

Programación Distribuida y Tiempo Real

Bases de Datos Distribuidas

Clase Introductoria 1

Bibliografía:

G. Coulouris, J. Dollimore, T. Kindberg, "Sistemas Distribuidos. Conceptos y Diseño", 3ra. Ed., Addison Wesley, 2003. "Distributed Systems - Concepts and Design", 4th Ed., George Coulouris, Jean Dollimore, Tim Kindberg. Addison-Wesley, June 2005, 5th Ed. George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair, Pearson, 2011 - ***El más conceptual***

A. S. Tanenbaum, M. van Steen, "Distributed Systems: Principles and Paradigms", 2nd Ed., Prentice Hall, 2007, "Distributed Systems", M. van Steen, A. S. Tanenbaum, 2017,
<https://www.distributed-systems.net/index.php/books/distributed-systems-3rd-edition-2017>

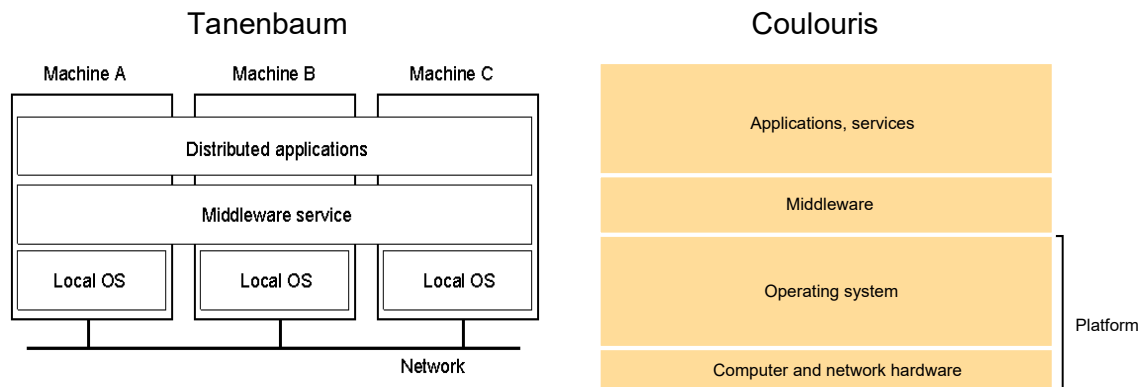
El que tiene más detalles específicos de algunos temas conceptuales

M. L. Liu, "Distributed Computing: Principles and Applications", Addison-Wesley, 2004 - ***Bien específico de algunos temas, incluye ejemplos/programas***

Hay bibliografía orientada a tratamiento formal/algebraico de sistemas distribuidos. Ventajas y desventajas.

1. Horarios y desarrollos de las materias: entrega de TPs en ambos casos
2. Definiciones de Sistemas Distribuidos:
 - 2.1. Tanenbaum (Tan-chap-01.ppt) "A collection of independent computers that appears to its users as a single coherent system"
 - 2.2. Coulouris: "... es aquel en el que los componentes localizados en computadores, conectados en red, comunican y coordinan sus acciones únicamente mediante el paso de mensajes." (pág. 1, primera frase)
 - 2.3. Liu (Liu-chapter1.ppt) "A distributed system is a collection of independent computers, interconnected via a network, capable of collaborating on a task"

3. El gráfico clásico de los libros/asignaturas de sistemas distribuidos



4. Evolución de los sistemas distribuidos (un poco de historia, quizás *no oficial*)

4.1. Redes locales

- 4.1.1. Hardware (impresora, almacenamiento)
- 4.1.2. Recursos (hardware y *datos-filesystem*)
- 4.1.3. Cliente/servidor
- 4.1.4. Protocolos (de redes locales)
- 4.1.5. APIs

4.2. El *salto* al exterior de las redes locales

- 4.2.1. IP: por qué no alcanza con el hardware (identificaciones-nombres)
- 4.2.2. Protocolos *de Internet*: por qué no alcanza con IP, *ports*
- 4.2.3. API(*¿s?*), *sockets*
- 4.2.4. Local - No local (*todo* sobre IP)

