



Hacia la Internet del Futuro: Web 3.0 e Internet de los Servicios

17 de Septiembre 2009, 10:00-13:00

Parque Tecnológico de Zamudio, Edificio ESI, #204

Dr. Diego Lz. de Ipiña Glz. de Artaza

dipina@eside.deusto.es

<http://paginaspersonales.deusto.es/dipina>

Agenda

- **10.00-10.25 La Internet del Futuro**
 - Limitaciones de la Internet actual
 - La Internet del Futuro: definición, objetivos y desafíos
 - Los pilares de la Internet del Futuro
 - Internet de los Servicios y de las Cosas
- **10.25-12.00 La Web del Futuro**
 - Evolución de la Web
 - ¿Qué es la Web 3.0?
 - El futuro de los navegadores web

11.20-11.45 Descanso

- El futuro de los navegadores, mensajería, buscadores web
- La Web como Plataforma de Servicios



Agenda

- **12.00-12.25 Cloud Computing**
 - Definición
 - Manifestaciones de Cloud Computing: SaaS, PaaS y IaaS
 - Aplicaciones Cloud más significativas
 - Plataformas Cloud: Google App Engine
 - Infraestructura Cloud: Amazon Web Services
- **12.25-12.50 Web Semántica**
 - ¿Qué es la Web Semántica?
 - Poniendo en práctica la Web Semántica: web semántica con minúsculas
 - Lenguajes de anotación semántica: RDF, OWL, RDFa, GRDDL, etc.
 - Los mash-ups web vs. Mash-ups semánticos
- **12.50-13.00 Conclusión y Preguntas**
 - Resumen de lo revisado
 - Sesión de preguntas y respuestas



3



Limitaciones de la Internet Actual (1)

- **Internet tiene más de 1500 millones de usuarios**
 - Es un éxito comunicando personas y sistemas de información
- **PERO ...**
 - **ha crecido mucho más allá de sus expectativas y objetivos de diseño en los 70s:**
 - Varios parches han permitido aplicaciones novedosas sin cambiarse su arquitectura subyacente
 - **debe superar sus limitaciones tecnológicas** para usarse como una infraestructura global crítica
 - **Las aplicaciones web del futuro requerirán más** movilidad, seguridad, ancho de banda, robustez e interactividad



4



Limitaciones de la Internet Actual (2)

- Varias “parches” han sido desarrollados y desplegados para permitir a Internet hacer frente a las **demandas incrementales de conectividad y capacidad**
- La **Internet actual ha alcanzado un punto de saturación** para alcanzar las expectativas funcionales del usuario y los desafíos tecnológicos en tiempo y precio
 - En términos de seguridad, escalabilidad, movilidad, disponibilidad o aspectos socioeconómicos



5



¿Qué es la Internet del Futuro?

- Término que resume los **esfuerzos para progresar a una mejor Internet**, bien mediante:
 - Pequeños pasos evolutivos incrementales o
 - Un rediseño completo (*clean slate*) y nuevos principios arquitectónicos
- Agentes clave:
 - **Global Environment for Network Innovations (GENI)**:
<http://www.geni.net/>
 - Más basado en redes
 - Proyecto **AKARI** en Japón, <http://akari-project.nict.go.jp/eng>
 - **Future Internet** – <http://www.future-internet.eu/>



6



Misión de la Future Internet (FI)

- Ofrecer a todos los usuarios un **entorno seguro, eficiente, confiable y robusto**, que:
 - Permita un **acceso abierto, dinámico y descentralizado** a la red y a su información y
 - Sea **escalable, flexible y adapte su rendimiento a las necesidades de los usuarios y su contexto**

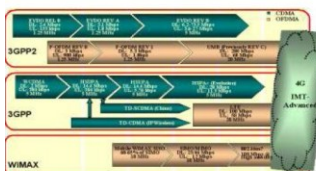
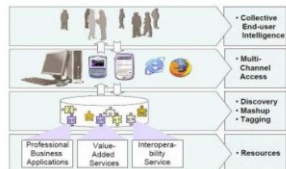


7



Visión de la Internet del Futuro

Internet of Services, Service Web



Networks of the Future

Trust



Security

3D Internet



Internet of Things

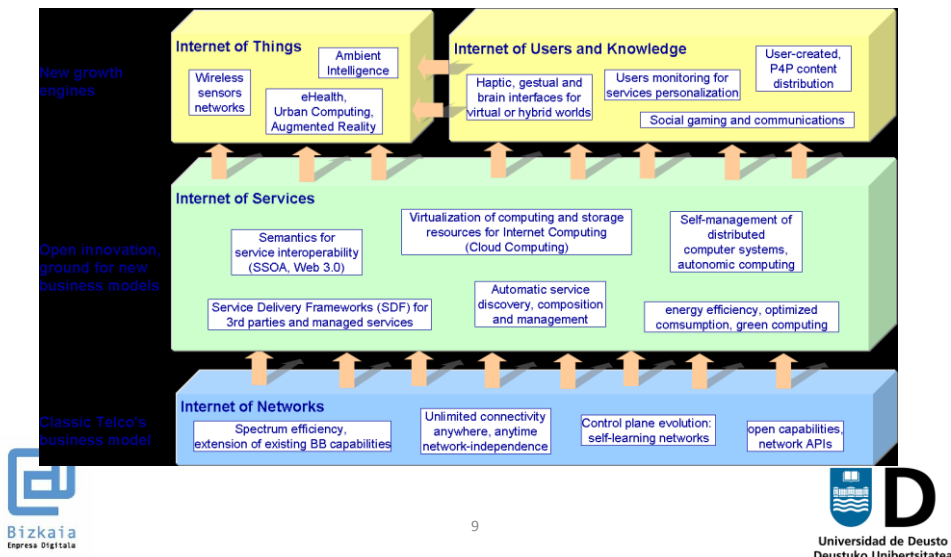


SOURCES: 3GPP, 3GPP2, Qualcomm, WIMAX Forum
<http://www.intu.osg.spu/publications/internetofthings/SecondLife>

8



Arquitectura de la Internet del Futuro

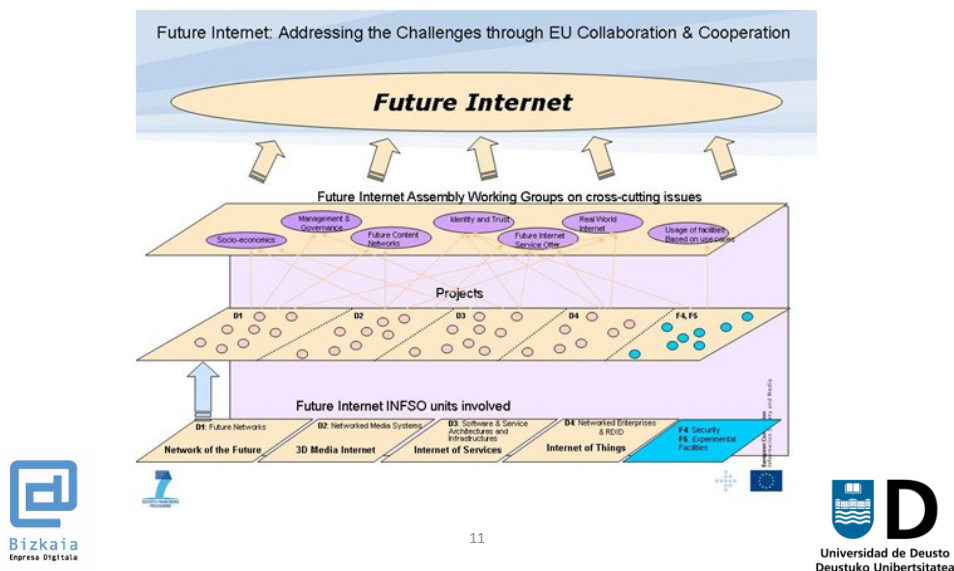


9

The BLED Declaration: Towards a European approach to the Future Internet

- “A significant change is required and the **European** Internet scientific and economic actors, researchers, industrialists, SMEs, users, service and content providers, now assert the **urgent necessity to redesign the Internet, taking a broad multidisciplinary approach, to meet Europe’s societal and commercial ambitions**”
 - http://www.future-internet.eu/fileadmin/documents/bled_documents/Bled_declaration.pdf

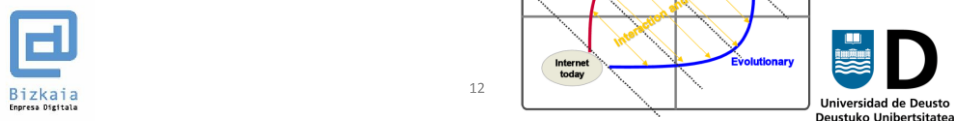
La Internet del Futuro en el 7º PM



11

Enfoques Evolutivo / Revolucionario (Clean-Slate)

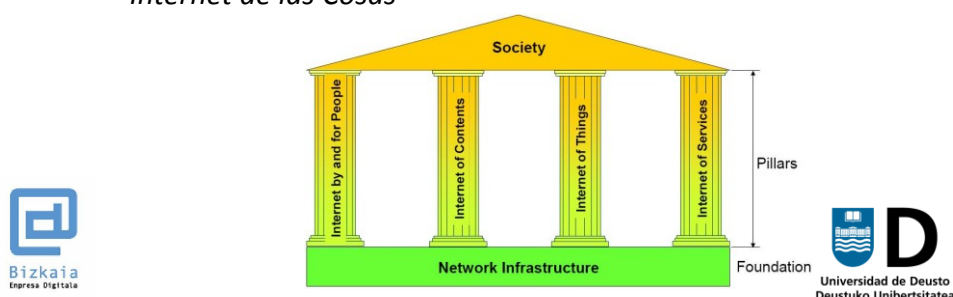
- Dos enfoques para hacer frente a los desafíos de Internet del Futuro:
 - **Evolutivo** – se basa en la evolución de la Internet actual para dar lugar a soluciones pragmáticas y viables para el despliegue comercial
 - **Revolucionario** – empieza desde cero para eliminar restricciones de diseño de la Internet actual
 - Necesario distinguir entre investigación y despliegue clean-slate
 - La investigación clean-slate debe alimentar la evolución de Internet
- Ambos enfoques están dirigidos a facilitar la visión de Future Internet y deberán ser sincronizados



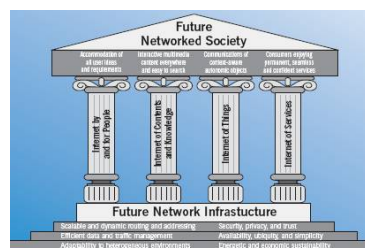
12

Los Pilares de la Internet del Futuro

- La Internet del Futuro consta de **4 pilares apoyados en una nueva infraestructura de red como base**:
 - *Internet Por y Para la Gente*
 - *Internet de los Contenidos y del Conocimiento*
 - *Internet de los Servicios*
 - *Internet de las Cosas*



Base y Pilares de la Internet del Futuro



- Tanto la base como los pilares de la Internet del Futuro son mutuamente dependientes
 - **Technology pull** → la aparición de nuevos servicios y aplicaciones supone un requisito de nueva infraestructura que los posibilite
 - **Technology push** → nuevas tecnologías e infraestructura abren nuevas oportunidades para crear nuevos servicios y aplicaciones
- **Aplicaciones de nueva generación + Mejor infraestructura de soporte = FUTURE INTERNET**

Requisitos de la Future Internet

1. Enrutamiento y direccionamiento escalable y dinámico
2. Diagnósis y gestionabilidad de recursos y tráfico de datos
3. Seguridad, privacidad, confianza y contabilidad
4. Disponibilidad, ubicuidad y simplicidad
5. Adaptación y evolución de entornos heterogéneos, contenidos, contexto y necesidades de aplicaciones (vehiculares, industriales, vida)
6. Movilidad independiente de sistema operativo, aplicación o máquina
7. Búsqueda/localización, selección, composición y adaptación
8. Más allá de la comunicación digital: semántica (inteligibilidad de cosas y contenidos), háptico, emocional, etc.
9. Sostenibilidad energética y económica



15



Desafíos Técnicos Clave

- **Internet by and for People:** acomodar las expectativas de la gente y comunidad teniendo en cuenta su nivel cultural (acceso universal a información)
- **Internet of Contents and Knowledge:** acceso mediante medios avanzados de búsqueda e interacción a contenido multimedia (3D y realidad virtual) creado, compartido y manipulado incluso por no profesionales.
- **Internet of Things:** objetos autónomos sensibles al contexto capaces de generar código y comportamiento controlado por humanos, explotando la comunicación punto a punto inspirada en el modo de comunicación de personas.
- **Internet of Services:** los “consumidores” de servicios buscan la interactividad perfecta en contexto, es decir, permanente (sin límites de interacción), directa (sobre el servicio que usa), transparente (usando distintos dispositivos) y confiable.

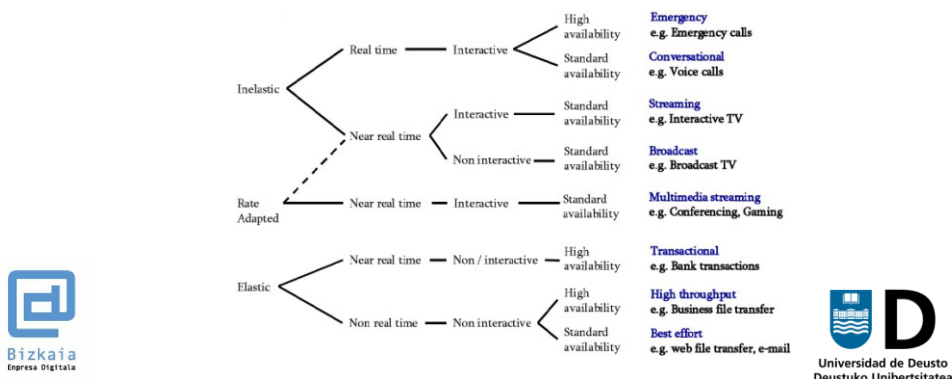


16



Infraestructura Base

- Infraestructura de red que **soporta los pilares y mantiene los requisitos de capacidad y rendimiento resultantes de la Internet del Futuro**
- Permite **usar la Internet como una infraestructura común para interconectar más que ordenadores**
 - sensor networks, dispositivos móviles, wearables, M2M, RFID and so on .



