



EVOLUCIÓN DE LOS MÉTODOS DE ANÁLISIS DE SISTEMAS TRADICIONALES

1. Evolución de los lenguajes de programación

No es muy difícil demostrar que la historia de los métodos de análisis de sistemas sigue a la evolución de los lenguajes de programación. Incluso son muchos los autores que opinan de esta manera. Una buena exposición del progreso de los lenguajes se puede ver en el trabajo de Grady Booch¹, obra que es posible considerar como un clásico de la orientación a objetos.

Al repasar la historia de la programación podemos encontrar, en primer término, los lenguajes de máquina y su versión un poco más legible, como es el lenguaje ensamblador. En un principio, estos lenguajes contenían un juego reducido de instrucciones que permitían realizar ciertas operaciones booleanas y se hallaban completamente ligados a las distintas arquitecturas de procesadores a las que pertenecían. Posteriormente, a mediados de la década del cincuenta, aparecieron los **lenguajes de primera generación**, que permitían "alejarse" un poco de la máquina y obtener soluciones, en especial, en el área de las matemáticas. Los programas escritos con ellos accedían a una masa común de datos, de manera que cualquier error en un programa podía tener "un devastador efecto de propagación a través del resto del sistema"².

Los **lenguajes de segunda generación**, aparecidos al comienzo de los sesenta, dieron origen a los subprogramas, que permiten enfocar la programación desde una óptica diferente. Los subprogramas constituyeron un gran salto conceptual al proveer un potente mecanismo de abstracción para factorizar funciones. Allí entraron en escena los lenguajes con pasaje de parámetros pero, fundamentalmente, comenzaron a sentarse las bases de las técnicas estructuradas. De todas maneras, se puede observar que la evolución se va dando sólo en la mejora del tratamiento de los procesos, porque los datos continúan siendo comunes a los distintos programas y no hay un progreso sustancial en este aspecto. En definitiva, el avance se observa en una mejora en el aspecto funcional y no tanto en el establecimiento de las estructuras de datos.

¹ Cfr. BOOCH, Grady. Op.cit. p.32 y ss.

² Ibidem p.34



Por último, los **lenguajes de tercera generación** permitieron realizar una programación modular, pero los datos aún no contaban con un desarrollo semejante al de la parte procedimental. Es cierto que el acceso a los datos pasó de los métodos de lectura de la cinta secuencial al empleo de los archivos de índices, pero a pesar de esta evolución tecnológica, igualmente ellos eran alcanzados desde cualquier punto de la aplicación; continuaban siendo una "masa de datos".

Observando la tendencia de esta evolución, no cabe menos que esperar que los métodos de diseño y luego de análisis, se preocuparan, ante todo, en brindar una solución a los procedimientos.

Inicialmente no existían los métodos de análisis. El desarrollo de sistemas consistía en construir directamente los códigos expresados en lenguaje de máquina. Con el advenimiento de los lenguajes que separaban al programador del hardware, comienzan a escribirse los programas que, si bien entonces eran empleados sólo por las organizaciones en donde se desarrollaban, con el pasar del tiempo se constituirían en productos con una distribución más amplia.

2. Evolución de los métodos de análisis orientado a procesos

En esta época inicial, el análisis consistía en determinar los requisitos del sistema con el empleo del **lenguaje natural**. Luego se escribía el código y casi no se documentaba. Incluso muchas veces, las recomendaciones acerca de los métodos que podían emplearse, provenían de las mismas empresas proveedoras del equipamiento, y que entrenaban luego al personal de sus clientes para que pudieran realizar de una forma metódica el desarrollo de los sistemas.

No es necesario explicar los problemas que se presentaban a la hora de interpretar las largas especificaciones de requisitos. Pueden consultarse muchos autores, en especial aquellos pertenecientes al ámbito del análisis estructurado, como para comprender cabalmente la problemática de este método que se conoce como especificación funcional³.

Una forma lógica para tratar de disminuir la complejidad de esta especificación de requisitos, consistía en describir el sistema descomponiendo las funciones principales en subfunciones, de allí su denominación, **descomposición funcional**⁴, en un claro **enfoque top-down**.