4.3. El *salto* de las aplicaciones sobre IP (Internet)

- 4.3.1. http: servidores y browsers/navegadores (buscadores, por favor)
- 4.3.2. Protocolos + desarrollo de software (APIs, IDEs, modelos, etc.)
- 4.3.3. Cantidades de computadoras (Cou-Chapter 1 slides.ppt, pp. 6-7)
- 4.3.4. “Aplicaciones web”

Warning, pre 2011 wikipedia def., (Web application, Web_application.htm):

“In software engineering, a web application or webapp [1] is an application that is accessed via web browser over a network such as the Internet or an intranet. It is also a computer software application that is coded in a browser-supported language (such as HTML, JavaScript, Java, etc.) and reliant on a common web browser to render the application executable.

Web applications are popular due to the ubiquity of web browsers, and the convenience of using a web browser as a client, sometimes called a thin client. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for their popularity. Common web applications include webmail, online retail sales, online auctions, wikis and many other functions.”

Warning, Feb. 2017 wikipedia def., In computing, a web application or web app is a client–server software application in which the client (or user interface) runs in a web browser.[1] Common web applications include webmail, online retail sales, online auctions, wikis, instant messaging services and many other functions.

...

The general distinction between a dynamic web page of any kind and a "web application" is unclear. Web sites most likely to be referred to as "web applications" are those which have similar functionality to a desktop software application, or to a mobile app. HTML5 introduced explicit language support for making applications that are loaded as web pages, but can store data locally and continue to function while offline.

Single-page applications are more application-like because they reject the more typical web paradigm of moving between distinct pages with different URLs. Single-page frameworks like Sencha Touch and AngularJS might be used to speed development of such a web app for a mobile platform.

(Diferencia entre Javascript y Flash o Java)

Browser (nosotros navegamos, el browser "muestra")

Browser + Plug-Ins

Browser + JavaScript

RIA (Rich Internet Application)

Warning, pre 2011 wikipedia def. (Rich Internet Application, Rich_Internet_Application.htm):

Rich Internet applications (RIAs) are web applications that have some of the characteristics of desktop applications, typically delivered by way of a proprietary web browser plug-ins or independently via sandboxes or virtual machines[1]. Examples of RIA frameworks include Adobe Flex/AIR, Java/JavaFX[2] and Microsoft Silverlight[3].

The term was introduced in the 1990s by vendors like Macromedia who were addressing limitations at the time in the "richness of the application interfaces, media and content, and the overall sophistication of the solutions" by introducing proprietary extensions[4].

As web standards (such as Ajax and HTML 5) have developed and web browsers' compliance has improved there is less need for such extensions, and Javascript compilers with their associated desktop-like widget sets reduce the need for browser extensions even further.[citation needed] HTML 5 delivers a full-fledged application platform; "a level playing field where video, sound, images, animations, and full interactivity with your computer are all standardized"[5].

It is now possible to build RIA-like Web applications that run in all modern browsers without the need of special run-times or plug-ins. This means that if one could run a modern Ajax-based Web application outside of a web browser (e.g. using Mozilla Prism or Fluid) it would essentially be an RIA[1], though there is some contention as to whether this is actually the case.[6]

Warning, Feb. 2017 wikipedia def., Google trends shows (as of September 2012) that frameworks based on a plug-in are in the process of being replaced by HTML5/JavaScript-based alternatives

No parece haber una línea definida que divida web application - RIA

Distribución (deployment), mantenimiento/soporte técnico

4.3.5. Aplicaciones móviles

4.3.5.1. ¿Por qué sería “sobre IP”?

4.3.5.2. Por qué sería “distribuido”

4.3.5.3. Capacidades y características (“responsivo”)

4.3.5.4. Las *restricciones* “a c/s”

5. Las historias “oficiales” involucran números de usuarios y computadoras, evolución de tecnología (Ej. Cap. 1 Liu, Liu-chapter1.ppt: muchas páginas con números de usuarios, computadoras, etc.), evolución de aplicaciones. Más asociado a Internet que a sistemas distribuidos. Quizás lo más relevante de números: Cou-Chapter 1 slides.ppt, pp. 6-7. A las que habría que “sumarle” ahora las evoluciones en cantidades de dispositivos móviles en general y teléfonos inteligentes en particular.