Desafíos de la Fundación de Red

- La nueva fundación de red (*Internet of Networks*) debe hacer frente a aspectos de:
 - Seguridad, privacidad y confianza
 - Contabilidad
 - Gestionabilidad y diagnóstico
 - Disponibilidad (mantenimiento y robustez)
 - Escalabilidad
 - Movilidad
 - Heterogeneidad (dispositivos, entornos, aplicaciones, redes)

Internet Por y Para la Gente

- Nueva visión del usuario en Internet que implica:
 - **Mejor experiencia de usuario:** desarrollo de nuevos interfaces ergonómicos y modelos de interacción multi-modal
 - **Usuarios activos (prosumers):** nuevas herramientas que permitan a los usuarios finales crear y compartir servicios personalizados
 - **Sensible al Usuario y Contenido:** personalización de contenido y servicios de acuerdo al contexto y preferencias personales
 - **Conocimiento de usuarios:** extraer información de los usuarios, monitorizar su comportamiento sin comprometer su privacidad



19



Desafíos de la Internet por y para la Gente

1. **Saber más del usuario** – conociendo sus hábitos y necesidades para diseñar mejor las aplicaciones, interfaces y servicios futuros
 - **Conocimiento de usuarios** – los servicios y la web deberían ser adaptables y personalizables al usuario
 - **Consciencia del usuario y del conocimiento** – para generar sistemas de recomendación, para la web móvil y los servicios sensibles a la localización
2. **Dar más poder al usuario** – más y mejores servicios personalizados y creados por los propios usuarios (enfoque *Do It Yourself – DIY*)
3. **Mejorar la experiencia de usuario** – nuevas interfaces ergonómicas y mecanismos de interacción incluyendo multi-modalidad y combinación y adaptación semántica de la información desde diferentes fuentes



20



Internet de los Contenidos y del Conocimiento

- Nueva visión del contenido en Internet que implica:
 - **Nuevos dispositivos/players:** dispositivos multimedia con más funcionalidad integrada, escalables, auto-configurables, programables
 - **Aplicaciones multimedia sociales:** software y herramientas para integrar contenidos multimedia orientados a relaciones sociales.
 - **Contenido digital:** herramientas para crear contenidos por el propio usuario y su distribución a través de diferentes localizaciones.
 - **Contenido etiquetado semánticamente:** para permitir que las aplicaciones puedan procesar información de modo inteligente
 - **Inteligencia cognitiva:** la evolución de la web a Web 3.0 permitirá motores de búsqueda multimedia avanzados.



21



Desafíos de la Internet de los Contenidos y Conocimiento

- Generación y procesamiento de contenido y su transformación en información útil, teniendo en cuenta su relación con el usuario:
 1. **Contenido digital inteligente** – información comprensible disponible al usuario, incluyendo tanto datos (esencia) como metadatos (facilitando su búsqueda, procesamiento, consumo, etc.)
 2. **Aplicaciones multimedia distribuidas sensibles al contexto** – los prosumers permitirán una selección automática de contenidos, sin importar ya tanto cómo originariamente fueron provistos los contenidos
 3. **Nuevos dispositivos de usuarios y terminales ubicuos** – los usuarios deben poder acceder a contenido allí donde están, desde cualquier terminal y sin importar los cambios de terminal que efectúen.



22



Internet de las Cosas

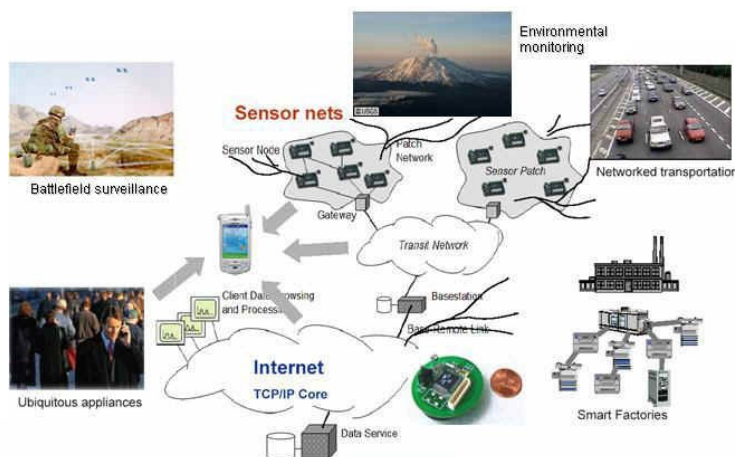
- **Red universal de objetos interconectados y direccionables basada en protocolos de comunicación estándar.**
 - IoT exhibirá un **alto nivel de heterogeneidad**, combinando objetos de distinta funcionalidad, tecnología o campos de aplicación.
 - **Protocolos semánticos** noveles serán desarrollados para permitir a IoT escalar y coordinar a los millones de objetos que nos rodean
 - **RFID y redes de sensores** proporcionan un mecanismo de bajo coste y robusto de **identificación y sensibilidad al contexto**
 - El uso de Internet pasará de **modelo request/reply** a **push-and-process**



23



Internet de las Cosas



24



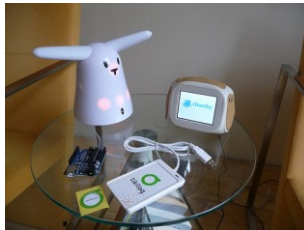
Evolución de las Aplicaciones RFID



¿Es IoT ya una Realidad?



- Trojan room coffee machine, Arduino, tiki tag, nabaztag, chumby, forecast umbrella



Desafíos de Internet de las Cosas

1. **Procesar grandes cantidades de información** de “cosas conectadas” y ofrecer servicios en respuesta
2. **Investigar en nuevos modos y mecanismos de encontrar, recuperar y transmitir datos dinámicamente**
 - **Descubrimiento** de datos sensoriales — en tiempo y espacio
 - **Comunicación** de datos sensoriales: consultas complejas (síncronas), publish/subscribe (asíncronas)
 - **Procesamiento** de gran variedad de streams de datos sensoriales: correlación, agregación y filtrado
3. **Dimensión ética y social:** mantener el equilibrio entre personalización, privacidad y seguridad



27



Internet de los Servicios



Una multitud de servicios IT conectados, que son ofrecidos, comprados, vendidos, utilizados, adaptados y compuestos por una red universal de proveedores, consumidores y agregadores de servicios o brokers - resultando en -
una nueva manera de ofrecer, utilizar, y organizar funcionalidad soportada por IT

Adapted from SAP Research, 2008, and SEEKDA, 2008



28



Internet de los Servicios

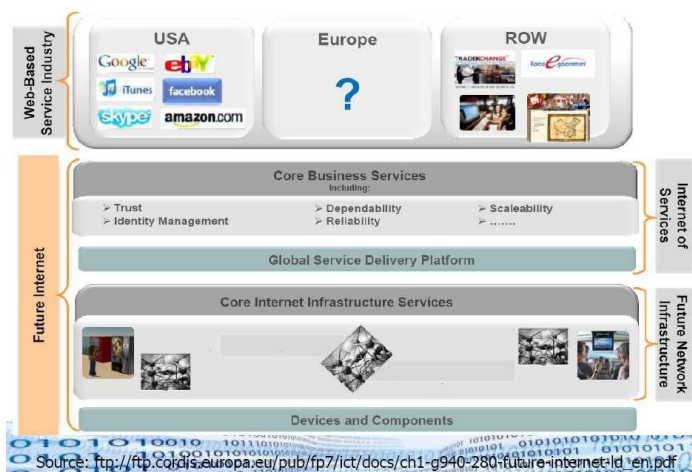
- Define cómo los servicios serán provistos y operados en la Internet del Futuro
 - http://services.future-internet.eu/index.php/Main_Page
- Tres principales áreas de investigación:
 1. **Internet-scale service oriented computing** –permite el acceso a recursos computacionales, datos o software, como servicios
 2. **Acceso a servicios contextualizado, proactivo y personalizado** – context-awareness, interacción multi-modal, end-user empowerment, colaborativo
 3. **Orquestación de servicios y servicios base** – diferentes capas de servicios, desde servicios de infraestructura a servicios específicos con interfaz de usuario.



29



Arquitectura de la Internet de los Servicios



30



Campos de Actuación de la Internet de los Servicios

- **Cloud computing:** virtualización de servicios y optimización de recursos tanto en capacidad de procesamiento como almacenamiento
- **Open service platforms:** modularidad de servicios para facilitar su integración por usuarios finales
- **Autonomic computing:** sistemas autogestionados
- **Green IT:** optimización del consumo energético



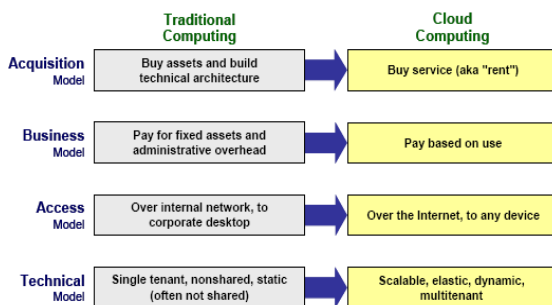
31



Infraestructura Virtualizada: Cloud Computing



Un paradigma de computación emergente donde los datos y servicios residen en centros de datos muy escalables que pueden ser accedidos ubicuamente desde cualquier dispositivo conectado a Internet¹



Source: Gartner (September 2008)

Merrill Lynch:
Cloud computing market opportunity by 2011 = \$95bn in business and productivity apps + \$65bn in online advertising = \$160bn



(1) Source: IBM



32



Desafíos de la Internet de los Servicios

1. **Permitir el acceso a servicios en base a** las características técnicas (localización IP o identificador) de servicios web, pero también en base a **información contextual** (e.j. usando contexto geográfico o de negocio).
2. **Los servicios** deben ser **buscados, identificados y compuestos en componentes de procesos de negocio**
3. **Progresar hacia la “interactividad perfecta” de los consumidores:**
 - **Permanente** – interactividad sin límites temporales
 - **Transparente** – el consumidor se concentra en los beneficios del servicio (en el qué) permitiendo la movilidad del usuario entre diferentes dispositivos sin interrupción
 - **Sensible al contexto** – la interacción se adapta al contexto incluyendo características de los dispositivos, localización, preferencias de usuarios o sus redes sociales
 - **Enriquecedor** – los usuarios configuran la manera en la que acceden a los servicios y
 - **Confiable** – los usuarios confían en que sus interacciones con servicios son seguras



33



Requisitos No Funcionales de Future Internet

- Contabilidad
- Seguridad
- Privacidad
- Disponibilidad (mantenimiento y robustez)
- Gestionabilidad y diagnosis
- Movilidad y nomadicidad
- Accesibilidad
- Apertura
- Transparencia (el usuario final/aplicación solamente conocen el servicio final)
- Neutralidad



34



Propiedades de la Internet del Futuro

- Distribuida, automatizada, y autónoma
- Escalable (escalabilidad en el direccionamiento)
- Robustez y estabilidad
- Generalidad (soportar muchos tipos de tráfico independientemente de las divisiones de infraestructura)
- Flexibilidad (soportar muchos modelos socio-económicos y operacionales)
- Simplicidad
- Evolución: capacidad de crecer y ser extendida
- Heterogeneidad (acceso cableado e inalámbrico)
- Sostenible con el entorno (green computing)



35



Proyectos del VII Programa Marco Afrontando Desafíos de FI

- Un listado detallado de proyectos en curso en el tema de Future Internet:
 - <http://www.future-internet.eu/activities/fp7-projects.html>



36



La Web del Futuro

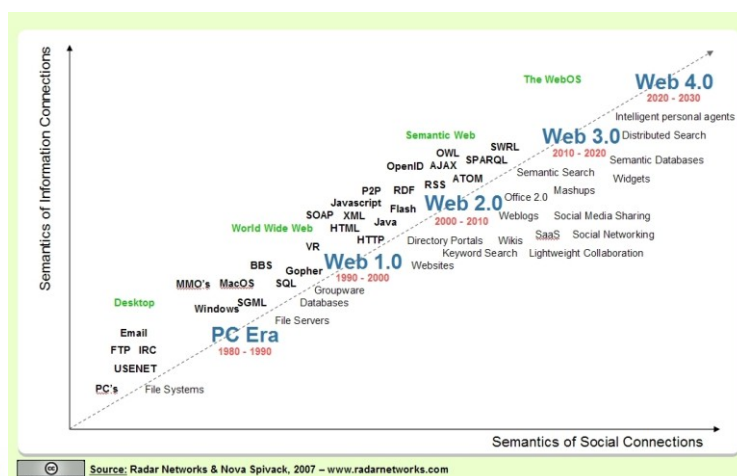
- **Web Semántica:**
 - Las máquinas entienden el contenido que han estado alojando y pueden obtener información de la semántica de los documentos
 - Tecnologías: XML, RDF, SPARQL, OWL, μ Formats...
- **Web Ubicua:**
 - Provee una framework para exportar capacidades de coordinación de dispositivos a las aplicaciones web
 - Pretende mejorar las capacidades de los navegadores para habilitar nuevos tipos de aplicaciones contextuales
 - Conectar un móvil con cámara con una impresora cercana, usar un móvil para dar una presentación en un proyector inalámbrico, etc.
- **Web Semántica + Web Ubicua = ¡Web 3.0 y más allá!**



37



La Evolución de la Web



38



La Evolución de la Web

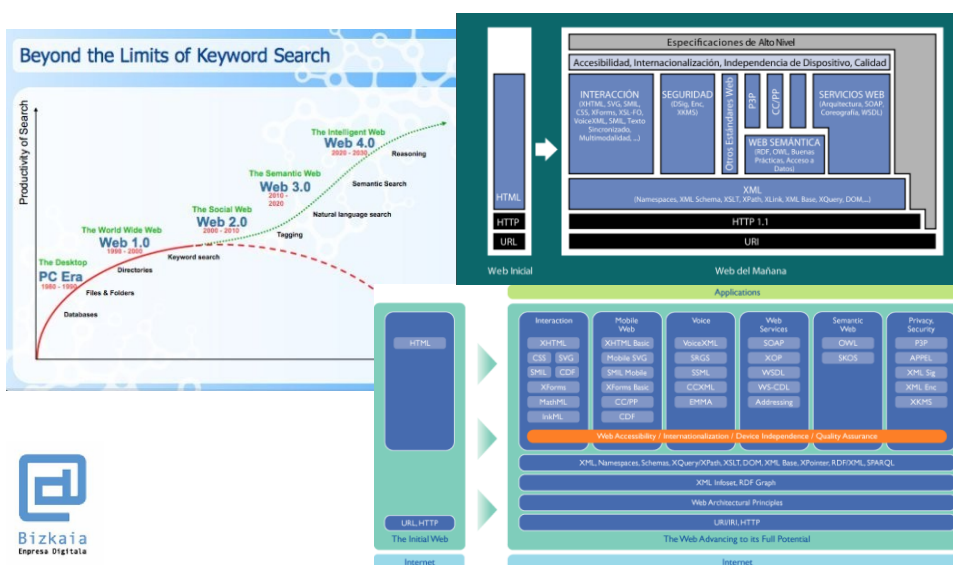
- Según Nova Spivack, el **desarrollo de la Web se mueve en ciclos de 10 años**:
 - En la **1ª década**, todo el **desarrollo centrado en la infraestructura** de la Web.
 - Los programadores crearon los **protocolos y lenguajes** que utilizamos en la Web actual
 - En la **2ª década**, el **foco se ha movido al front end y comenzó la era Web 2.0**
 - Ahora, se utilizan páginas Web como plataformas para otras aplicaciones.
 - Se crean mash-ups y experimenta con nuevos modos de hacer la web más interactiva
 - **Web 2.0: web social y web como plataforma** → estamos al final del ciclo!!
 - El **3er ciclo será la Web 3.0, y el foco será de nuevo el back-end**
 - Los programadores redefinirán la infraestructura de Internet para dar apoyo a las avanzadas capacidades de los navegadores y buscadores Web 3.0.
 - La **4ª y subsiguientes fases corresponderán a la Web 4.0 y Web X.0.**
 - Se volverá a dar importancia al front-end, nuevas y novedosas aplicaciones harán uso de la infraestructura generada en la Web 3.0
- Fuente:
 - http://www.howstuffworks.com/framed.htm?parent=web-30.htm&url=http://www.intentblog.com/archives/2007/02/nova_spivack_th.html



39



La Evolución de la Web



Definiciones de Web 3.0

*"People keep asking what Web 3.0 is. I think maybe when you've got an **overlay of scalable vector graphics** - everything rippling and folding and looking misty - on Web 2.0 **and access to a semantic Web** integrated across a huge space of data, you'll have access to an unbelievable data resource."*

Tim Berners-Lee, 2006

*"**Web 3.0**, a phrase coined by John Markoff of the New York Times in 2006, refers to a supposed third generation of Internet-based services that collectively comprise what might be called "the intelligent Web" -- such as those using semantic web, microformats, natural language search, data-mining, machine learning, recommendation agents, and artificial intelligence technologies - which emphasize **machine-facilitated understanding of information in order to provide a more productive and intuitive user experience.**"*

– http://www.intentblog.com/archives/2007/02/nova_spivack_th.html

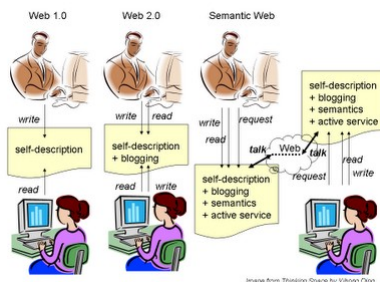


41



Concepto Web 3.0

- **Web 3.0** es una mezcla de diferentes tecnologías que cambian **cómo los usuarios interactúan con la Web y la Web interactúa con el mundo físico**.
 - Semantic Web
 - Ubiquitous Web
 - 3D Interfaces
 - Artificial Intelligence
 - ...