A fines de la década del sesenta aparecieron los métodos de **análisis de sistemas estructurado**. Ya se había producido la famosa **crisis del software** que hizo replantear los

³ YOURDON, Edward. Análisis Estructurado Moderno. México, Prentice-Hall Iberoamericana, 1989, p.138.

⁴ COAD, Peter y Yourdon, Edward. Op.cit., p.20.



Análisis de Sistemas con UML para Estudiantes
Versión 1.1 – Diciembre de 2004

métodos de programación. En una época en la que no existía el abrumador cambio tecnológico actual, y menos en el ámbito de los lenguajes, no se contaba con muchos caminos a seguir. Había que mejorar las técnicas de desarrollo. La solución no podía esperarse de parte de las empresas desarrolladoras de los lenguajes de programación, como sucede en la actualidad, sino que la única alternativa consistía en optimizar los métodos de construcción de software existentes. Con la aparición de las técnicas estructuradas pareció que, en un principio, se iban a terminar estos problemas, y si bien no se solucionaron por completo, se redujeron en algunos casos. Este enfoque se trasladó paulatinamente al diseño y, por último, al análisis.

El análisis estructurado tiene como máximos exponentes a Tom de Marco, Yourdon, Constantine, y Gane y Sarson. Sus métodos fueron muy difundidos durante la década del setenta, y empleados con éxito en un sinnúmero de desarrollos de aplicaciones. Pero no todas las veces se llegó a buen puerto, pues no se encontró una real solución; "el diseño y la programación estructurados les permitían a algunos equipos de encargados de proyectos llegar a un desastre más rápidamente que antes, al construir una brillante solución al problema equivocado"⁵. Es que el hecho de contar con un método no garantiza en absoluto el logro satisfactorio de los resultados.

La herramienta principal de estos métodos es el **Diagrama de Flujo de Datos**, también conocido como **DFD**, motivo por el que también se denomina a estos métodos enfoque de flujo de datos⁶. Dicho por los mismos autores, "el diagrama de flujo de datos en el nivel lógico es la herramienta clave para comprender y trabajar con un sistema de cualquier complejidad, junto con el refinamiento de la notación para su uso en el análisis"⁷. Si bien Gane y Sarson habían depurado la notación utilizada hasta entonces, logrando una mejor apariencia de los diagramas y una amplia difusión de los símbolos empleados, se mantenían en la línea del análisis estructurado tradicional.

Las herramientas de análisis que propone el método de estos autores son, en primer término, el DFD, del que se nutren las restantes y que se puede expresar con diferentes niveles de detalle. En segundo término establecen el empleo de un Diccionario de Datos, en el que se detallan todos los elementos del DFD, los datos que contienen los almacenamientos y los que circulan a lo largo de los flujos de datos, y la descripción de la lógica de los procesos involucrados. También incluyen los Diagramas de Acceso Inmediato a los Datos, que definen las diferentes alternativas de acceso que posee el usuario para elaborar sus consultas, como así también el contenido de los almacenamientos. Puede elaborarse un Diagrama de Flujo de Materiales para mostrar la contraparte física de los DFD, ya que estos últimos se deben mantener en un nivel lógico apropiado.

⁵ YOURDON, Edward. Op.cit. p.140.

⁶ COAD, Peter y Yourdon, Edward. Op.cit. p.22.

⁷ GANE, Chris y Sarson, Trish. Análisis Estructurado de Sistemas. Buenos Aires, El Ateneo, 1988, p.26. El destacado es mío.



Si hacemos un repaso de lo expuesto hasta el momento, podremos observar una tendencia principal durante la evolución de los métodos. El mayor énfasis está puesto en los procesos. Tanto en los lenguajes de programación como en los métodos de análisis vemos que la mayor atención está centrada en lo que el sistema debe hacer. La mayor preocupación está en definir la estructura de procesos del sistema, muchas veces de manera compleja. Por eso mismo podemos hablar de una **orientación a los procesos**, de métodos orientados a procesos. Hoy se los caracteriza también como **orientados a funciones**⁸.

