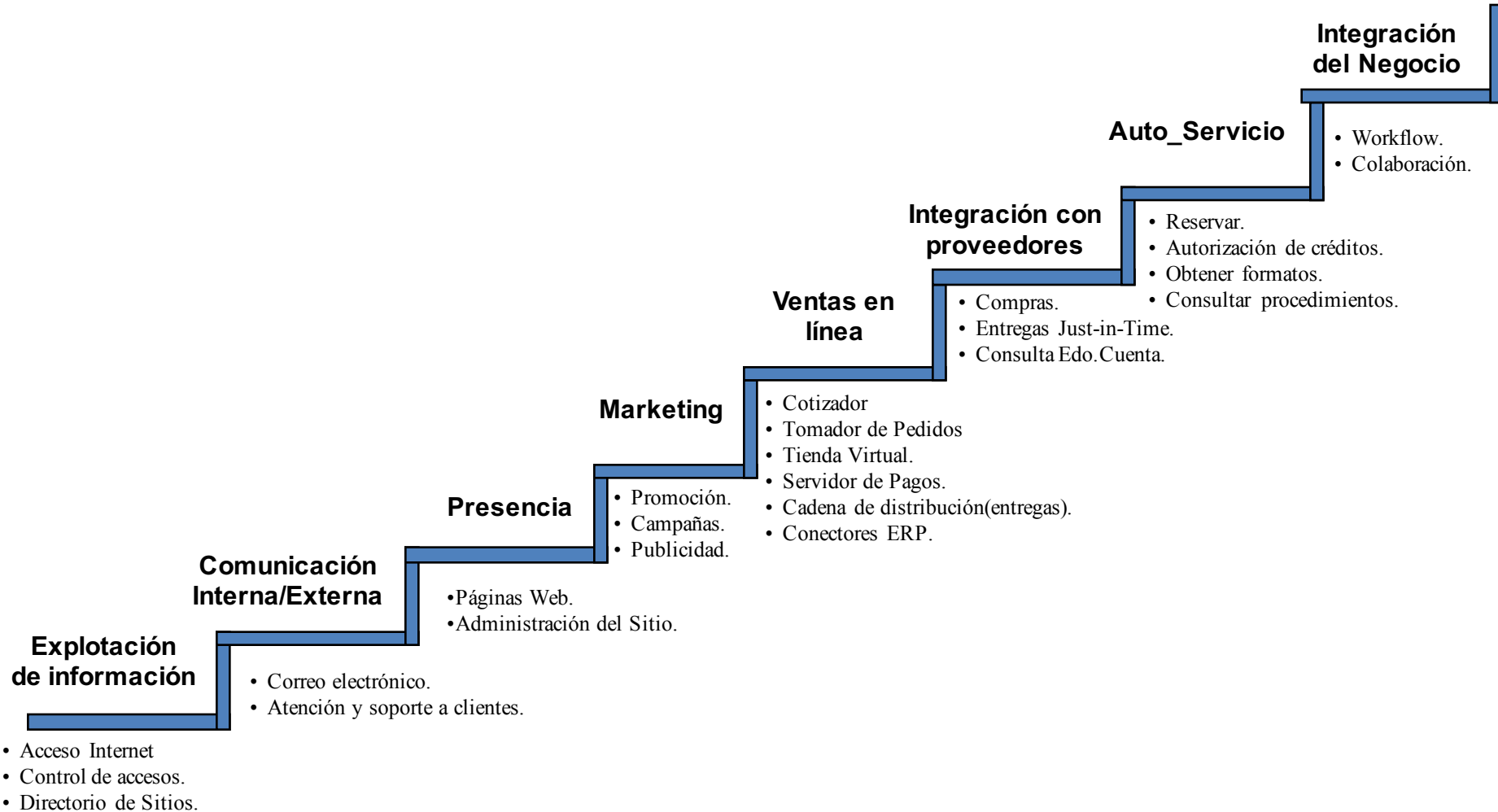
Antonio Muñoz Gallego

amunoz@lcc.uma.es

- **2.1. B2C. Tecnologías y soluciones**
- 2.2. B2B. Tecnologías e integraciones
- 2.3. Protocolos de transmisión de datos
- 2.4. Lenguajes para comercio electrónico

- Dale Traxler. “The Commerce EvRolution” (3 partes). Practical eCommerce. Agosto 2013
 - <http://www.practicalecommerce.com/articles/57618-The-Commerce-EvRolution-Part-1-Shopping-Trends>
- Graham Charlton. “How is ecommerce likely to evolve? Five key trends”. eConsultancy. Junio 2013
 - <http://econsultancy.com/es/blog/62927-how-is-ecommerce-likely-to-evolve-five-key-trends>
- **International Trends in E-Commerce - Part I. ChannelAdvisor. Agosto 2013.**
 - http://ebaystrategies.blogs.com/ebay_strategies/2013/08/international-trends-in-e-commerce-part-ii.html
- **International Trends in E-Commerce - Part II. ChannelAdvisor. Agosto 2013.**
 - http://ebaystrategies.blogs.com/ebay_strategies/2013/07/international-trends-in-e-commerce-part-1.html

Niveles de adopción de NE



- **PRESENCIA**
 - Información a clientes
- **MARKETING**
 - Campañas, ofertas...
- **EXPLOTACIÓN**
 - Capacidad de realizar transacciones
- **TRANSFORMACIÓN**
 - Rediseñar o reorientar todo el negocio

- **Puntos fuertes**
 - comodidad,
 - horario
 - ahorro de tiempo,
 - información,
 - ahorro de costes,
 - posibilidad de comparar precios,
 - acceso a productos difíciles de encontrar.
- **Puntos débiles**
 - desconfianza del proveedor,
 - desconfianza los medios de pago,
 - desconfianza del producto,
 - costes y plazos de envío,
 - estrés web

- **Qué funciona en Internet?**
 - Lo que sea útil:
 - *fuentes de información,*
 - *agregación y análisis de información*
 - *servicios de valor añadido*
 - Lo que es fácil de usar
 - Lo que te aporta un incentivo
 - Lo que es mejor que la realidad
 - Lo que sea personalizado
 - ... dos más?