42



Web 3.0: Convergencia de Tendencias

- **Conectividad ubicua**
 - Adopción de broadband
 - Acceso a Internet móvil
 - Dispositivos móviles

- **Computación en red**
 - Modelo de negocio software-como-servicio
 - Interoperabilidad de servicios web
 - Computación distribuida (P2P, grid computing, granjas de servidores alojando "cloud computing" como Amazon S3)

- **Tecnologías abiertas**
 - APIs y protocolos abiertos
 - Formatos de datos abiertos
 - Plataformas software open-source
 - Datos abiertos (Creative Commons, Open Data License, etc.)



Bizkaia
Empresa Digitala

43



Universidad de Deusto
Deustuko Unibertsitatea

Web 3.0: Convergencia de Tendencias

- **Autorización e Identificación Abierta**
 - Identidad y autorización (OAuth) abierta
 - Refutación abierta
 - Identidad y datos personales portables entre diferentes servicios (OpenID)

- **La Web Inteligente**
 - Tecnologías de Web Semántica (RDF, OWL, SWRL, SPARQL, plataformas de aplicaciones semánticas, y bases de conocimiento o bases de datos basadas en sentencias (triplestores)
 - Bases de datos distribuidas – "la web como una BBDD" (interoperabilidad entre bases de datos distribuidas gracias a la Web Semántica)
 - Aplicaciones inteligentes (procesamiento del lenguaje natural, machine learning, machine reasoning, agentes autónomos)



Bizkaia
Empresa Digitala

44



Universidad de Deusto
Deustuko Unibertsitatea

Web 1.0 vs. Web 2.0 vs. Web 3.0

- **Web 1.0 es como una librería.** Puedes consultar información pero no puedes contribuir con o cambiar la información.
- **Web 2.0 es la web social colaborativa** que pone en contacto a gente y permite entrada y salida (input/output web)
- **Web 3.0 usa la Internet para conectar información.** Hace la web más inteligente, teniendo en cuenta (enchufando) nuestras preferencias, hábitos y contexto



45



Características de la Web 3.0

- **Asistente personal** que nos conoce y puede acceder a información en Internet para responder a nuestras preguntas
- Cada usuario dispondrá de un **único perfil de Internet basado en la historia de navegación del usuario**
 - Si dos usuarios realizan la misma búsqueda por Internet recibirán **resultados diferentes acordes con su perfil y contexto**
- Proveerá a los usuarios **experiencias más ricas y relevantes**
- **Su bloque funcional serán las Open APIs**
 - Crear mash-ups será tan sencillo que cualquiera podrá hacerlo
 - Se podrán mezclar widgets para hacer mashups usando drag&drop



46



Enfoque Semántico de la Web 3.0

- Algunos ven a la **Web 3.0** como una **primera manifestación de la Web Semántica**, donde ...
 - Los ordenadores escanearán e interpretarán información en páginas web usando **agentes software** buscando información relevante
 - Posible porque la Web Semántica definirá conceptos y relaciones entre los mismos mediante **ontologías**
 - Por ejemplo, el concepto “primo” hace referencia a la relación de parentesco entre dos personas que comparten abuelos



47



Evolución de los Navegadores

- **Tradicionalmente** los navegadores web se ejecutan en **ordenadores** conectados a Internet, PERO ...
 - Los móviles (y otros dispositivos) incluyen también acceso a servicios de datos mediante pequeños navegadores web
 - Las aplicaciones web deben poder ejecutarse en todo tipo de dispositivos
- Los lenguajes HTML y JavaScript no son ya sólo utilizados para acceder a la World Wide Web
 - Pueden definir **interfaces de usuario para muchos tipos de aplicaciones locales y remotas**, reduciendo el ciclo de desarrollo
- Los navegadores están emergiendo como **habilitadores de interacción global y *players* de aplicaciones web**

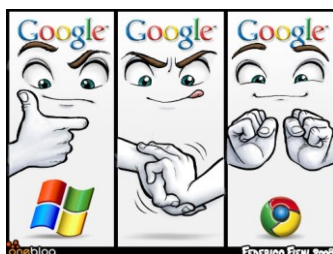


48

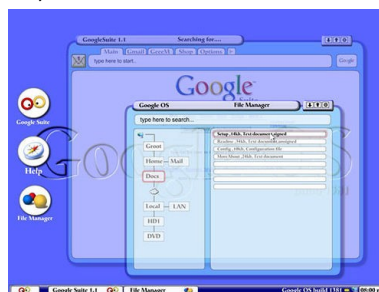


Navegadores Futuros

- ¿Cuál será el rol de los navegadores en el futuro?
 - Mozilla, Opera, Google y otros se están moviendo hacia un modelo donde el **navegador es el sistema operativo** – bien denominándolo navegador como SO o cloud SO
 - Todas las aplicaciones, documentos, ficheros y servicios accedidos a través de una interfaz web y se comportan de igual modo independientemente del SO (Windows, Linux, Android, Chrome ...)



49



Características de los Navegadores del Futuro

- Serán reproductores (players) de aplicaciones web y catalizadores de interacciones con servicios:
 - Caché de aplicaciones
 - Modo de operación offline y online
 - Capacidad de almacenamiento
 - Soporte multi-hilo
 - Integración con escritorio
 - Interacción multi-modal?
 - Interacción con el entorno?
 - Motores de ejecución de servicios?



50



El Browser como Broker de Información

- Nuevo rol de los navegadores web: de renderizador de HTML a **broker de información**
 - Detectan información en páginas web y la transfieren a otras aplicaciones
- La detección de microformatos y contenido semántico debería ser diseñado como una plataforma abierta y extensible
 - Gestión de contactos, calendario y aplicaciones de mapas usan APIs del navegador para integrarse con la capacidad de detección de microformatos y contenido semántico del browser



Mozilla Labs Prism

- Es una aplicación que permite a los usuarios sacar a aplicaciones web fuera del navegador y ejecutarlas directamente sobre el escritorio
 - Es la integración de plataforma web en la experiencia de las aplicaciones de sobremesa
 - Permite a los usuarios añadir sus aplicaciones web favoritas al desktop
 - Se ejecutan en su propia ventana, incrementando sus capacidades con almacenamiento de datos offline y accediendo a hardware de gráficos 3D
- Referencia: <http://labs.mozilla.com/2007/10/prism>



Mozilla Ubiquity



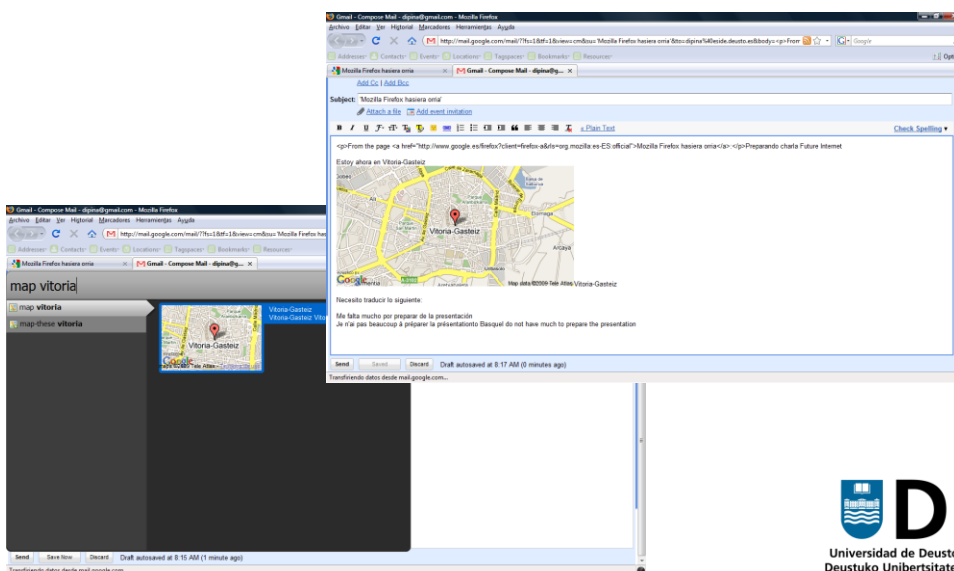
- **Un experimento conectando la web con el lenguaje**
 - Mediante Ubiquity **puedes decir a Firefox qué quieres hacer escribiendo comandos** en una caja de entrada de datos
 - Ubiquity ofrece un conjunto de comandos que hacen a las tareas web comunes más fáciles y rápidas
 - Te **permite crear mash-ups generados por el usuario**
- Se pueden ejecutar comandos directamente o seleccionado previamente fragmentos de texto:
 - Ctrl+Space → muestra la ventana de comandos en Firefox
 - wiki deusto → busca Deusto in Wikipedia
 - translate (text) (from language) (to language)
 - map [location | selected text]
 - command-list → te permite pedir ayuda de todos los comandos



53

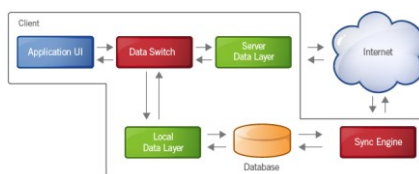


Mozilla Ubiquity



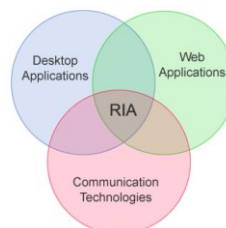
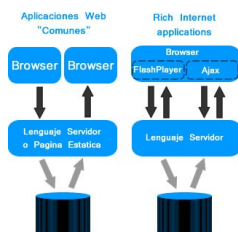
Google Gears

- **Gears**, es un software que permite el desarrollo de aplicaciones web más poderosas, añadiendo las siguientes capacidades al navegador:
 - Un **módulo de BBDD**, basado en SQLite, que permite guardar datos en el navegador
 - Una **librería de multithreading** (WorkerPool) para ejecutar en paralelo código JavaScript
 - Un **módulo llamado LocalServer que cachea recursos obtenidos previamente** (HTML, JavaScript, imágenes, etc).
 - Un **módulo desktop** que permite a las aplicaciones interactuar con la máquina local
 - Un **módulo de geo-localización** que permite a las aplicaciones web detectar la localización geográfica de sus usuarios
 - <http://gears.google.com/>
 - <http://code.google.com/intl/es-ES/apis/gears/sample.html>



Rich Internet Applications (RIA)

- Una tendencia industrial actual que mejora la experiencia del usuario cuando consume servicios SOA
 - Combinan la rapidez de respuesta e interactividad de las aplicaciones de sobremesa con el amplio despliegue y facilidad de uso de las aplicaciones web



56



Características RIA

- Son **aplicaciones web con funcionalidad similar a las aplicaciones de sobremesa** tradicionales
- Son aplicaciones cliente que **mantienen estado y están desacopladas de la capa de servicios en el back-end**
- Requieren el siguiente **entorno de ejecución**:
 - Son ejecutadas en un **navegador web**, no requieren instalación
 - Son ejecutadas en un entorno seguro llamado sandbox
- Ventajas:
 - Permiten al usuario encontrar información más fácilmente
 - Permiten la compleción de tareas más rápidamente de una manera precisa
 - Generan visualizaciones de datos ricas que pueden ayudar en el proceso de toma de decisiones

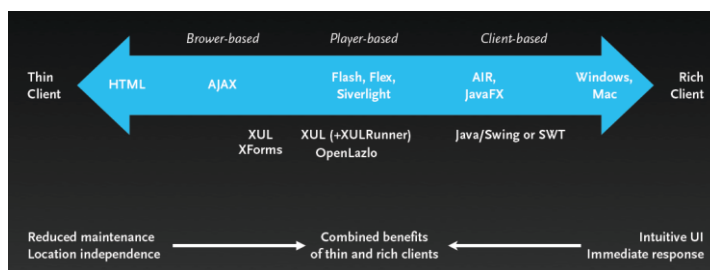


57



Taxonomía de las Aplicaciones RIA

- El *modo declarativo* de las nuevas herramientas RIA cambia el modelo para desarrollar software de un modelo **CÓMO** a un modelo **QUÉ**
- Tres tipos:
 - **Client-based** – aplicaciones basadas en una aplicación de sobremesa
 - **Player-based** – aplicaciones basadas en una extensión del navegador
 - **Browser-based** – aplicaciones basadas en una framework JavaScript



58



Demos RIA

- jQuery:
 - <http://www.webdesignerwall.com/demo/jquery/>
 - http://docs.jquery.com/Tutorials:Live_Examples_of_jQuery
- XUL
 - <http://www.hevanet.com/acorbin/xul/top.xul>
- Flex
 - <http://flex.org/showcase/>
- Silverlight
 - <http://silverlight.net/Showcase/>



59



Una Nueva Generación: las Aplicaciones Web

- La nueva generación de aplicaciones web está tomando forma con tecnologías como jQuery, AIR, Silverlight, JavaFX, Gears, XUL, Web Applications 1.0 (HTML 5.0)
 - Permiten a los desarrolladores ir más allá de AJAX **hacia una nueva generación de aplicaciones web con mejor rendimiento, más funcionalidad y mejor integración con el escritorio**
 - ¡CUIDADO! Podemos pasar del “DLL hell” al “plug-in hell”
 - Para evitarlo deberíamos adoptar un enfoque basados en estándares
 - » <http://www.techcrunch.com/2008/05/29/the-next-gen-web-browser-storage-support/>