3. Evolución de los métodos de análisis orientado a datos

Pero, paralelamente a este desarrollo, se produce el advenimiento de las **bases de datos relacionales**. Durante los setenta, este modelo, basado en la **teoría relacional** de Codd, de IBM, comenzaba a prevalecer sobre el jerárquico y el de red. Si bien su comercialización y aplicación masiva recién se dio durante la década de los ochenta, "la flexibilidad de sus almacenamientos de datos y la sencillez de los lenguajes de programación que las acompañaban, sedujeron rápidamente a legiones de programadores, cuyas filas se vieron engrosadas por muchos disidentes de los fieles del lenguaje COBOL, cansados de pasar horas convirtiendo archivos y de perseguir el flujo del control a través de largas sábanas de código fuente"⁹.

El **modelado de la información** aparece durante la década de los setenta. Su principal exponente es Chen, quien logra imponer el diagrama de entidad-relación como herramienta estándar de esta corriente.

Fueron propuestos muchos otros métodos para la construcción de modelos basados en datos, entre los que cuentan los pertenecientes a Jackson y a Warnier y Orr. Esta corriente se conoce como **orientación a los datos**, o también **orientación a la estructura de los datos**.

⁸ IEEE. Guide to the Software Engineering Body of Knowledge. Trial versión 1.00. Mayo de 2001, p.3-2.

⁹ PINCIROLI, Fernando. El Análisis de Sistemas Orientado a Objetos: Actualidad y Perspectivas Futuras. En: Análisis Universitario, Mendoza, Facultad de Ciencias Económicas "San Francisco", anexa a la Universidad Católica Argentina, nº 10, dic 1995. p.4.



4. El enfoque de datos se impone sobre el enfoque de procesos

Los nombres de estas metodologías opuestas entre sí -modelado haciendo énfasis en los datos o en la representación de los procesos- dan idea de dos conceptos diferentes: mientras que orientación al flujo de datos pone de manifiesto que se trata de un enfoque dinámico, la orientación a la estructura de los datos presenta una **visión estática** del modelado de sistemas.

Muchas fueron las discusiones acerca de cuál de esta dos posturas era la más adecuada para el modelado de un sistema. Por una serie de razones -entre ellas el hecho de que la orientación a procesos fuera más antigua; el que las bases de datos no estuvieran difundidas de manera apropiada y que prevalecieran las aplicaciones en COBOL o en lenguajes nativos de los distintos equipamientos; el que no existieran las estructuras suficientes para capacitar y entrenar a la gente de sistemas en nuevas metodologías; el que la resistencia al cambio fuera mayor debido a que todo el conocimiento que se tenía brindaba más espacio de poder; y muchas más-, la decisión de los encargados de proyectos se veía perturbada. Por estos motivos enunciados, hasta ese momento "las metodologías de ingeniería de software más ampliamente usadas son aquellas basadas en diagramas de flujo de datos"¹⁰, o como expresa Booch, "el método de análisis más popular hoy es el análisis estructurado... es bien conocido, mucha gente está entrenada en sus técnicas, y muchas herramientas soportan su notación"¹¹.

Las bases de datos tuvieron que demostrar durante mucho tiempo las ventajas de su gran flexibilidad como para que fueran admitidas realmente, aunque poco a poco fueron ganando su espacio, al punto tal que pasaron a constituir una importante alternativa a los métodos tradicionales orientados a procesos y lograron equilibrar su aceptación: "la mayoría de los métodos que se usan en la industria hoy, para desarrollo de sistemas de información y técnicos, se basan en una descomposición funcional o dirigida a los datos del sistema"¹². Muchos metodólogos no tuvieron otra opción que reconocer esta realidad, como en el caso de Page-Jones, quien comenta: "ocho años han pasado desde que escribí la primera edición de este libro. Durante este tiempo ha sucedido mucho en el campo del desarrollo del software. Por ejemplo, hemos visto el advenimiento de la edad de la modelación de la información"¹³.

¹⁰ RUMBAUGH, James y otros. Object-Oriented Modeling and Design. Englewood Cliffs, Prentice-Hall, 1991, p.266. La traducción es mía.

¹¹ BOOCH, Grady. Object-Oriented Design with Applications. Redwood City, Benjamin/Cummings, 1991, p.201. La traducción es mía.

¹² JACOBSON, Ivar y otros. Op.cit. p.75. El destacado es mío.

¹³ PAGE-JONES, Meilir. The Practical Guide to Structured Systems Design. 2ª ed. Englewood Cliffs, Prentice-Hall, 1988, p.XV. La traducción y el destacado son míos.