- **Tecnologías específicas**
 - Se han desarrollado diversas tecnologías para ayudar a compradores y/o vendedores durante el proceso de compra B2C.
- **Claves:**
 - Inteligencia
 - *A menudo, estas soluciones utilizan técnicas inteligentes para optimizar/automatizar procesos*
 - Automatización
 - *Dada la gran cantidad de información (de potenciales compradores, vendedores y productos), automatizar el procesamiento puede ser crucial.*

- **Compras**
 - Comparación
 - Recomendación
 - Selección
 - Negociación
 - Post-venta
- **Estudios de opinión**
 - Estudios de mercado
 - Identificación de nichos
 - Evaluación de opiniones

- **Recomendaciones automáticas.**
 - Recomendaciones personalizadas de productos a cada usuario
- **Tecnologías:**
 - Sistemas basados en Reglas
 - *Declarativas*
 - *Matrices de recomendación*
 - *Sistemas de perfiles*
 - Filtrado colaborativo
 - *Los usuarios evalúan una serie de ítems.*
 - *El algoritmo identifica:*
 - Para un usuario, encontrar a otros usuarios que suelen evaluar los ítems de la misma forma y recomendarles productos que esos otros usuarios han evaluado bien y que él no ha visto.
 - Para un ítem, encontrar los ítems que suelen ser evaluados de la misma forma por casi todos los usuarios. Si a un usuario le gusta un ítem del grupo, recomendarle el resto.
- **Ejemplos: Amazon, Netflix, etc.**

- **Evaluación de Opiniones**
 - Combinan técnicas de procesamiento de lenguaje natural y de minería de datos.
- **Técnicas:**
 - Sentiment Classification
 - *Dado un texto, se intenta determinar si el sentimiento general es positivo o negativo.*
 - Feature-based Opinion Mining
 - *Intenta extraer frases valorativas sobre características concretas (“la batería del portátil ACME dura muy poco”).*
 - Comparative sentence mining
 - *Busca frases comparativas (“la batería del portátil MECA dura más que la del ACME”).*

- **Pagos**
 - Pasarelas de pago
 - Sistemas de pago
 - Micropagos
 - Moneda virtual
- **Medios sociales**
 - Comunidades virtuales
 - Redes sociales
 - Sistemas de información
 - Peticiones y activismo

- **Tecnologías no específicas**
 - Web semántica
 - *XML, Ontologías (RDF, OWL, ...)*
 - *Identificadores únicos. DOI, URI, OKKAM*
 - Federaciones
 - *Identidad, servicios*
 - Servicios Web

- 2.1. B2C. Tecnologías y soluciones
- **2.2. B2B. Tecnologías e integraciones**
- 2.3. Protocolos de transmisión de datos
- 2.4. Lenguajes para comercio electrónico

- Transmisión electrónica de documentos comerciales y administrativo-contables (pedidos, facturas, etc.) entre aplicaciones
- Envío de documentos en un formato normalizado para proceso automático
- EDIFACT (Electronic Data Interchange For Administration, Commerce and Transport)
 - “ISO 9735 - Intercambio Electrónico de Datos para la Administración, Comercio y Transporte (EDIFACT) - Reglas de aplicación de la sintaxis”
- <http://re-orientation.com/manual-para-entender-edi-edifact>

- Electronic Commerce Modeling Language (**ECML**) define un conjunto estándar de nombres de campos XML organizados jerárquicamente para facilitar la automatización de las transacciones electrónicas
- Universal Business Language (**UBL**) es una librería de documentos XML estandarizados para negocios (incluye facturas, órdenes de compra, etc.).
- Electronic Business using eXtensible Markup Language, (**ebXML**) es un conjunto de estándares basados en XML para fomentar el uso estandarizado, interoperable y seguro de información de negocio.

- **Mercados B2B verticales**
 - Se centran en sectores de una actividad
 - Ofrecen servicios como petición de ofertas, concursos, licitaciones
 - Permiten comparar los precios, la calidad, los servicios
 - El beneficio para la empresa es la rapidez de las transacciones

- **Mercados B2B verticales**
 - Ejemplos: <http://www.ghx.com> (Salud)
 - <http://www.avendra.com> (Hostelería)
 - <http://www.1worldsync.com> (Múltiples)