60



HTML 5.0

- El Web Hypertext Application Technology Working Group (WHATWG) ha dado lugar a la especificación de HTML 5.0 (www.w3.org/html/wg/html5/):
 - “**The main area that has not been adequately addressed by HTML is Web Applications.** This specification attempts to rectify this, while at the same time updating the HTML specs to address issues raised in the past few years”
 - **El enfoque predominante de WebApps basado en plug-ins debe quedar obsoleto**
- HTML 5 proporciona un conjunto de nuevos elementos y atributos que reflejan el uso típico en portales Web modernos:
 - **Reemplazos semánticos para usos de elementos genéricos** como <div> o elements, por ejemplo <nav> (bloque de navegación de website) y <footer>.
 - Otros elementos proveen **nueva funcionalidad a través de una interfaz estándar**, como los elementos <audio> o <video>
 - **Elimina elementos de HTML 4.** Por ejemplo, elementos presentacionales como y <center>, cuyos efectos son mejor logrados mediante CSS
 - Hay un nuevo énfasis en la **importancia de scripting DOM para comportamiento Web**



61



Características de HTML 5.0

- Introduce nuevos elementos de estructura como: header, nav, article, section, aside, y footer.
 - aside – ofrece soporte para barras laterales, datagrid, dialog, figure, Web Forms 2.0
- **El gran potencial de HTML 5 vendrá de sus nuevas APIs para la autoría de aplicaciones basadas en la Web :**
 - El interfaz WebSocket permitirá crear **sockets TCP/IP bidireccionales** entre la aplicación y el servidor
 - El servicio Remote-Events permite a los **servidores enviar eventos** document-objet-model (DOM) a aplicaciones
 - Nueva **API de almacenamiento local** que permitirá a los agentes de usuario guardar y recuperar más información de la que podría crearse como un conjunto de pares nombre-valor como en las cookies.
- HTML 5.0 incluye **soporte explícito para la ejecución offline de aplicaciones web**: la caché de aplicaciones y application cache manifest
- Nuevas **características multimedia** como gráficos 2D y APIs de control de audio y video
- Especificaciones de **webworker** (multithreading) y **geolocation**



62



Despliegue de HTML 5.0

- Pasarán varios años antes de que encontremos soporte para HTML 5.0 en los navegadores
- Entretanto:
 - Opera y Webkit están implementando activamente partes de HTML 5.0
 - Google Gears (<http://code.google.com/apis/gears>) proporciona muchas de las características de HTML 5 a tu navegador
 - APIs para geolocation
 - Almacenamiento en la parte cliente
 - Operación offline
 - Web workers
 - Google contribuye a las especificaciones relacionadas en el W3C
- La **combinación HTML+JavaScript** se está configurando como “**la plataforma de aplicaciones web**” para el front-end



63



RIA para Móviles



- La popularidad de iPhone y Android implican una gran necesidad en plataformas de aplicaciones móviles avanzadas
- Los navegadores web móviles están mejorando mucho:
 - Los navegadores de Nokia, Apple, Opera y Microsoft pueden gestionar cualquier contenido
- Tecnologías de Web 2.0 como XMLHttpRequest o Flash que han creado potentes aplicaciones web deben ser portadas a los dispositivos móviles
 - Webkit (<http://webkit.org>), Opera, Mozilla e Internet Explorer ya dan soporte para la ejecución de JavaScript con un alto rendimiento
 - Nokia Web Run Time: basado en widgets personalizados y sensibles al contexto que acceden a información en Internet



64





El Futuro del Navegador: Opera Unite

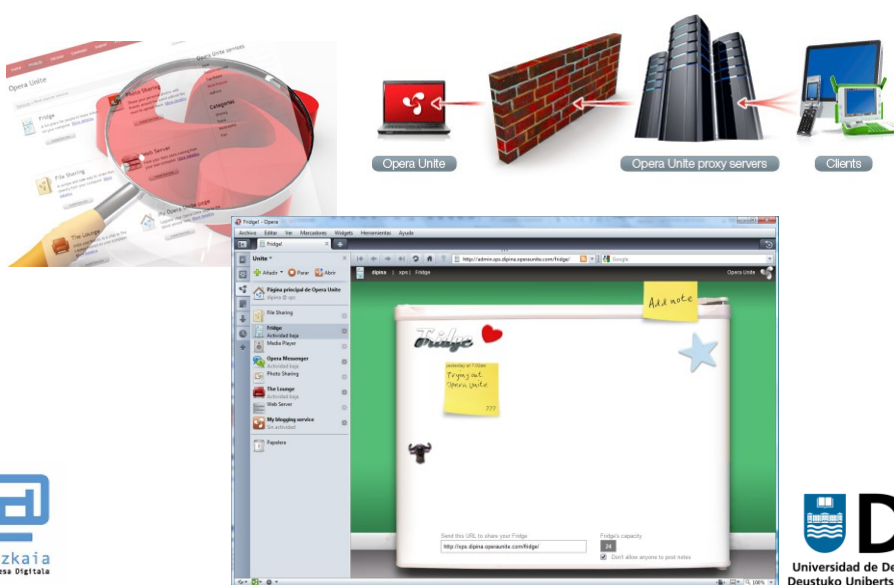
- Un servidor web ejecutándose en un navegador Opera
 - Tecnología cooperativa que utiliza un servidor web compacto dentro del navegador Opera para compartir datos y servicios
 - Aloja **Opera Unite Services** que proporcionan desde tu PC contenido a usuarios remotos sin necesidad de aplicaciones o servidores terceros
 - Imágenes, documentos, vídeos, aplicaciones colaborativas, etc.
 - Utiliza un proxy (<http://operaunite.com>) para evitar cortafuegos
 - Realmente intenta enviar datos directamente mediante NAT transversal
- Referencias:
 - <http://unite.opera.com/>
 - <http://dev.opera.com/articles/view/opera-unite-developer-primer/>
 - <http://dev.opera.com/libraries/unite/>



65



Opera Unite Services





El Futuro de la Mensajería: Google Wave

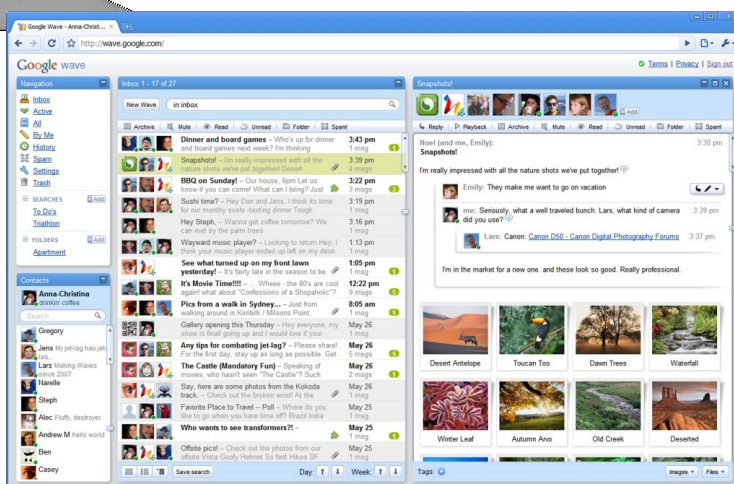
- **Google Wave** es una herramienta de comunicación y colaboración entre personas
 - Es un servicio web, plataforma de computación y protocolo de comunicación diseñado para combinar en uno email, mensajería instantánea, wikis y redes sociales
 - Está concebido para ser la siguiente generación en comunicación por Internet
- Un “wave” es un hilo completo de mensajes multimedia (blips) localizados en un servidor central
- Referencia:
 - <http://wave.google.com/>



67



Google Wave



68



El Futuro de las Búsquedas

- Últimamente han aparecido varios servicios similares a buscadores interesantes:
 - Wolfram|Alpha
 - Microsoft Bing and
 - Google Squared
 - Rich Snippets & SearchMonkey



69



Wolfram|Alpha

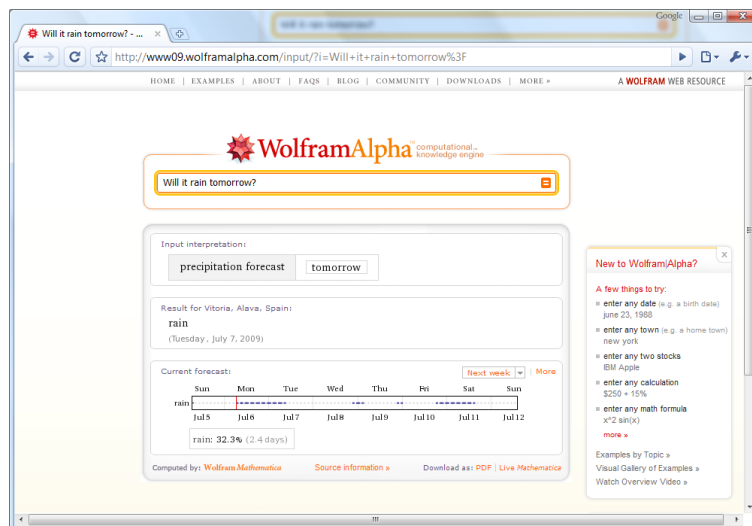
- Es un **motor de conocimiento** que genera salidas realizando cálculos a partir de su propia base de conocimiento, en vez de hacer búsquedas en la web y devolver vínculos.
 - Contiene algoritmos que permiten obtener datos sobre hechos y fórmulas planteados en lenguaje natural, apoyándose en la infraestructura de Mathematica
 - Los resultados vienen como gráficos, mapas, hechos y tablas
- **¿Para qué sirve Wolfram|Alpha?**
 - Para mostrar información o realizar computaciones sobre hechos
 - El servicio conoce mucho del mundo:
 - Geografía, economía y datos socioeconómicos, física, química, ingeniería, deportes, unidades de medida, tiempo e incluso música.
- Diferencia con los buscadores:
 - Todos los hechos, fórmulas y relaciones han sido preparados por usuarios humanos
 - Wolfram|Alpha **intenta determinar el significado real de una pregunta realizada**
 - Procesadores de lenguaje natural determinan el dominio de la pregunta para así facilitar el cálculo de la respuesta



70



Wolfram | Alpha



71



Microsoft Bing

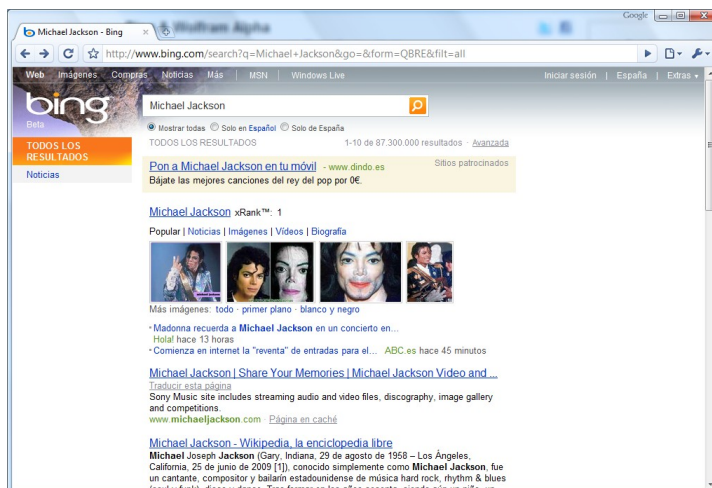
- Bing es un reemplazo de Live Search, **nuevo motor de búsqueda de Microsoft**
 - Ayuda a los usuarios en ciertos dominios de búsqueda de gran interés: compras, viajes, información de negocios locales, salud
- Algunas de sus mejoras frente a la competencia:
 - Listado de sugerencias de búsqueda en tiempo real
 - Listado de refinamientos de búsqueda en la parte izquierda
 - La parte izquierda también sugiere “búsquedas relacionadas”
 - La interfaz tiene bastantes mejoras – pequeños thumbnails de videos empezarán a reproducirse al mover el ratón sobre ellos, ...
- Probar:
 - <http://www.bing.com?cc=us>
 - <http://www.bingandgoogle.com>



72



Microsoft Bing



Google Squared

- “Squared” se refiere a la facilidad para construir tablas de hechos a partir de dos conceptos de búsqueda
 - En un eje se puede usar un término de búsqueda general (montañas rusas o “roller coasters”)
 - En el otro eje se pueden añadir cabeceras que representan hechos sobre el término inicial (altura, velocidad)
 - Resultando en una tabla bidimensional de resultados
- Comparado con Wolfram|Alpha, no hay intermediación humana, la información viene de la Web
- <http://www.google.com/squared>

Using Google Squared

The screenshot shows a Google Squared search for 'roller coaster'. The results are displayed in a table with columns for Item Name, Image, Description, Length, Height, and Top Speed. The table lists several roller coasters with their respective specifications and descriptions.

Item Name	Image	Description	Length	Height	Top Speed
Racer		The Racer at Kings Island is a wonderful racing out and back wooden roller coaster designed by John Allen. The trains race at speeds of up to 61 mph as they ...	3400 feet	90 feet	220 km/h
Cyclone		For other roller coasters named Cyclone, see Cyclone (disambiguation). ... The Coney Island Cyclone is an ACE Coaster Classic and Coaster Landmark. ...	2640 feet	85 feet	60 mph
Batman The Ride		Batman: The Ride is a steel inverted roller coaster found in many Six Flags theme parks, as well as other parks around the world, including Six Flags Great ...	2,700'	100'	50 mph
Colossus		Six Flags Magic Mountain Colossus Wooden Roller Coaster POV. Added. 3:19. Six Flags Magic Mountain Colossus Wooden Roller... 6601 views ...	2789 feet	125'	62 mph
Texas Giant		At more than 14 stories tall, the Texas Giant is one of the tallest, fastest wooden roller coasters to be found anywhere. The Giant was named the #1 roller ...	4,920 ft	143 ft	62 mph
Revolution		CGA Revolutions, the fourth segment or "project" of the Cyber Girls Akhabara anime. In Bru Revolution, a roller coaster at Blacktop! Pleasure Beach, ...	3457 feet	113'	55 mph
Scoby Doo					

La Web como Plataforma: Parte Cliente

- Tecnologías como AIR, XUL, BrowserPlus o Gears comparten la misión de permitir una “nueva generación de aplicaciones web con mejor rendimiento, más funcionalidad y mejor integración con el escritorio”
 - XUL, Flex, Silverlight permiten crear aplicaciones RIA avanzadas
 - AIR permite hacer que aplicaciones web se ejecuten de manera offline
 - BrowserPlus se ejecuta fuera del navegador para hacer a tu escritorio una extensión de la web
 - Gears se ejecuta dentro del navegador mejorando la experiencia de ejecución de aplicaciones web
- Con el tiempo, conceptos de estos productos se volverán estándar, mientras tanto “graceful degradation”
 - <http://www.whatwg.org/specs/web-apps/2005-09-01/>