5. Ventajas del enfoque de datos sobre el enfoque de procesos

No obstante el rumbo que siguió la historia, y aun cuando existen opiniones opuestas¹⁴, a mi criterio los métodos orientados a datos son más adecuados, pues permiten modelar la realidad a partir de los **elementos estáticos** que se encuentren en ella, lo que posibilita la determinación de la estructura presente en la realidad. Sentar las bases de un sistema en su estructura es mucho más firme que hacerlo sobre elementos que varían con mayor rapidez a través del tiempo, como sucede con los procesos. Como muy bien expone Bertrand Meyer, un buen método de diseño es aquel que "produce arquitecturas que no necesitan cambiarse abruptamente para cada pequeño cambio en los requerimientos del sistema"¹⁵, y agrega que "si la arquitectura se basa principalmente en las funciones, parece imposible asegurar que la evolución del sistema será tan suave como la evolución de los requerimientos"¹⁶, mientras que un programa "siempre manipulará más o menos los mismos órdenes de datos"¹⁷.

Pero más allá de toda discusión, nadie podía negar una realidad determinante. En los sistemas hay que tener en cuenta tanto los procesos como los datos. Tolerar uno en desmedro del otro adolece de un grave problema, el hecho de quedar expuesto cuando la solución del sistema pasa por la consideración de los datos y se cuenta con herramientas que modelan sólo procesos, o viceversa. Es así que, poco a poco, la buena voluntad de los metodologistas hizo que se fueran encontrando puntos de contacto entre ambas corrientes, y así se puede observar el empleo de herramientas comunes en los distintos métodos que siguieron apareciendo.

6. Los autores reconocen sus errores: ambos son necesarios, datos y procesos

Sin lugar a dudas, Gane y Sarson debían dar solución a la deficiente situación en la que se encontraban los almacenamientos de datos al ser considerados principalmente como sumideros de los elementos de datos que transportan los flujos de datos. Eligieron la teoría relacional de Codd y así, por medio de la normalización, mejoraron la presentación de los datos en su método, tan cuidadosamente pulido para el modelado de los procesos del sistema.

Pero los métodos orientados a procesos que aparecen durante la década de los ochenta dan un paso más en la dirección de incorporar aspectos del enfoque de datos. Incluso Gane y Sarson

¹⁴ Cfr. ParcPlace. MethodWorks User's Guide. Sunnyvale, ParcPlace Systems, Inc., 1995. p. 1.

¹⁵ MEYER, Bertrand. Object-Oriented Software Construction. Hertfordshire, Prentice-Hall, 1988, p.42. La traducción es mía.

¹⁶ Ibidem

¹⁷ Ibidem



Análisis de Sistemas con UML para Estudiantes
Versión 1.1 – Diciembre de 2004

habían anunciado que esto sería así, al decir que "la normalización se hará probablemente más y más importante en los próximos años, a medida que las bases de datos se vuelvan más complejas y la simplicidad se haga más vital"¹⁸, afirmación que puede ser extendida no sólo a la normalización, sino también a todos los conceptos del ámbito de las bases de datos relacionales.

McMenamin y Palmer también consideraron muy necesario incorporar los elementos provenientes de la orientación a datos, aspectos que, según su opinión, no habían sido tomados muy en cuenta por los autores de métodos orientados a procesos. Para ellos, "los analistas deben por consiguiente usar un tema diferente, uno denominado particionado en objetos"¹⁹. Contrariamente a lo que sucedía una década atrás, ya no se podían negar los resultados de la aplicación de los métodos orientados a datos. Estos autores reconocen que tardaron mucho tiempo en aceptar los beneficios del enfoque de datos: "el concepto de subdivisión en objetos, o análisis de entidad-relación, es severamente nuevo para el análisis estructurado, y existe como parte de varias técnicas de desarrollo de sistemas desde los sesenta"²⁰. Y más aún, reconocen las ventajas sobre la importancia del empleo de la orientación a datos: permite elaborar un modelo de datos más conciso, es independiente de las limitaciones que impone una implementación física y minimiza la repetición de almacenamientos de datos²¹.