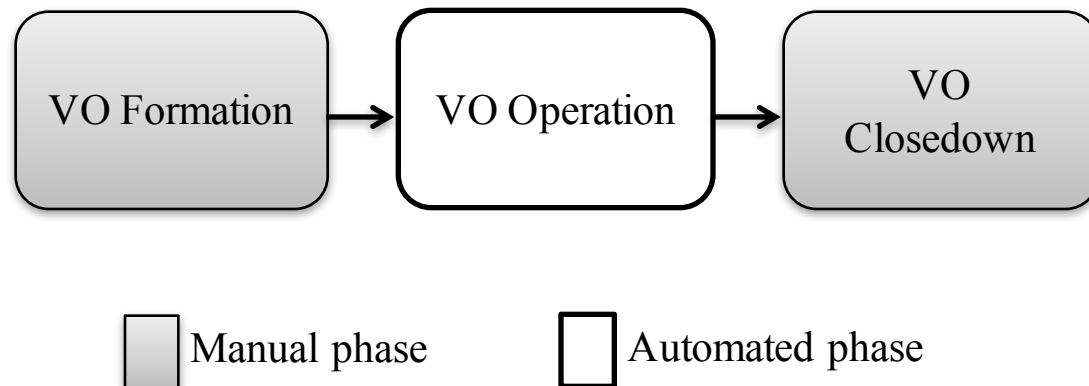
 - Empresa compradora realiza su petición
 1. Los proveedores seleccionados son notificados (por e-mail o por sistemas propios)
 2. Los proveedores realizan sus ofertas
 3. Los compradores adjudican las ofertas e informan a los proveedores

- **Mercados B2B verticales**
 - La información es confidencial
 - El intermediario simplifica al máximo todas las gestiones de compra-venta
 - Se proporciona reducción de costes
 - Oportunidades de negocio y acceso a nuevos clientes
 - Cuota anual – Suele ser en forma de comisión (aprox. 1%) que se factura al proveedor

- **Mercados B2B horizontales**
 - Permiten comprar, vender, subastar cualquier tipo de producto
 - *Agregadores de catálogo*
 - *Subastas*
 - *Ventas directas*
 - ...
 - Facilita operaciones de compra-venta entre empresas de todos los sectores y países
 - Ejemplo: Opciona <http://www.opcion.com>
 - Desgraciadamente ya cerrado ☹

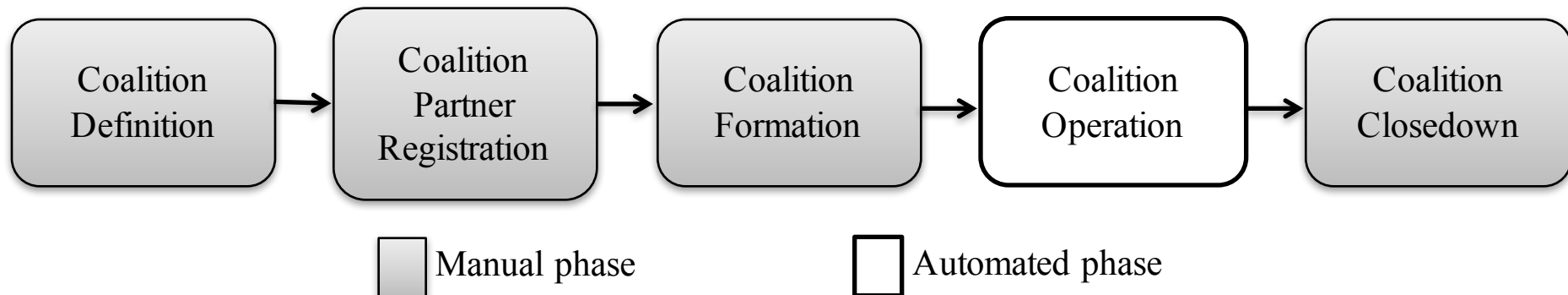
- **Virtual Organizations (VO)**

- Se trata del equivalente electrónico de una UTE
- Una Organización Virtual se forma para responder a una necesidad específica, la atiende y luego se disuelve
- Normalmente esto ocurre en un tiempo “largo” lo que permite una formación “manual”