La Web como Plataforma: Parte Servidora

SOA: Service Oriented Architecture

- Define la utilización de **servicios** para dar soporte a los **requisitos del negocio**
- Sistemas altamente **escalables**, estándar de exposición e invocación de servicios (comúnmente pero no exclusivamente servicios web), lo cual **facilita la interacción entre diferentes sistemas** propios o de terceros.
- Éxito de SOA depende de que los desarrolladores de software se orienten a esta mentalidad de crear **servicios comunes** que son orquestados por clientes o middleware para implementar los procesos de negocio
- Hablar de SOA significa generalmente hablar de un juego de servicios residentes en Internet o una intranet, usando tecnologías como (XML, HTTP, SOAP, WSDL, UDDI)



WOA: Web Oriented Architecture

- Arquitectura de software que extiende SOA a aplicaciones basadas en web
- La información se representa en forma de **recursos** en la red y es accedida y manipulada mediante el protocolo especificado en la URI.
- Los recursos se manipulan mediante CRUD HTTP (POST, GET, PUT, DELETE) usando **REST**.
- Estos recursos sólo son manipulados por componentes pertenecientes a la red (esencialmente browsers & web servers).
- Es responsabilidad de los componentes entender la representación y estados de transición válidos de los recursos.
- Los recursos tienen embebidas URIs que construyen una red más grande de recursos



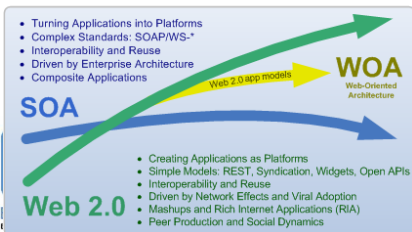
77

SOA vs. WOA

The SOA Core with Reach: Web-Oriented Architecture

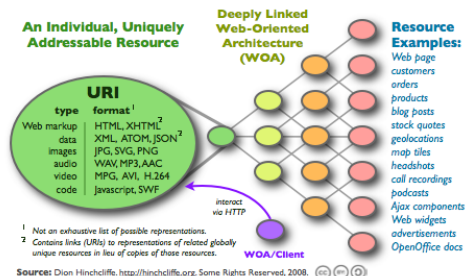


The High Levels of Success of Web 2.0 Models for Creating Software Ecosystems Helped "Discover" WOA and Inform SOA



78

SOA Reshaped by the Web 2.0 Era: Granular, Radically Distributed, Web-Oriented, Open, Highly Consumable



La Web como Plataforma: Parte Servidora

SOAP: Simple Object Access Protocol

- Define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos **XML**.
 - Deriva de un protocolo creado por David Winer en 1998, llamado **XML-RPC**.
- Fue creado por **Microsoft, IBM**, y otros y actualmente se encuentra bajo el auspicio de la **W3C**.
- Su arquitectura consiste en varias capas de especificaciones para formato de mensajes:
 - Message Exchange Patterns (MEP)
 - Protocolos de transporte (SMTP y HTTP/S)
 - Modelos de procesamiento de mensajes
 - Protocolo de extensibilidad



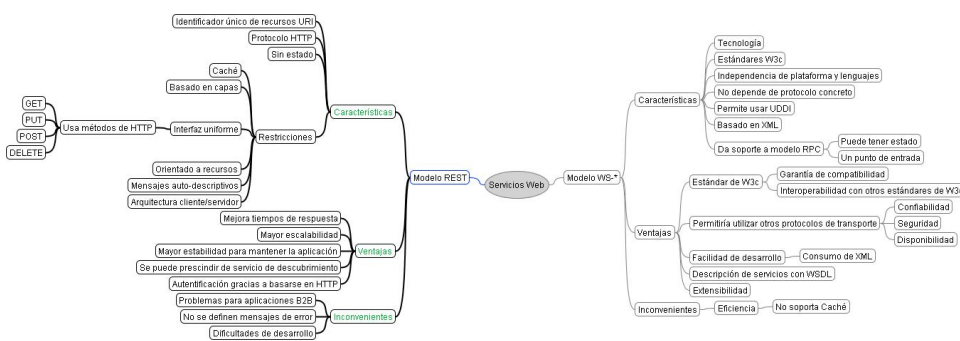
REST: Representational State Transfer

- Estilo** de arquitectura software para sistemas de **hypermedia** distribuidos como la **WWW**.
 - Introducido en la tesis doctoral de **Roy Fielding** en el año 2000.
- Se refiere a una colección de principios de arquitectura de red, que marcan cómo definir e invocar los recursos.
- El término se usa a veces para describir una simple interfaz que transmite datos de un dominio específico por **HTTP** sin capas adicionales como SOAP o uso de cookies.
- Los sistemas que cumplen los principios marcados por Fielding suelen ser referidos como sistemas **RESTful**.

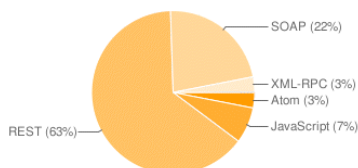


79

SOAP vs. REST



Protocol Usage by APIs

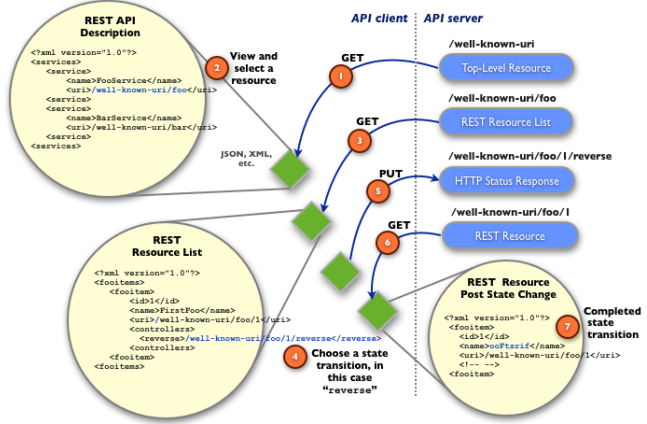


Usando REST

REST API Usage Scenario

Key Aspects Depicted

- A single well known, top level resource describes the entire API and URIs into the entire resource graph.
- Allowable state changes are documented by the resource via URIs.
- State is changed by accessing the URI via POST. GET is kept idempotent.
- The interaction pattern forms HATEOS (hypermedia as the engine of application state.)

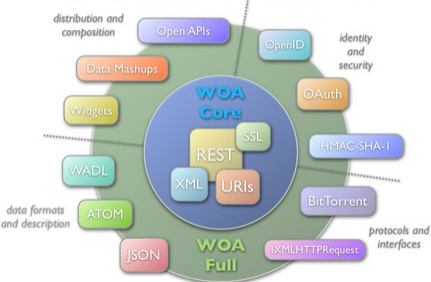


Source: Dion Hinchcliff, 2009. <http://hinchcliff.org>



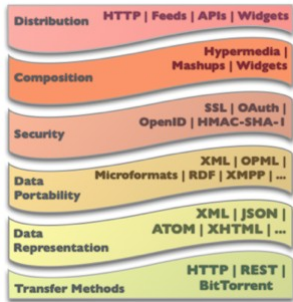
Web Oriented Architecture (WOA)

The Elements of Web-Oriented Architecture



Source: Dion Hinchcliff, 2009. <http://hinchcliff.org>

The WOA Stack



Source: Dion Hinchcliff, 2009. <http://hinchcliff.org>



Servidores Móviles

- Una tendencia actual es tener dispositivos móviles sirviendo aplicaciones:

- **Nokia Mobile Web Server**

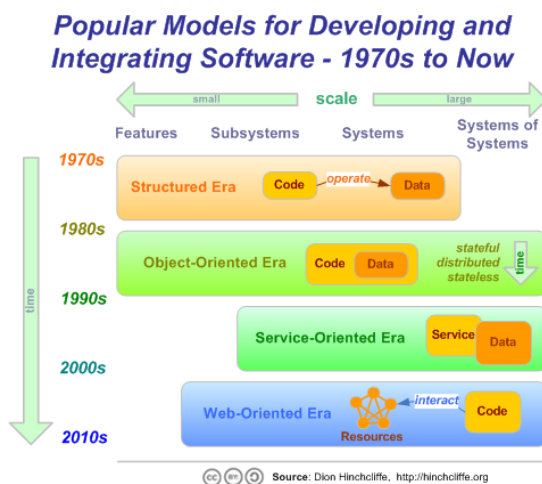
- Mapeo de apache httpd y mod_python a Nokia S60
- Provee pasarela para prevenir problemas con cortafuegos
- <http://mymobilesite.net/>



83



El Futuro del Desarrollo Software



84



Cloud Computing es ...

- ... **capacidad computacional y almacenamiento virtualizada** expuesta mediante **infraestructura agnóstica a la plataforma y accedida por Internet**
 - **Recursos IT compartidos en demanda, creados y eliminados eficientemente y de modo escalable** a través de una variedad de interfaces programáticos **facturados en base a su uso**

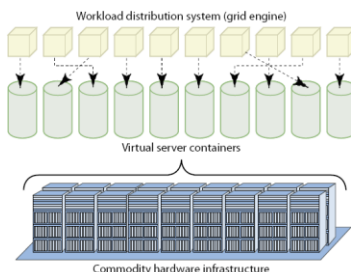


85



Forrester Research

“A pool of **abstracted, highly scalable, and managed compute infrastructure** capable of hosting end-customer applications and **billed by consumption**¹”



1- "Is Cloud Computing Ready for The Enterprise?" Forrester Research, Inc.

86



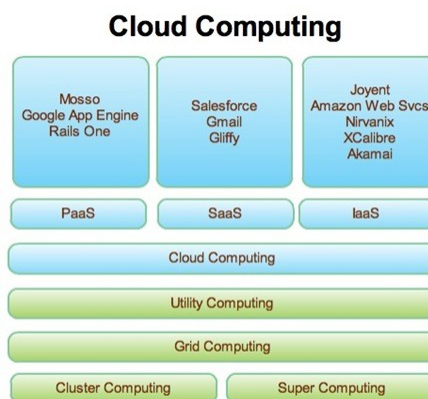
The “Cloud” = 10X Improvement

- **Fácil de usar:** hazlo tu mismo remotamente de cualquier lugar en cualquier momento
- **Escalable:** controla tu infraestructura con tu aplicación
- **Riesgo:** nada que comprar, cancela inmediatamente
- **Robustez:** basado en gran hardware empresarial
- **Coste:** paga sólo por lo que uses



Evolución hacia Cloud Computing

- La coexistencia y limitaciones de cluster computing y supercomputing dieron lugar a grid computing
- De grid computing progresamos hacia **utility computing**, i.e. Servicios computacionales empaquetados como agua, electricidad, etc.
- Esto derivó en Cloud Computing, es decir, **todo como servicio (XaaS)** :
 - *Plataforma como Servicio*
 - *Software como Servicio*
 - *Infraestructura como Servicio*



1.0 In blue you have what is lately called Cloud Computing. In green, some of the underlying work done that led to Cloud Computing. At the top are examples of each XaaS type.

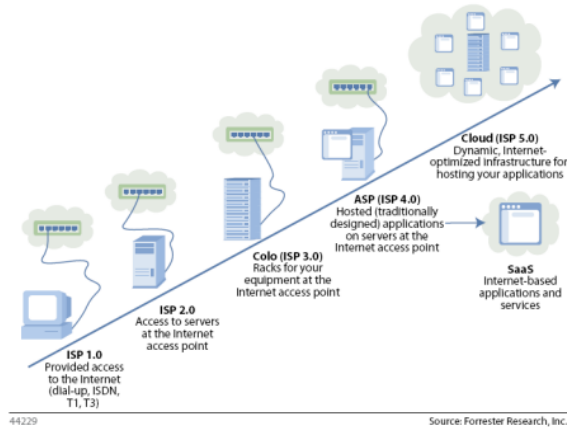


88

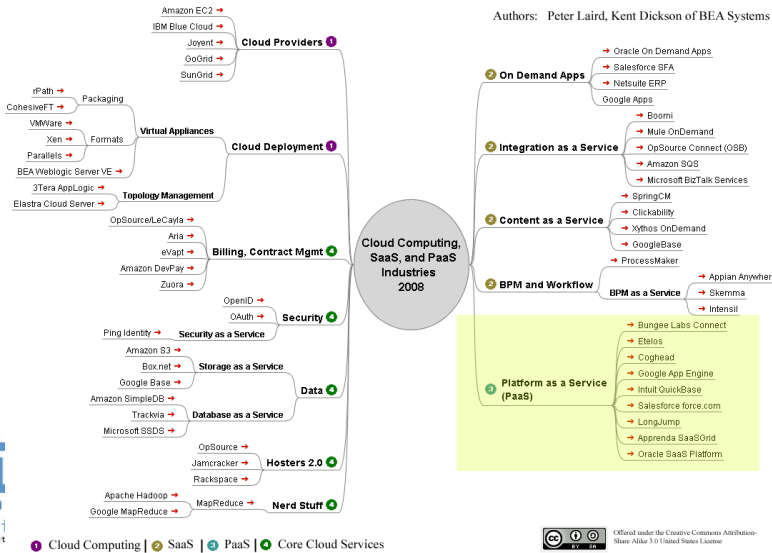


Múltiples Descripciones Gráficas de la "Cloud"

Figure 3 Cloud Computing: The Latest Evolution Of Hosting



Múltiples Descripciones Gráficas de la "Cloud"



Características de Cloud

Tipos de despliegue

- **Cloud privada**
 - Propiedad de o alquilada por una empresa (centros de datos,...)
- **Cloud comunitaria**
 - Infraestructura compartida por una comunidad específica
- **Cloud pública**
 - Vendita al público, gran escala (ec2, S3,...)
- **Cloud híbrida**
 - Composición de dos o más clouds



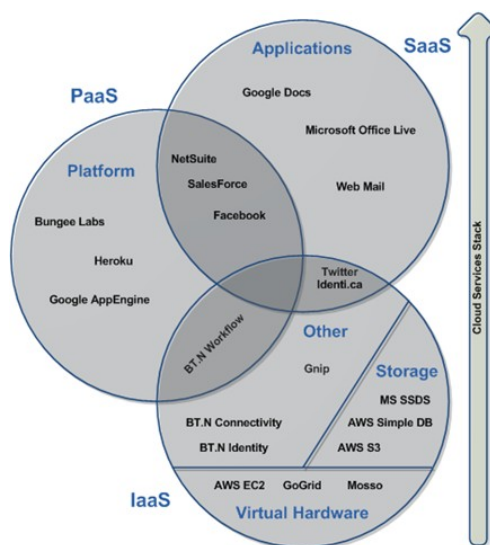
Manifestaciones

- **Cloud Software as a Service (SaaS)**
 - Uso de la aplicación del proveedor sobre la red, e.j., Salesforce.com,...
- **Cloud Platform as a Service (PaaS)**
 - Despliega aplicaciones creadas por los clientes a la nube, e.j. Google App Engine, Microsoft Azure, ...
- **Cloud Infrastructure as a Service (IaaS)**
 - Alquilar procesamiento, almacenamiento, capacidad de red y otros recursos computacionales e.j., EC2 – Elastic Computer Cloud, S3 – Simple Storage Service, Simple DB,...

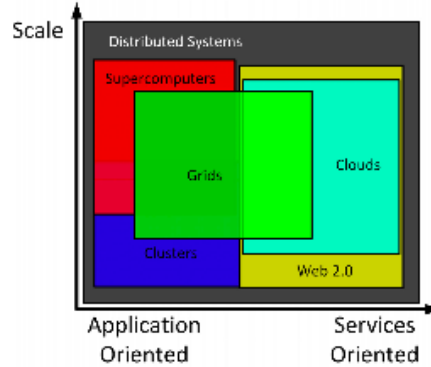
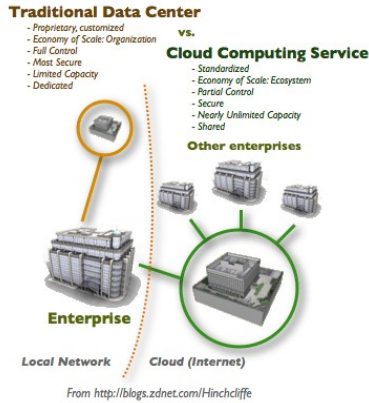
91



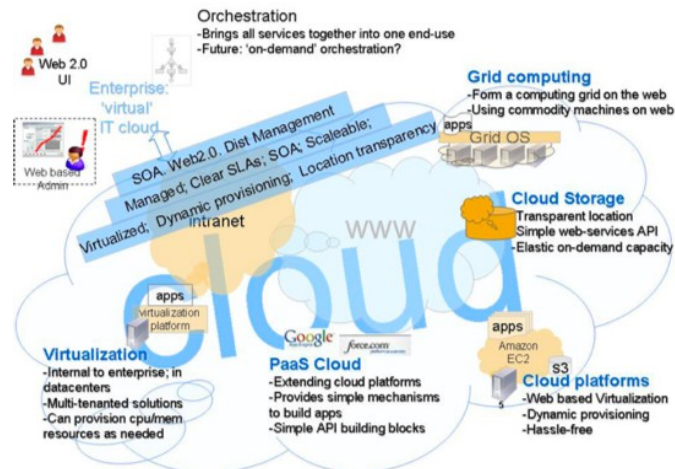
Diferentes Manifestaciones



Cloud Computing vs. Centros de Datos Tradicionales



Componentes de Cloud Computing



Efecto Bola de Nieve

- Maduración de tecnología de virtualización
- La virtualización permite nubes de computación
- Las nubes de computación demandan nubes de almacenamiento
- Las nubes de almacenamiento y computación crean infraestructura cloud
- La infraestructura cloud da lugar a plataformas y aplicaciones cloud
- Diferentes tipos de cloud dan lugar a Cloud Aggregators
- Nichos de requisitos dan lugar a Cloud Extenders



95



SaaS: Software as a Service

- Modelo de envío de software:
 - Incrementalmente popular con PYMEs
 - No hay hardware o software a gestionar
 - Servicio enviado a través de un navegador
- Ventajas:
 - Pago por uso
 - Escalabilidad instantánea
 - Seguridad
 - Robustez
 - APIs
- Ejemplos: CRM, Financial Planning, Human Resources, procesamiento de texto
 - Servicios comerciales: Salesforce.com, emailcloud, Google Apps

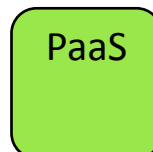


96



PaaS: Platform as a Service

- Modelo de envío de plataforma:
 - Las plataformas son construidas encima de infraestructura que es cara
 - ¡La estimación de demanda no es una ciencia!
 - La gestión de la plataforma es costosa
- Ventajas:
 - Pago por uso
 - Escalabilidad instantánea
 - Seguridad
 - Robustez
 - APIs
- Ejemplos: Google App Engine, Mosso, AWS: S3
 - Servicios populares: Almacenamiento, Bases de datos, Escalabilidad



97



IaaS: Infrastructure as a Service

- Modo de envío de infraestructura computacional:
 - Acceso a la pila de infraestructura:
 - Acceso completo al SO
 - Cortafuegos
 - Routers
 - Balanceo de carga
- Ventajas:
 - Pago por uso
 - Escalabilidad instantánea
 - Seguridad
 - Robustez
 - APIs



Ejemplos: Flexiscale, AWS: EC2

98



Aplicaciones Cloud

- SaaS reside aquí
- Manifestación de cloud más popular
- Ejemplos: Examples: Salesforce, Gmail, Yahoo! Mail, Quicken Online, Google Docs
- Ventajas: Libre, Fácil, Adopción de consumo
- Desventajas: funcionalidad limitada, no hay control o acceso a la tecnología subyacente



99

Plataformas Cloud

- Contenedores de aplicaciones
- Entornos cerrados
- Ejemplos: Google App Engine, Heroku, Mosso, Engine Yard, Joyent o Force.com (SalesForce Dev Platform)
- Ventajas: buenas para desarrolladores, más control que en las aplicaciones cloud, configuradas estrechamente
- Desventajas: restringidas a lo que está disponible, otras dependencias



100

Infraestructura Cloud

- Proveen nubes de computación y almacenamiento
- Ofrecen capas de virtualización (hardware/software)
- Ejemplos: Amazon EC2, GoGrid, Amazon S3, Nirvanix, Linode
- Ventajas: control completo del entorno y la infraestructura
- Desventajas: precio premium, competencia limitada



101



Extensores de la Cloud

- Proveen extensiones a infraestructura y plataformas cloud con funcionalidad básica
- Ejemplos: Amazon SimpleDB, Amazon SQS, Google BigTable
- Ventajas: extienden la funcionalidad de las nubes de computación y almacenamiento para integrar sistemas heredados u otras cloud
- Desventajas: a veces requieren el uso de plataformas o infraestructura específica



102



Agregadores Cloud

- Se apoyan en varias infraestructuras cloud para su gestión
- Ejemplos: RightScale, Appistry
- Ventajas: proveen más opciones para entornos cloud
- Desventajas: dependientes de proveedores de cloud





103



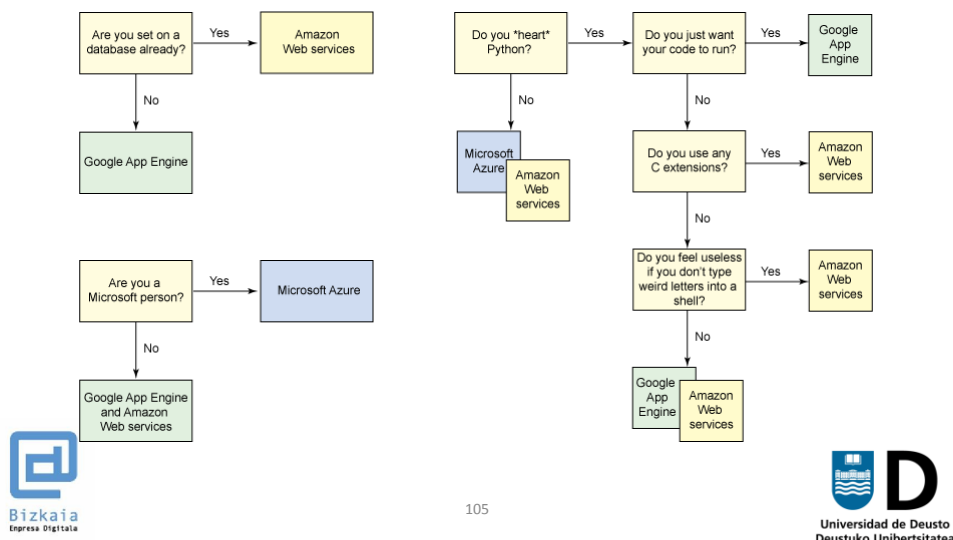
Plataformas Cloud Existentes

	Amazon Elastic Compute Cloud (EC2)	Google App Engine	Microsoft Azure	Sun Network.com (Sun Grid)	Grid Lab Aneka
Focus	Infrastructure	Platform	Platform	Infrastructure	Software Platform for enterprise Clouds
Service Type	Computer, Storage (Amazon S3)	Web Application	Computer, Storage	Compute	Compute
Virtualisation	OS Level running on a Xen hypervisor	Application container	Windows Azure OS	Job management system (Sun Grid Engine)	Resource Manager and Scheduler
Dynamic Negotiation of QoS Parametres	None	None	None	None	SLA-based Resource Reservation on Aneka side
User Acces Interface	Amazon EC2 Command-line Tools	Web-based Administration Console	Web-based Live Desktop and any devices	Job submission scripts, Sun Grid Web portal	Workbench, Web-based portal
Web APIs	Yes	Yes	Yes	Yes	Yes
Value-added Service Providers	Yes	No	No	Yes	No
Programming Framework	Customizable Linux-based Amazon Machine Image (AMI)	Python	.Net Framework	Solaris OS, Java, C, C++, FORTRAN	APIs supporting different programming models in C# and other .Net lengauges

104



¿Cuál utilizar?



Ejemplo Plataforma Cloud: Google App Engine

- Google App Engine es una herramienta para el alojamiento de aplicaciones web escalables sobre la infraestructura de Google
 - Su misión es permitir al desarrollador web crear fácilmente aplicaciones web escalables sin ser un experto en sistemas
- Aporta las siguientes características a los desarrolladores:
 - **Limita la responsabilidad del programador al desarrollo y primer despliegue**
 - Google App Engine provee recursos computacionales dinámicamente según son necesarios
 - **Toma control de los picos de tráfico** – si nuestro portal crece en popularidad no es necesario actualizar nuestra infraestructura (servidores, BBDD)
 - Ofrece replicación y balanceo de carga automática apoyado en componentes como Bigtable
 - **Fácilmente integrable con otros servicios de Google** – los desarrolladores pueden hacer uso de componentes existentes y la librería de APIs de Google (email, autenticación, pagos, etc.)



Google App Engine: Características

- Ofrece una plataforma completa para el alojamiento y escalado automático de aplicaciones, consistiendo en:
 - Servidores de aplicaciones Python y ahora Java
 - La base de datos BigTable
 - El sistema de ficheros GFS
- Como desarrollador simplemente tienes que subir tu código Python a Google, lanzar la aplicación y monitorizar el uso y otras métricas
- Google App Engine incluye la librería estándar de Python 2.5
 - No todas las acciones se permiten (acceso a ficheros, llamadas al SO, algunas llamadas de red)
 - Ejecuta en un entorno restringido para permitir que las aplicaciones escalen
- Ejemplo:
 - <https://appengine.google.com>
 - <http://enpresadigitala.appspot.com/encuestas>
 - <http://code.google.com/intl/en/appengine/docs/>

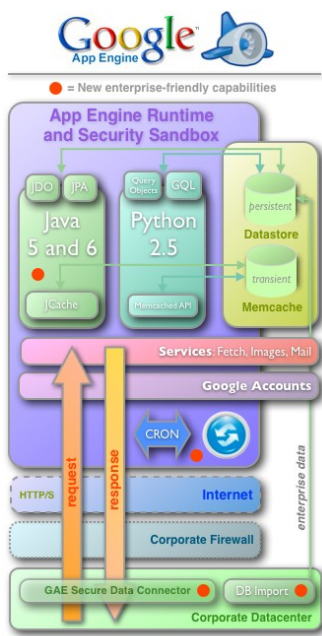


Bizkaia
Empresa Digitala

107



Universidad de Deusto
Deustuko Unibertsitatea



From <http://blogs.zdnet.com/Hinchcliffe>



Bizkaia
Empresa Digitala



Universidad de Deusto
Deustuko Unibertsitatea



Amazon Web Services (AWS)

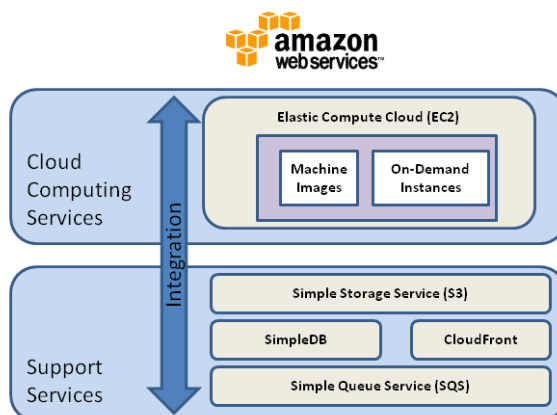
- AWS proporciona una **infraestructura de servicios elástica donde alojar computación, almacenamiento o sistemas empresariales**
 - **Amazon Elastic Cloud (EC2)** – permite configurar y ejecutar un Amazon Machine Instance (AMI) – servidores en demanda
 - **Amazon Simple Storage Service (S3)** – permite guardar y recuperar datos en la nube
 - **Amazon SimpleDB** – proporciona la funcionalidad de una base de datos sobre S3 – basada en pares clave-valor
 - **Amazon Simple Queue Service (SQS)** – servicio de mensajería para encolar tareas y mensajes
 - ...
 - <http://aws.amazon.com/documentation/>



109



Amazon Web Services (AWS)



110



Google App Engine vs. Amazon Web Services

- A primera vista Google App Engine es un competidor a la suite de servicios web ofrecida por Amazon:
 - S3 para almacenamiento
 - EC2 para la creación de servidores virtuales
 - SimpleDB como base de datos
- Pero ...
 - Google App Engine es una plataforma mucho más acoplada y menos flexible
 - Si quieres hacer uso de BigTable tienes que escribir y desplegar un script de Python a su servidor de aplicaciones ofreciendo una interfaz web accesible a BigTable



111



Web Semántica

- Problema de la Web Actual:
 - El significado de la web no es comprensible por máquinas
- **Web Semántica** → crea un medio universal de intercambio de información, aportando semántica a los documentos en la web
 - Añade significado comprensible por ordenadores a la Web
 - Usa técnicas inteligentes que explotan esa semántica
 - Liderada por Tim Berners-Lee del W3C
- **Misión** → “turning existing web content into machine-readable content”



112

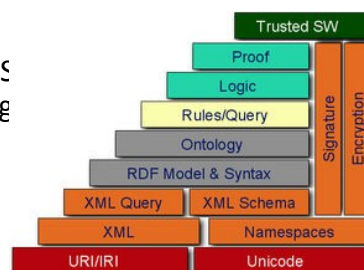


La Pila de la Web Semántica

- La Web Semántica está compuesta de:
 - XML, sintaxis para documentos estructurados
 - XML Schema, restringe la estructura de documentos XML
 - RDF es un modelo de datos que hace referencia a objetos y sus relaciones
 - RDF Schema, vocabulario para definir propiedades y clases de recursos RDF
 - OWL, añade más vocabulario que RDF: relaciones entre clases, cardinalidad, etc.