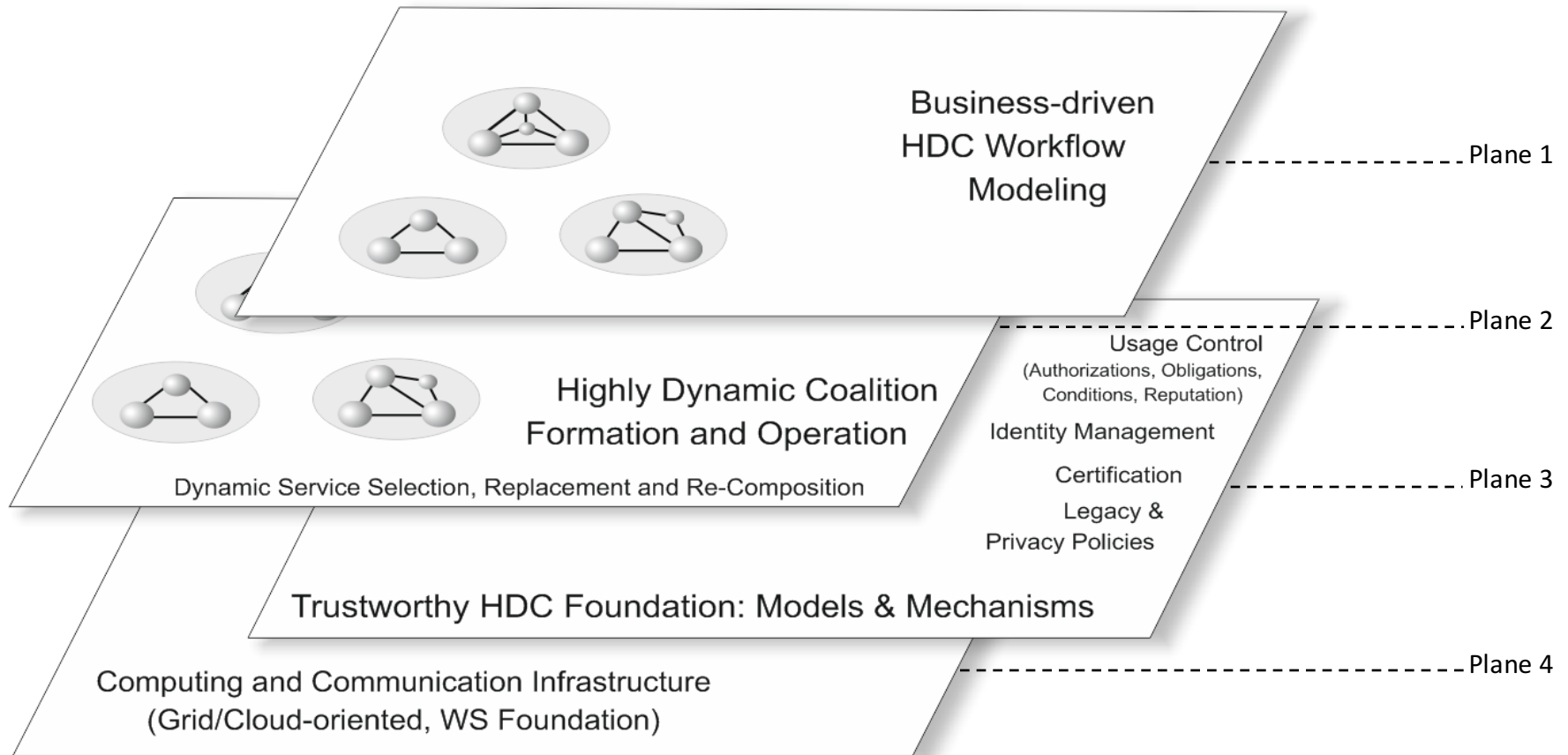


- **Dynamic Coalitions (DC/HDC)**

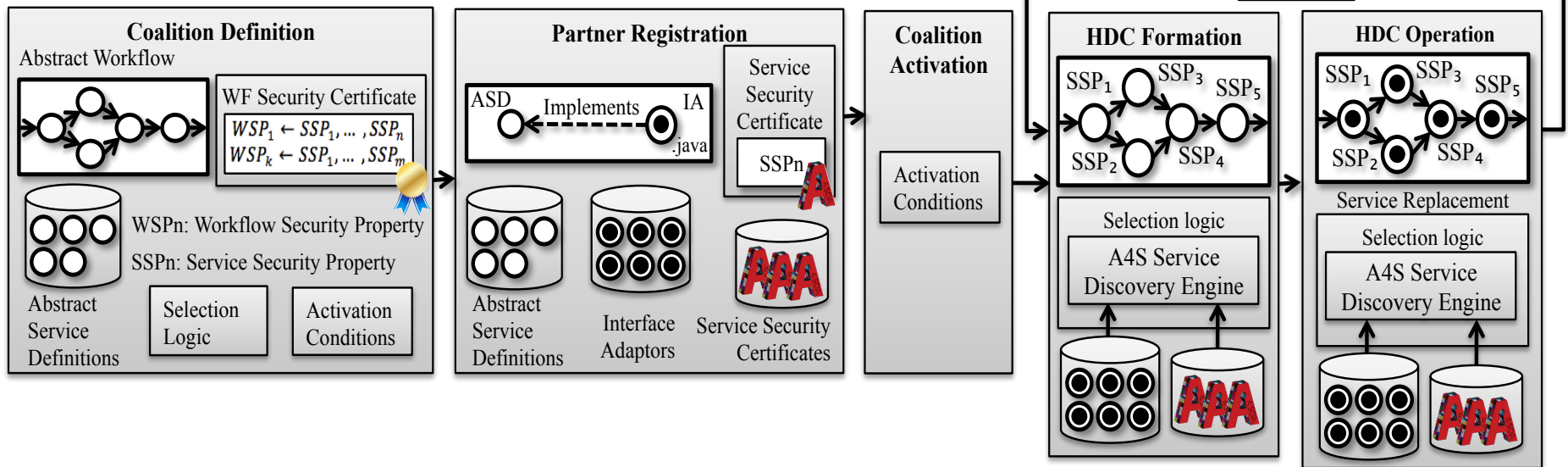
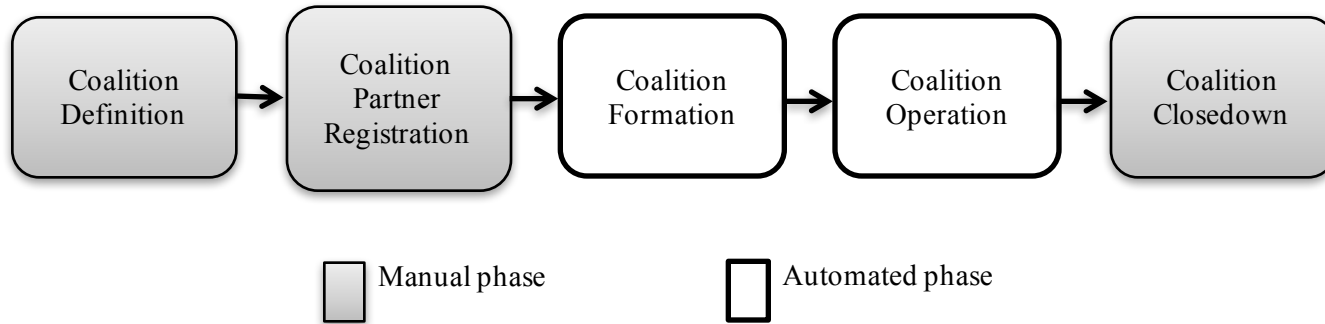
- Se trata de una versión más avanzada y flexible de las organizaciones virtuales
- Trabajan en tiempos “cortos” por lo que se requiere una formación automática
- Se forman para atender una petición de cliente
- Se proporcionan mecanismos para la interoperabilidad (adaptación de interfaces, control de acceso, etc.)
- Suelen basarse en Servicios Web