113



RDF: Recurso, Propiedad y Valor

- RDF identifica conceptos usando identificadores Web (URIs), y describe recursos con propiedades y valores de las mismas
- Definiciones:
 - Un **Recurso** es cualquier cosa que puede tener una URI, como por ejemplo "http://www.w3schools.com/RDF"
 - Una **Propiedad** es un Recurso que tiene un nombre, como "autor" o "página web"
 - Un **Valor de propiedad** es el valor de una Propiedad, tal como "Diego Ipiña" o "http://www.w3schools.com" (un valor de propiedad puede corresponder a un recurso)



114



Resource Description Framework (RDF)

- Un grafo RDF crea una web de conceptos
 - Realiza aserciones sobre relaciones lógicas entre entidades
- Información en RDF puede ligarse con grafos en otros lugares
 - Mediante software se pueden realizar inferencias
 - Lenguajes de consulta sobre triple stores como SPARQL
- Mediante RDF hacemos que la información sea procesable por máquinas
 - Agentes software pueden guardar, intercambiar y utilizar metadatos sobre recursos en la web
- **Ontología** → jerarquía de términos a utilizar en etiquetado de recursos



115



Ejemplo de Formatos RDF

- **Formato RDF/XML:**

```

1: <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
2:   xmlns:dc="http://purl.org/dc/elements/1.1/"
3:   xmlns:geo="http://www.w3.org/2003/01/geo/wgs84_pos/"
4:   xmlns:edu="http://www.example.org/">
5:   <rdf:Description rdf:about="http://www.deusto.es">
6:     <geo:lat>43.270737</geo:lat>
7:     <geo:long>-2.939637</geo:long>
8:     <edu:hasFaculty>
9:       <rdf:Bag>
10:        <rdf:li rdf:resource="http://www.eside.deusto.es" dc:title="Facultad de
            Ingeniería"/>
11:        <rdf:li rdf:resource="http://www.lacomercial.deusto.es" dc:title="Facultad de
            Económicas y Empresariales"/>
12:       </rdf:Bag>
13:     </edu:hasFaculty>
14:   </rdf:Description>
15: </rdf:RDF>

```
- **Formato N3/Turtle:**

```

1: @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
2: @prefix dc: <http://purl.org/dc/elements/1.1/> .
3: @prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .
4: @prefix edu: <http://www.example.org/> .
5: <http://www.deusto.es> geo:lat "43.270737" ; geo:long "-2.939637" .
6: <http://www.eside.deusto.es> dc:title "Departamento de Ingeniería" .
7: <http://www.deusto.es> edu:hasFaculty <http://www.eside.deusto.es> .

```

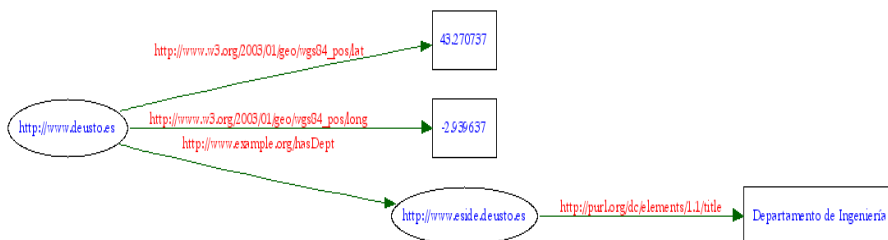


116



Ejemplo de Grafo RDF

- Generador de grafos RFD:
<http://www.w3.org/RDF/Validator/>



117



¿Qué es una Ontología?

- Una ontología define conceptos de un dominio y relaciones entre ellos
- Los bloques básicos que componen el diseño de una ontología son:
 - clases o conceptos
 - propiedades de cada concepto describiendo varias características y atributos del concepto
 - restricciones sobre las propiedades
- Una **ontología junto con las instancias de sus clases** individuales constituyen un **knowledge base**.



118



Características del Ontology Web Language (OWL)

- Una ontología difiere de un esquema XML en que es una representación de conocimiento, no un formato de mensaje
- La principal ventaja de una ontología escrita en OWL es que hay disponibles herramientas que pueden razonar sobre ella
- La sintaxis de intercambio de información en OWL es RDF/XML.
- OWL es una extensión del vocabulario de RDF
- Las ontologías Web son distribuidas
- Pueden ser importadas y extendidas para crear ontologías derivadas.



119



Ejemplo OWL Ontology Reasoning

- Supongamos el siguiente modelo RDF:


```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
<http://www.ipina.org/> foaf:author
  <http://www.ipina.org/osgi/> .
<http://www.deusto.es/dipina/> foaf:author
  <http://www.deusto.es/dipina/ajax/> .
<http://www.eside.deusto.es/dipina/> foaf:author
  <http://paginaspersonales.deusto.es/dipina/> .
```
- Aunque pertenecen al mismo autor, no están relacionadas entre ellas, con la ayuda de OWL podemos mapear estas URIs


```
@prefix owl: <http://www.w3.org/2002/07/owl#> .
<http://www.deusto.es/dipina/> owl:sameAs
  <http://www.ipina.org/> .
<http://www.eside.deusto.es/dipina/> owl:sameAs
  <http://www.ipina.org/> .
```
- Si mezclamos ambos modelos y ejecutamos un razonador podríamos responder a "dime todo lo que ha escrito "<http://www.ipina.org>":


```
<http://www.ipina.org/osgi/>, <http://www.deusto.es/dipina/ajax/> y <http://paginaspersonales.deusto.es/dipina/>
```



120



SPARQL

- SPARQL (<http://www.w3.org/TR/rdf-sparql-query/>) permite la consulta de grafos RDF a través de un lenguaje sencillo
- SPARQL es idóneo para extraer y consultar información mantenida por aplicaciones, servicios o repositorios ad-hoc de terceras partes expresados en RDF
- Consta de 3 elementos:
 - Lenguaje de consultas.
 - Mecanismo para transmitir una consulta a un servicio de procesamiento de consultas remoto
 - Formato XML en el que devolver los resultados



121



Ejemplo SPARQL

```

PREFIX table: <http://www.daml.org/2003/01/periodictable/PeriodicTable#>
SELECT ?symbol ?number
FROM <http://www.daml.org/2003/01/periodictable/PeriodicTable#>
WHERE
{
  {
    ?element table:symbol ?symbol;
              table:atomicNumber ?number;
              table:group table:group_17.
    OPTIONAL { ?element table:color ?color. }
  }
  UNION
  {
    ?element table:symbol ?symbol;
              table:atomicNumber ?number;
              table:group table:group_18.
  }
}
ORDER BY DESC(?number)
LIMIT 10
OFFSET 10

```



122



Microformatos



- Añade semántica simple al marcado para hacer que no sólo sea legible y renderizable sino también comprensible por la máquina
- Los microformats aportan elementos de datos (eventos, contactos o localizaciones) sobre HTML de modo que pueden ser detectados y la información contenida en ellos extraída por software, indexada, etc.
- Ejemplo:

```
<span class="geo"><span class="latitude">52.48</span>,
  <span class="longitude">-1.89</span></span>
```
- Firefox 3 e IE8 proveen soporte para microformatos
- Ejemplo:
 - hCard creator: <http://tantek.com/microformats/hcard-creator.html>
 - Herramienta: Firefox Operator (<https://addons.mozilla.org/es-ES/firefox/addon/4106>)



123



RDFa

- **RDFa = Haciendo más fácil empotrar RDF en XHTML y XML**
 - Resuelve la principal limitación de RDF: es muy comprensible por máquinas pero muy poco por humanos
 - Añade metadatos a XHTML sin afectar la visualización de navegadores
 - La página web es ahora legible tanto por humanos como procesos automatizados
 - Permite la agregación de datos y la asociación de metadatos para realizar tareas más sofisticadas que aquellas permitidas mediante *screen scrapping*
 - Utiliza algunos atributos de XHTML 1 existentes y otros nuevos de XHTML 2 para guardar sujetos, predicados y objetos de tripletas RDF
 - Los atributos XHTML 1 href, content, rel, rev, y datatype
 - Los nuevos atributos de meta-información about, role y property de XHTML 2 (<http://www.w3.org/TR/xhtml2/mod-meta.html>) module
 - Ejemplo:

```
<p about="http://www.deusto.es/events/event1"
  instanceof="cal:Vevent">
```
- Referencia: <http://rdfa.info/>



124



GRDDL

- **GRDDL** es un mecanismo para extraer descripciones de recursos de dialectos de lenguajes
 - Define un estándar para declarar que una página web o XML puede ser transformada en un grafo RDF, así como identifica el algoritmo a utilizar para tal transformación
 - glean -verb (used with object)
 1. to learn, discover, or find out, usually little by little or slowly.
 - verb (used without object)
- Es un **lenguaje de marcado para ...**
 - ... **declarar que un documento XML incluye datos extraíbles** (PROFILE) o
 - ... **ligarlo a un algoritmo** (típicamente XSLT) **para extraer los datos RDF** del documento (TRANSFORMATION)
- El lenguaje de marcado incluye:
 1. Un atributo de espacio de nombres para ser incluido en documentos XML


```
xmlns:grddl='http://www.w3.org/2003/g/data-view#'
grddl:transformation="glean_title.xsl"
```
 2. Un enlace asociado a un perfil para incluirse en documentos XHTML válidos.


```
<head profile="http://www.w3.org/2003/g/data-view">
<link rel="transformation" href="glean_title.xsl" />
```



125



Resumen de Mecanismos para Añadir Semántica

- **Microformatos**
 - Diseñados primero para los humanos y después para las máquinas
 - Pequeños pasos para resolver problemas específicos uno a uno
 - No puede ser validado fácilmente (por ejemplo si mezclas los microformatos hCard y hCal no hay modo de garantizar una correcta interpretación)
 - ```
Jane
```
- **RDFa**
  - Cuando la ontología subyacente es demasiado complicada para ser expresada como un microformato
    - los microformatos son buenos para micro-metadatos (e.j. nombre, dirección, fechas de eventos) pero no para datos complejos (e.j. proteínas, datos geológicos)
  - Cuando necesitas combinar varios vocabularios en una página
    - gestionar posibles conflictos entre primitivas e.j. class="name" and class="name" vs. rel="foaf:name" and rel="prod:name"
- **GRDDL**
  - Middleware para juntar microformatos y RDFa
    - Pegamento para las diferentes técnicas de semantización



**¡¡Todas juntas nos ayudarán a alcanzar la visión de una web con más significado, pero todavía comprensible tanto a humanos como máquinas!**

126



## Web Semántica vs. web semántica

|           | Web Semántica                                                                                                                                             | web semántica                                                                                   |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| Filosofía | Construye un formato común de datos para expresar el significado de los datos. Utiliza ontologías para ayudar a las máquinas a entender el contenido web. | Los humanos primero, las máquinas después. Codifican el contenido web con etiquetas especiales. |
| Lenguaje  | RDF, RDFS, OWL                                                                                                                                            | Microformats (basados en XHTML), RDFa, GRDDL                                                    |
| Formato   | Debe contener documentos RDF bien formados                                                                                                                | Cualquier cosa vale, siempre que sea XHTML                                                      |
| Semántica | Definida por el modelo ontológico subyacente (e.j., OWL)                                                                                                  | Desacoplado. No hay modelo semántico formal, a no ser que se use RDFa                           |
| Ejemplos  | FOAF, OWL-S, OWL-Time                                                                                                                                     | XFN (red social), hCard (contacto), hReview (opiniones), rel-tag (etiquetado)                   |

## Yahoo! SearchMonkey

- Precursor de generador de resultados de búsquedas mejorados
  - Precedió a Google Snippets y ya es funcional
- Usa datos estructurados para mejorar la presentación de resultados de búsquedas
- Ejemplos:
  - <http://search.yahoo.com/search?p=avila+boston>
  - <http://gallery.search.yahoo.com/>





[Drooling Dog Bar B.Q. - Colfax, CA](#)

★★★★☆ 15 reviews - Price range: \$\$

Drooling Dog has some really good BBQ. I had the pulled pork sandwich, .... Drooling Dog BBQ is a great place to stop at on your way up the hill to Tahoe ...

[www.yelp.com/biz/drooling-dog-bar-b-q-colfax](http://www.yelp.com/biz/drooling-dog-bar-b-q-colfax) - 75k - [Cached](#) - [Similar pages](#)

# Google Rich Snippets

- Si colocas en tus páginas ciertos microformatos (hCard, hReview, hProduct) o anotaciones que servirán para mejorar los resultados de búsqueda
  - Dan un resumen de sus resultados de búsqueda de un golpe
  - Interesante porque:
    - Google sugiere el uso de marcado semántico, sin recurrir a su especialidad en machine learning
    - Hay un móvil económico para el uso de marcado semántico

```
<div class="hreview-aggregate">
<div class="item vcard">
<h1 class="h org">Drooling Dog Bar B Q</h1>
...

based on 15 reviews
...
Price range: $$
</div>
</div>
```

```
<div xmlns:v="http://rdf.data-vocabulary.org/#" typeof="v:Review-aggregate">
<div rel="v:itemreviewed">

<h1 property="v:name">Drooling Dog Bar B Q</h1>
...

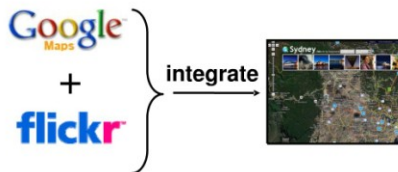
based on 15 reviews
...
Price Range: $$

</div>
</div>
```

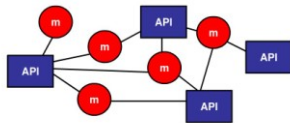
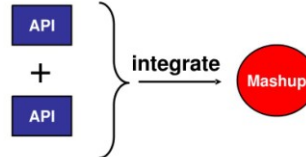


## A Mash-up ...

is **made** of...



is **made** of...



and these are **registered** in

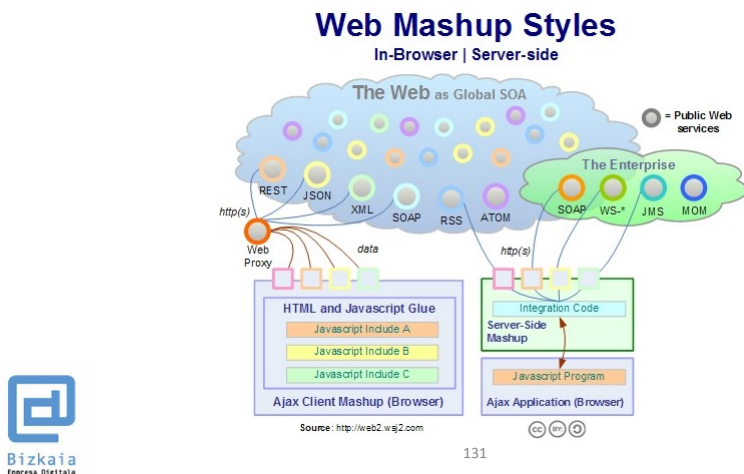


**mashup ecosystem**



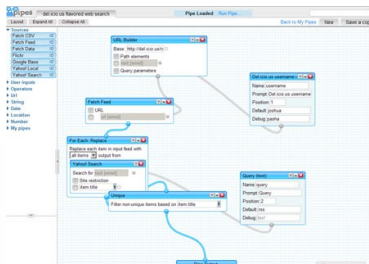
## Estilos de Mash-ups

- <http://hinchcliffe.org>



## Mecanismos Actuales para Crear un Mashup

- Escribirlo a mano
  - <http://www.programmableweb.com/matrix>
- Usar herramientas que te ayudan a su construcción (<http://mashable.com/2007/07/08/mashups/>):
  - Google Mashup Editor y Microsoft PopFly (<http://www.popfly.com>)
  - Intel Mash Maker (<http://mashmaker.intel.com>)
    - <http://mashmaker.intel.com/web/gallery.php#Google+Images>
  - Yahoo! Pipes (<http://pipes.yahoo.com>)
    - Ejemplo: [http://pipes.yahoo.com/pipes/pipe.edit?\\_id=fELaGmGz2xGtBTC3qe5lKa](http://pipes.yahoo.com/pipes/pipe.edit?_id=fELaGmGz2xGtBTC3qe5lKa)
  - IBM's QEDWiki