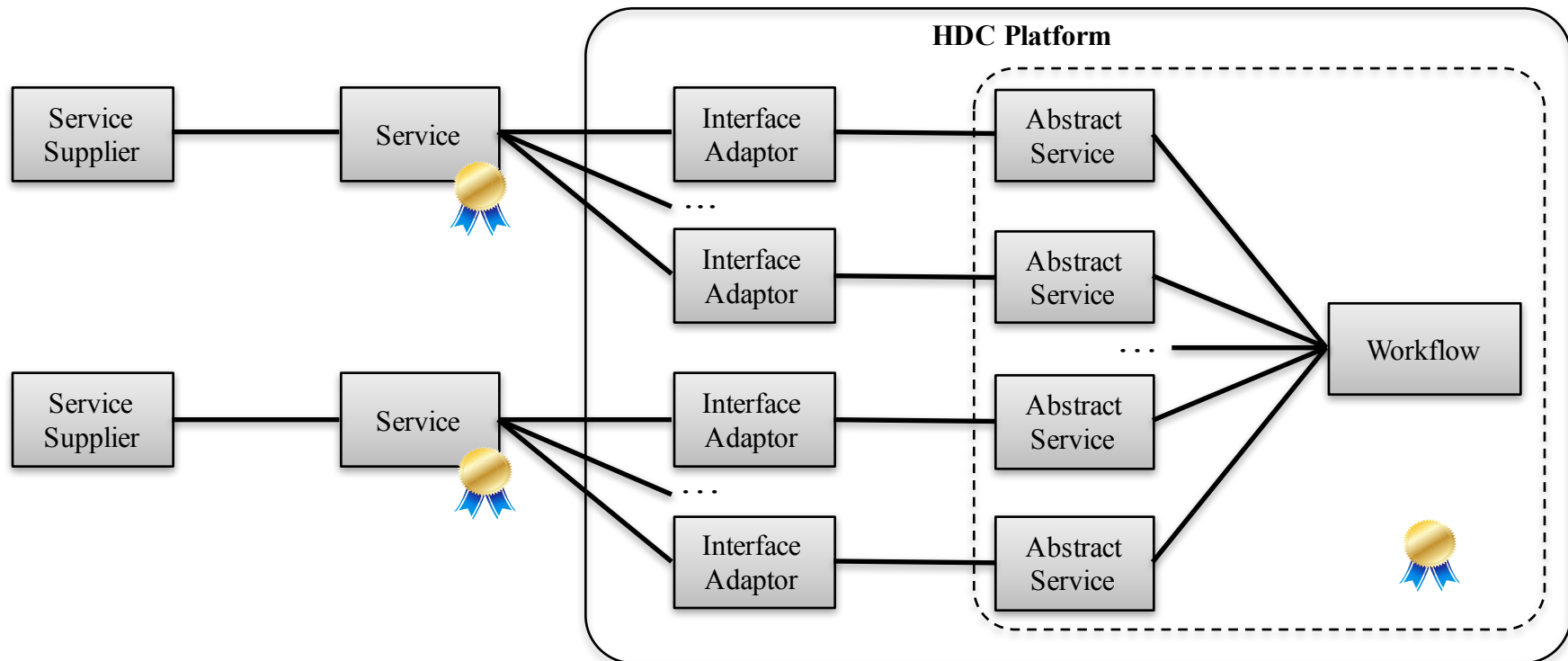
- **Dynamic Coalitions**



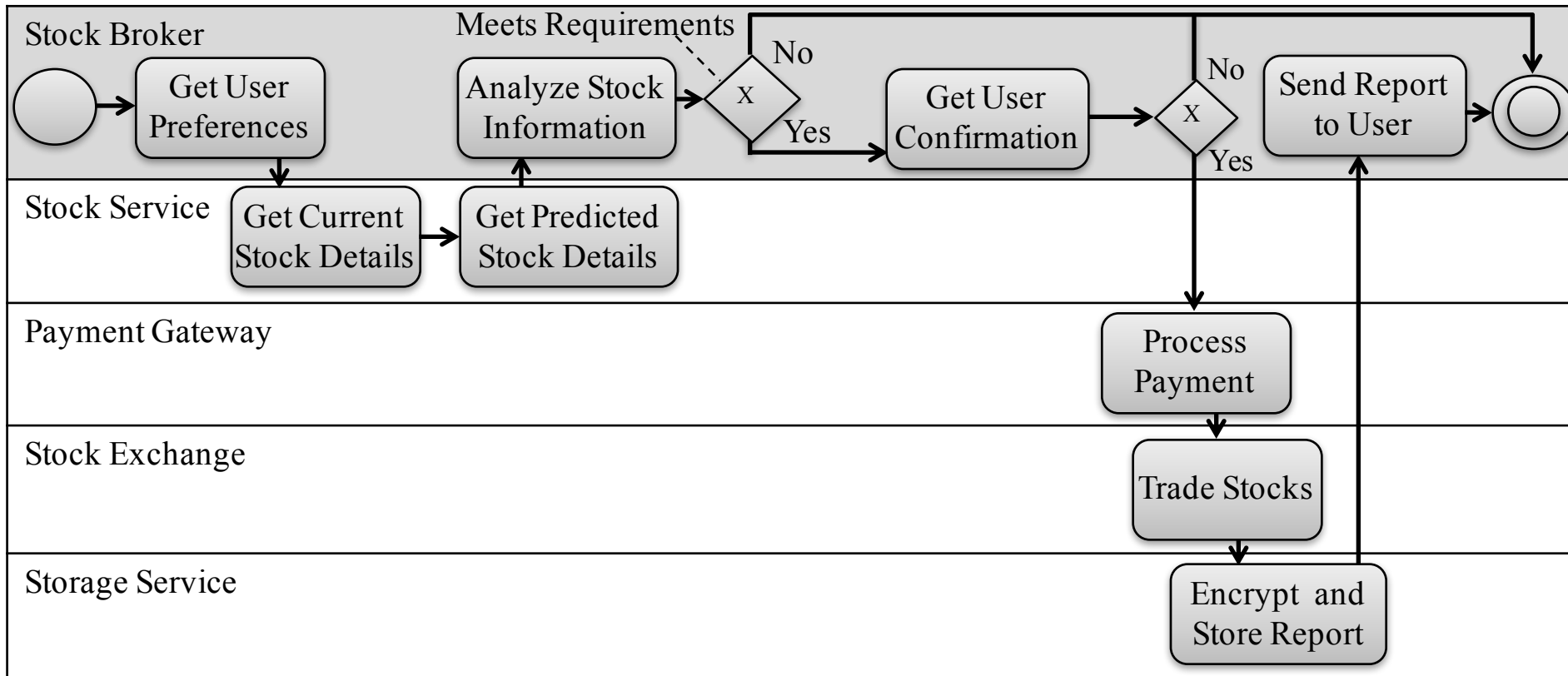
- Highly Dynamic Coalitions (HDCs)**



- **Highly Dynamic Coalitions**



- **Dynamic Coalitions**
 - Ejemplo en inversión en bolsa



- **A2B: Administration to Business**
 - Abarca transacciones entre empresas y organizaciones gubernamentales
 - Publicación de disposiciones administrativas; intercambio electrónico de impuestos, pago de tasas, cotizaciones SS, etc;
 - Información de subvenciones; trámite para creación de empresas, etc.
 - <http://www.060.es/>
 - [EC Participant Portal](#)

- **A2C: Administration to Citizen**
 - Abarca transacciones entre ciudadanos y organizaciones gubernamentales
 - Ventanilla única electrónica
 - Empleo público, autorizaciones para instalaciones publicas, consulta trámites legales, pago de impuestos, dar de alta en el padrón, tarjeta de descuento de transporte, conseguir certificado, etc.

- **A2C: Administration to Citizen**
 - Ejemplo: Seguridad Social
 - <http://www.seg-social.es>
 - <http://www.agenciatributaria.es>

- **A2A: Administration to Administration**
 - Conectan diferentes entidades de la administración
 - *Portal de compra de la Junta de Extremadura*
 - <https://contratacion.gobex.es/>

- **Un Servicio Web es un recurso programable y accesible mediante una URL.**
- **Los Servicios Web permiten que:**
 - Las aplicaciones compartan información.
 - Las aplicaciones invoquen funciones de otras aplicaciones
- **Página web vs. Servicio Web:**
 - La PÁGINA puede ser visitada por cualquier individuo interesado.
 - El SERVICIO sólo lo visitan programas que lo requieren.
- **El conjunto de Servicios Web en Internet es una World Wide Web paralela, de carácter no humano, sino cibernético.**

- **Los Servicios Web son la piedra angular de cualquier desarrollo de sistemas distribuidos actual**
- **Los ordenadores hablan unos a otros a través de la web usando HTTP y otros protocolos.**
- **Un servicio web no tiene interfaz gráfica**
- **Provee una API de métodos que pueden ser invocados en la web**
- **Diseñados para proveer “servicios”**

- **Un Web Service es una clase que se auto-define usando XML, y accesible mediante HTTP.**
 - Autodefinida con XML, para ofrecer al exterior una interfaz estándar con el que utilizar los métodos que proporciona.
 - Accesible mediante HTTP por ser este el protocolo de aplicación utilizado por los navegadores web.
- **¿para qué sirve un Servicio Web? Para lo mismo que sirve en programación una clase o un conjunto de métodos.**

- **Los servicios web han dado lugar a un nuevo modo de diseñar sistemas distribuidos:**
 - Arquitecturas SOA (Service Oriented Architecture)
- **SOA = colección de servicios**
 - Más información en <http://www.service-architecture.com/>

- **Se espera que en el futuro los negocios expondrán aplicaciones a clientes como Servicios Web con pago por su uso**
- **Los sistemas de diferentes empresas competirán y cooperarán unos con otros a través de la Web**
- **Los servicios web serán el centro de todo un nuevo ecosistema de computación apoyado por otras tecnologías como Virtualización y Cloud Computing**