## Problemática Web 2.0 y Mashups

- La Web 2.0 social define **un portal como un “jardín cerrado”** (*walled garden*) donde sus frutos difícilmente pueden ser recogidos y combinados con los de otros portales.
- Una manera de **romper los muros de los jardines de datos** de los portales Web 2.0 actuales es ofrecer **Open APIs**
  - A menudo, estas APIs dan lugar a estructuras de datos en XML que deben ser procesadas e integradas por los consumidores
  - RDF y OWL permiten combinar las respuestas de un portal con las de otro
    - haciendo que las sentencias RDF de dos localizaciones diferentes hagan referencia al mismo concepto (URI)
    - estableciendo correspondencias mediante OWL indicando que dos conceptos son equivalentes
- El uso de RDF para generar respuestas desde diferentes portales de datos distribuidos permite el uso del lenguaje de consultas SPARQL



133



## Problemática de la Web Semántica

- La Web Semántica presenta **dos problemas principales para su adopción global**:
  - **disponibilidad de los datos y**
    - Necesidad de creación de interfaces que permitan a usuarios web convencionales contribuir a la Web Semántica
  - **diseño de interfaces de interacción**
    - Desarrollar interfaces que permitan visualizar más cómodamente los grafos de relaciones entre conceptos que constituyen la Web Semántica



134



## Mash-ups Web 2.0

- HASTA AHORA ...
  - Los mashups funcionan si estás buscando un único tipo de cosa (cafés, hoteles, gimnasios) y vienen de una fuente
- En los mash-ups Web 2.0 tradicionales, **cada vez que se integra una nueva fuente de información es preciso desarrollar un nuevo adaptador** que convierta los datos capturados al formato interno utilizado en el mash-up.
  - En el mash-up semántico bastaría rellenar un formulario web donde se especifica la URL de publicación de eventos de un portal, así como un conjunto mínimo de metadatos para automatizar el proceso



135



## Mash-up Semánticos

- Los mashups semánticos **mezclan información proveniente de diferentes fuentes** y organizan los resultados
- Los **mash-up semánticos son mucho más flexibles** – convierten la información recuperada a formato RDF (*lingua franca*) fácilmente filtrable y consultable con SPARQL.
  - **Tienen la capacidad de evolucionar** sin requerir cambios en su código
    - Los datos provistos en formatos de representación sintácticos diferentes, pero tales **datos, semánticamente, deben proveer una información muy similar**, fácilmente convertible a un vocabulario RDF común.



136



## Mash-up Semánticos

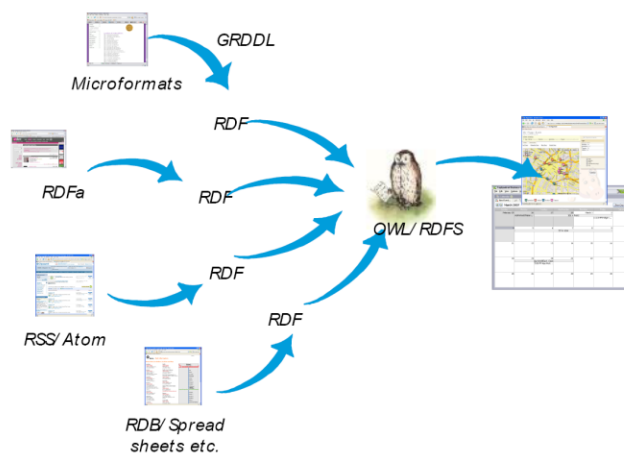
- ... o como la Web 2.0 actual puede beneficiarse de la adopción de ciertas tecnologías semánticas para dar lugar a portales web más inteligentes
  - Potencial combinación de Web 2.0 y Web Semántica
- Ejemplo:
  - mash-up semántico capaz de agregar información heterogénea sobre eventos proveniente de diferentes organizaciones y de mostrarla de manera conjunta sobre un mapa de GoogleMaps



137



## Mash-ups Semánticos



138



# Problemas de las Herramientas Actuales

- Un servicio REST o SOAP puede hacer cualquier cosa
  - Cualquier Input
  - Cualquier Output
  - Diferentes modos de invocarse
- Debe efectuarse mediación de datos para poder ligar las salidas de un servicio REST o SOAP con otro, es decir, para crear ecosistemas de mashups
- Hay varias iniciativas que ayudan en ello:
  - SAWSDL (Semantically Annotated Web Services)
  - SA-REST – añadir semántica a servicios basados en REST
    - Web Application Description Language (WADL)
      - <https://wadl.dev.java.net/wadl20090202.pdf>
    - WSDL 2.0
      - <http://www.ibm.com/developerworks/webservices/library/ws-restwsdl/>



139



## Ejemplo de SA-REST

```
<html xmlns:sarest="http://1sdis.cs.uga.edu/SAREST#">
 <body>
 <p about="http://1sdis.cs.uga.edu/~jlathem/FindPerson">
 This service takes a

 http://xmlns.com/foaf/0.1/#person
 as an input.
 The service returns a

 http://otheront/#location

 as the result.
 The page should be invoked by an
 <meta property="sarest:catapouamus" content="HTMLGET"/>
 <meta property="sarest:upcast" content="test"/>
 <meta property="sarest:downcast" content="test2"/>
 <meta property="sarest:operation" content="LookupPerson"/>
 </p>
 </body>
</html>
```



140



## Ejemplo Ontología de Mash-up

- Ontología que modela “Eventos geo-localizados organizados por una entidad”
  - en vez de diseñar una ontología desde cero, hemos preferido hacer uso de tres vocabularios RDF bien conocidos tales como Calendar, vCard y Geo, que nos permiten representar eventos con las características que deseamos
    - RDF Calendar (<http://www.w3.org/TR/2005/NOTE-rdfcal-20050929/>)
    - vCard-RDF (<http://www.w3.org/TR/vcard-rdf>)
    - Geo RDF (<http://www.w3.org/2003/01/geo/>)
  - Y de sus microformatos equivalentes:
    - hCalendar (<http://microformats.org/wiki/hcalendar>)
    - hCard (<http://microformats.org/wiki/hcard>)
    - geo (<http://microformats.org/wiki/geo>)



141



## Evento geolocalizado expresado con hCalendar

```
<div id="empresadigital: event1" class="vevent">
 <h1>OSGI: una Plataforma Modular y Ligera para
 Construir Aplicaciones basadas en Servicios</h1>
 <h2><abbr title="20080401T0900" class="dtstart">22 Mayo del 2008 9am a
 </abbr><abbr title="20080401T1400" class="dtend">2pm</abbr> en Parque Tecnológico de Álava, Miñano</h2>
 <abbr class="geo" title="42.883; -2.760"></abbr>
 <p class="description">OSGi es una de las tecnologías de software
 distribuido que más impacto está causando últimamente. Ven a este curso
 y aprende que hay detrás de OSGi.Pulsa para
 obtener más información.</p>
 <p>Organizado por
 <span class="fn
 org">Araba Empresa Digitala. Contáctanos en <a class="email"
 href="mailto:events@empresadigitala.net"> events@empresadigitala.net
 </p>
 <p> </p>
</div>
```



142



## Evento geolocalizado en RDFa

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!--<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN" "http://www.w3.org/Markup/DTD/xhtml1-rdfa-1.dtd"-->
<html xmlns:cal="http://www.w3.org/2002/12/cal/ical#"
 xmlns:contact="http://www.w3.org/2001/vcard-rdf/3.0#"
 xmlns:geo="http://www.w3.org/2003/01/geo/wgs84_pos#">
 <head>
 <title>Eventos de la Facultad de Ingenieria de la Universidad de Deusto</title>
 </head>
 <body>
 <p about="http://www.deusto.es/events/event1" instanceof="cal:Vevent">
 Charla técnica:

 Web con minusculas

 a celebrarse el dia

 12 de Marzo a las 4pm.

 en Auditorio Principal Universidad de Deusto

 43.270737
 -2.939637

 organizado por
 Diego Lopez de

 </p>
 ...
 </body>
</html>
```



Ipina</span>  
</p>  
...

143



## Evento geolocalizado expresado en RDF

```
<?xml version="1.0" encoding="iso-8859-1"?>
<rdf:RDF xmlns:cal="http://www.w3.org/2002/12/cal/icaltzd#"
 xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
 xmlns:geo="http://www.w3.org/2003/01/geo/wgs84_pos#"
 xmlns:vcard="http://www.w3.org/2001/vcard-rdf/3.0#">
 <rdf:Description rdf:about="http://www.revistasprofesionales.com/solop#event2">
 <rdf:type rdf:resource="http://www.w3.org/2002/12/cal/icaltzd#Vevent"/>
 <cal:organizer rdf:resource="http://www.revistasprofesionales.com/solop"/>
 <cal:dtend rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2008-05-27</cal:dtend>
 <cal:summary rdf:datatype="http://www.w3.org/1999/02/22-rdf-syntax-ns#XMLLiteral">Charla sobre Web 3.0</cal:summary>
 <cal:description rdf:datatype="http://www.w3.org/1999/02/22-rdf-syntax-ns#XMLLiteral">Ven a ver lo último en la sinergia entre Web 2.0
 y Web Semántica (GRDDL, RDFa, microformátos)</cal:description>
 <cal:url rdf:resource="http://www.revistasprofesionales.com/solop/charla2"/>
 <cal:dtstart rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2008-05-27</cal:dtstart>
 <cal:location rdf:datatype="http://www.w3.org/1999/02/22-rdf-syntax-ns#XMLLiteral">Madrid, España</cal:location>
 <geo:Point>
 <rdf:Description rdf:about="#p1">
 <geo:lat rdf:parseType="Literal">40.437</geo:lat>
 <geo:long rdf:parseType="Literal">-3.625</geo:long>
 </rdf:Description>
 </geo:Point>
 </rdf:Description>
</rdf:RDF>
```



</rdf:RDF>

144





## Consulta que extrae eventos agregados

```

PREFIX cal: <http://www.w3.org/2002/12/cal/icaltzd#>
PREFIX contact: <http://www.w3.org/2001/vcard-rdf/3.0#>
PREFIX geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
SELECT DISTINCT ?summary ?description ?dtstart ?dtend ?lat ?long ?org ?url ?email ?logo
WHERE {
 ?event cal:summary ?summary.
 ?event cal:dtstart ?dtstart.
 OPTIONAL { ?event cal:organizer ?organizer.
 ?organizer contact:fn ?org.
 ?organizer contact:url ?url.
 ?organizer contact:email ?email.
 OPTIONAL { ?organizer contact:logo ?logo. } }
 OPTIONAL { ?event geo:Point ?point.
 ?point geo:lat ?lat.
 ?point geo:long ?long. }
 OPTIONAL { ?event cal:geo ?loc.
 ?loc <http://www.w3.org/1999/02/22-rdf-syntax-ns#first> ?lat.
 ?loc <http://www.w3.org/1999/02/22-rdf-syntax-ns#rest> ?restgeo.
 ?restgeo <http://www.w3.org/1999/02/22-rdf-syntax-ns#first> ?long. }
 OPTIONAL { ?event cal:dtend ?dtend. }
 OPTIONAL { ?event cal:description ?description. }
 FILTER (xsd:dateTime(?dtstart) >= xsd:dateTime("2008-04-14T00:00:00Z"))
 FILTER (xsd:dateTime(?dtend) <= xsd:dateTime("2008-04-21T00:00:00Z"))
 ORDER BY ?event;

```



145



## Mashup Semántico

146



## Conclusiones

- Sólo una nueva Internet posibilitará de un modo robusto nuevos servicios y funcionalidades
  - Internet of Services, Internet of Things, ...
- La Web del Futuro será una plataforma de ejecución de servicios RIA cada vez más inteligentes
- Los paradigmas Cloud Computing e Internet de las Cosas van a cambiar el modo de desplegar funcionalidad empresarial y cotidiana en la Web
  - Todo va a alojarse en la Web: la Web va a ser el ordenador y el navegador el SO que gestiona los recursos de la Web



147



## Referencias

- European Future Internet Portal, <http://www.future-internet.eu/>
- The Future of the Internet, Bled 31 March 2008, [ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/ch1-g940-280-future-internet-ld\\_en.pdf](ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/ch1-g940-280-future-internet-ld_en.pdf)
- Future Internet: The Cross-ETP Vision Document, Version 1, 8. January, 2009, [http://www.future-internet.eu/fileadmin/documents/reports/Cross-ETPs\\_FI\\_Vision\\_Document\\_v1\\_0.pdf](http://www.future-internet.eu/fileadmin/documents/reports/Cross-ETPs_FI_Vision_Document_v1_0.pdf)
- How Web 3.0 Will Work, <http://computer.howstuffworks.com/web-30.htm>
- Three New Search Services: Wolfram|Alpha, Microsoft Bing, Google Squared, <http://dltj.org/article/alpha-bing-squared/>
- Bing Search Engine, [http://en.wikipedia.org/wiki/Bing\\_\(search\\_engine\)](http://en.wikipedia.org/wiki/Bing_(search_engine))



148



## Referencias

- Cloud Computing – Disruptive Innovation & Enabling Technology,  
<http://blog.gogrid.com/2008/08/20/presentation-cloud-computing-disruptive-innovation-enabling-technology/>
  - [http://www.slideshare.net/aditya\\_n\\_jha/cloud-computing-and-amazon-web-services-presentation](http://www.slideshare.net/aditya_n_jha/cloud-computing-and-amazon-web-services-presentation)
- Web Applications 1.0, <http://www.whatwg.org/specs/web-apps/2005-09-01/>
- The Internet of Things
  - [http://www.itu.int/osg/spu/publications/internetofthings/InternetofThings\\_summary.pdf](http://www.itu.int/osg/spu/publications/internetofthings/InternetofThings_summary.pdf)
- Enterprise 2.0, Web 2.0, WOA
  - <http://blogs.zdnet.com/Hinchcliffe/>
  - <http://hinchcliffe.org/>
- Programmable Web
  - <http://www.programmableweb.com>



149



## Referencias

- Opera Labs – <http://labs.opera.com/>
  - <http://unite.opera.com/>
- Mozilla Labs – <http://labs.mozilla.com/>
  - <http://wave.google.com/>
- Google App Engine
  - <http://code.google.com/intl/en/appengine/docs/python/gettingstarted/>
  - <http://code.google.com/intl/en/appengine/docs/java/gettingstarted/>
- Amazon Web Services
  - <http://aws.amazon.com/>
  - <http://developer.amazonwebservices.com/connect/kbcategory.jspa?categoryID=59>



150





## **Hacia la Internet del Futuro: Web 3.0 e Internet de los Servicios**

17 de Septiembre 2009, 10:00-13:00

Parque Tecnológico de Zamudio, Edificio ESI, #204

**Dr. Diego Lz. de Ipiña Glz. de Artaza**

[dipina@eside.deusto.es](mailto:dipina@eside.deusto.es)

<http://paginaspersonales.deusto.es/dipina>