- **Web Services Protocol Stack**: conjunto de servicios y protocolos de los servicios Web.
- **XML** (Extensible Markup Language): Es el formato estándar para los datos que se vayan a intercambiar.
- **SOAP** (Simple Object Access Protocol) o **XML-RPC** (XML Remote Procedure Call): Protocolos sobre los que se establece el intercambio.
- Otros protocolos: los datos en XML también pueden enviarse de una aplicación a otra mediante protocolos normales como **HTTP** (Hypertext Transfer Protocol), **FTP** (File Transfer Protocol), o **SMTP** (Simple Mail Transfer Protocol).
- **WSDL** (Web Services Description Languages): Es el lenguaje de la interfaz pública para los servicios Web. Es una descripción basada en XML de los requisitos funcionales necesarios para establecer una comunicación con los servicios Web.
- **UDDI** (Universal Description, Discovery and Integration): Protocolo para publicar la información de los servicios Web. Permite a las aplicaciones comprobar qué servicios web están disponibles.
- **WS-Security** (Web Service Security): Protocolo de seguridad aceptado como estándar por OASIS (Organization for the Advancement of Structured Information Standards). Garantiza la autenticación de los actores y la confidencialidad de los mensajes enviados.

- **En general los Servicios Web se invocan por medio de protocolos estándar basados en XML.**
- **Existen dos grandes tendencias: XML-RPC y SOAP, estandarizadas por el consorcio W3C.**
- **Pueden atravesar firewalls y routers, que lo tratan como una comunicación HTTP.**
- **La diferencia entre SOAP, REST y XML-RPC es su complejidad.**
 - XML-RPC está diseñado para ser sencillo, pero sigue usando XML
 - SOAP está creado con idea de dar un soporte completo y minucioso de todo tipo de servicios web.
 - SOAP incluye una infraestructura a su alrededor. No es un mero protocolo de comunicación entre ordenadores, sino que además se rodea de términos como WSDL y UDDI.
 - REST surgió para facilitar situaciones que con SOAP son complejas, pero a veces es más simple usar SOAP
- **Comparaciones**
 - <http://blog.smartbear.com/apis/understanding-soap-and-rest-basics/>
 - <http://stackoverflow.com/questions/11710507/xml-rpc-vs-rest>

- **HTTP. Se llama a las funciones con:**
 - Get
 - Post
- **XML**
- **SOAP, XML-RPC**
- **WSDL**
- **UDDI**

- **HTTP**

- Un esquema de direccionamiento comprensible.
- Arquitectura Cliente-Servidor.
 - *HTTP se asienta en el paradigma solicitud/respuesta. La comunicación se asienta sobre TCP/IP. El puerto por defecto es el 80, pero se pueden utilizar otros.*

- **HTTP**

- Es un protocolo sin conexión y sin estado.

- *Después de que el servidor ha respondido la petición del cliente, se rompe la conexión entre ambos.*

- Está abierto a nuevos tipos de datos.

- *Cuando un servidor HTTP transmite información de vuelta a un cliente, incluye una cabecera que le indica al cliente sobre los tipos de datos que componen el documento.*

- **GET**

- Forma en que normalmente se solicita una página Web

- **POST**

- En vez de adjuntar los parámetros al URL, se colocan en el mensaje de petición HTTP.

- **XML (eXtended Markup Language)**
 - Base de los servicios Web
 - Permite comunicación entre sistemas de forma estándar
 - Información con estructura
 - Utilizado ampliamente en la industria
 - Futuro estándar de estructuración de información

- **SOAP**
 - Protocolo de acceso a objetos simple
 - soporte completo y minucioso de todo tipo de servicios Web
 - Mas potente que XML-RPC
 - Incluye toda una arquitectura a su alrededor



Petición SOAP

```
<soap:Envelope
  xmlns:soap="http://schemas.xmlsoap.org/soap
/envelope/">
  <soap:Body>
    <getProductDetails
  xmlns="http://warehouse.example.com/ws">
      <productId>827635</productId>
    </getProductDetails>
  </soap:Body>
</soap:Envelope>
```



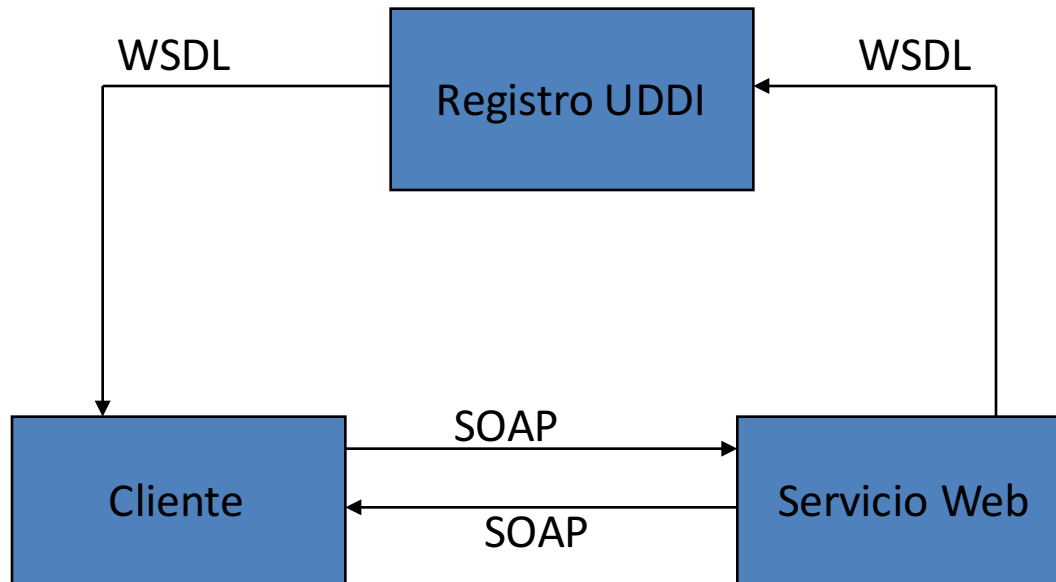
Respuesta SOAP

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <getProductDetailsResponse
      xmlns="http://warehouse.exempl.com/ws">
      <getProductDetailsResult>
        <productName>Toptimate 3-Piece Set</productName>
        <productId>827635</productId>
        <description>3-Piece luggage set.  Black Polyester.
        </description>
        <price>96.50</price>
        <inStock>true</inStock>
      </getProductDetailsResult>
    </getProductDetailsResponse>
  </soap:Body>
</soap:Envelope>
```

- **WSDL y UDDI**

- El archivo WSDL, en formato XML, indica al sistema cliente sobre qué servicios dispone
- WSDL además da una referencia precisa sobre el servicio, para poder invocarlo usando los parámetros adecuados
- UDDI es un Servicio Web en línea que se puede utilizar desde las aplicaciones para descubrir de forma dinámica otros servicios en línea

- **WSDL y UDDI**



- Aportan **interoperabilidad** entre aplicaciones de software independientemente de sus propiedades o de las plataformas sobre las que se instalen.
- Los servicios Web fomentan los estándares y protocolos basados en texto, que hacen más fácil acceder a su contenido y entender su funcionamiento.
- Al apoyarse en HTTP, los servicios Web pueden convivir con sistemas de **seguridad** como los cortafuegos sin necesidad de cambiar las reglas de filtrado.
- Permiten que servicios y software de diferentes compañías ubicadas en diferentes lugares geográficos puedan ser combinados fácilmente para proveer servicios integrados.
- Fomentan la **reusabilidad** del software

- **BPEL se usa para especificar colaboraciones de negocio que se implementan como composición de servicios web**
 - Captura la lógica de negocio y el comportamiento de las interacciones entre servicios
 - Posibilita la composición de servicios a nivel ejecutable
- **BPEL se basa en lenguajes imperativos, y se extiende con conceptos de SW y procesos de negocio**
 - **Mensajería:** send, receive, send/receive
 - **Concurrencia:** block-structured parallel execution, race conditions, event-action rules
 - **Tipado XML:** XML Schema, WSDL, XPath, XSLT

- **Se especifica**
 - *Activities* y orden de ejecución
 - *Partners* que interactúan con los procesos
 - *Data* de entrada y salida
 - *Messages* intercambiados
 - *Fault handing* si aparecen errores
- **Ejemplo de estructura simple**

```

1 <process name="..."
2   targetNamespace="http://www..." >
3   ...
4   <partnerLinks> ...
5   <messageExchanges> ...
6   <variables> ...
7   ...
8   <faultHandlers> ...
9   <eventHandlers> ...
10  ...
11   activity
12 </process>

```

Actividades simples BPEL

- *invoke*: invocación de operaciones ofrecidas por otros servicios
- *receive*: espera de respuestas de otros servicios
- *reply*: captura de interacciones
- *wait*: retrasos en el proceso del servicio
- *assign*: dá valor a variables
- *throw*: emite errores
- *rethrow*: propaga errores no tratados
- *compensate*: dispara un compensador cuando una transacción compuesta falla (el compensador devuelve el sistema al estado previo al inicio de la transacción compuesta)
- *empty*: no hace nada
- *exit*: finaliza un proceso inmediatamente

```
1 <invoke name = "..."  
2   partnerLink = "..."  
3   operation = "..."  
4   inputVariable = "..."  
5   outputVariable = "..."/> />
```

```
1 <receive name = "..."  
2   partnerLink = "..."  
3   operation = "..."  
4   variable = "..."  
5   createInstance = "yes/no"  
6   messageExchange = "..."/> />
```

Actividades estructuradas BPEL

- *sequence*: secuencia
- *flow*: paralelo
- *if*: condicional
- *pick*: errores
- *while*: bucle
- *repeatUntil*: bucle
- *forEach*: activación múltiple
- *scope*: agrupación
 - *Fault handler*
 - *Event handler*
 - *Compensation handler*

```
1 <scope name="..." >
2   <variables> ...
3   <partnerLinks> ...
4   <messageExchanges>...
5   ....
6   <eventHandlers>...
7   <faultHandlers>...
8   <compensationHandlers>...
9   ...
10  activity
11 </scope>
```

- **Los lenguajes, tecnologías y modelos en este entorno son altamente volátiles**
- **Existen tantos modelos como combinaciones de tipos de entidades**
 - Cada modelo tiene sus necesidades y cuenta con características y aspectos específicos
- **La seguridad suele ser un elemento esencial**
 - Cada modelo tiene necesidades diferentes
- **La tendencia es hacia un mayor dinamismo, inteligencia y automatización**
- **Modelos genéricos como HDC pueden llegar a representar un cambio significativo, especialmente en B2B2C**