En una época en donde las bases de datos proliferaban por todas partes, ya era imposible negar la necesidad de un modelado de datos apropiado. Aumentaba la probabilidad de tener que dar solución a problemas en donde la porción de datos debe ser modelada como factor principal en la estructura del sistema. Es así, entonces, que Yourdon incorpora en su método orientado a procesos los diagramas de entidad-relación y lo justifica diciendo, entre otras cosas, "¿Qué tal si el propósito del sistema no es hacer algo, sino meramente ser el recipiente de una gran cantidad de información interesante?"²².

Por el lado de la orientación a los datos, también se pueden observar avances hacia el modelado de procesos. Michael Jackson pasó de su método conocido como programación estructurada de Jackson, a un enfoque en donde suma, a la estructura de datos del sistema, los procesos que manipulan estos datos, método conocido como desarrollo de sistemas de Jackson²³. Incluso es posible considerarlo como un método basado en objetos.

¹⁸ GANE, Chris y Sarson, Trish. Op.cit. p. 118.

¹⁹ McMENAMIN, Stephen y Palmer, John. Op.cit. p.57. El término objeto representa, para estos autores, un concepto similar al de entidad de los métodos orientados a datos. De ninguna manera corresponde al concepto de objeto de la orientación a objetos. El destacado es mío.

²⁰ Ibidem. El destacado es mío.

²¹ Ibidem p.66.

²² YOURDON, Edward. Op.cit. p.261. El destacado es del autor.

²³ Cfr. PRESSMAN, Roger. Op.cit. p.450.



En él, se deben identificar inicialmente todas las entidades del dominio del problema, para luego establecer las acciones que corresponden a cada una de estas entidades, ordenadas temporalmente, describiendo el ciclo de vida, y representados por medio de unos diagramas conocidos como diagramas de Jackson. En definitiva, de la estructura de los datos, derivará la funcionalidad del sistema.

No cabe la menor duda, los métodos de análisis de sistemas deben poseer un **enfoque complementario de datos y procesos**, y atender con firmeza a ambos aspectos. La evolución de los métodos de análisis de sistemas a lo largo de la historia así lo demuestra.

7. Los datos y procesos son necesarios, pero no suficientes: la realidad está constituida por objetos

Llegados a la conclusión de la obvia necesidad de datos y procesos, como lo demuestran las tendencias y las palabras de los metodologistas, surge un nuevo inconveniente. Si bien entre los métodos orientados a procesos y orientados a datos, son preferibles estos últimos, igualmente no son totalmente adecuados, ya que ellos también realizan una visión parcializada de la realidad.

Tanto una corriente como la otra, debe "limpiar" la realidad de los elementos que no le son útiles, es decir, el analista se debe concentrar sólo en los procesos o sólo en los datos, según sea el enfoque elegido. Y así la representación de la realidad resulta incompleta.

Si aseguramos, entonces, que es fundamental contar con ambos enfoques -modelado de datos y de procesos en forma conjunta-, una solución que podría proponerse sería, por lo tanto, unir en un mismo método las principales herramientas de ambas corrientes. Esto no sería una verdadera solución -de hecho es lo que se puede encontrar en el método de Yourdon mencionado en el punto anterior- puesto que, más allá de que se consideraran con la misma importancia los datos y los procesos, seguirían siendo modelados separadamente; estos dos aspectos seguirían permaneciendo inconexos.

Es que el error está en que no se deben buscar procesos y datos en la realidad, puesto que es imposible encontrarlos aisladamente como tales. La realidad es diferente, "el mundo real está compuesto por objetos y no por datos o procesos independientes"²⁴. Esto es una verdad de orden filosófico ya que incluso "el Angélico [Santo Tomás de Aquino] considera que el número,

²⁴ TAKAHASHI, Tadao. Introducción a Programación Orientada a Objetos. Curitiba, III EBAI, 1988, p.4. La traducción es mía.



considerado abstractamente, no existe fuera de nuestra mente"²⁵, y el mismo Aristóteles afirma que existen cosas "que no se pueden separar, a los objetos en movimiento, por ejemplo"²⁶ puesto que "hay una multitud de accidentes que son esenciales a las cosas, en tanto que cada uno de ellos reside esencialmente en ellas"²⁷.

El **paradigma de objetos** no sólo permite modelar los elementos que existen en la realidad, y exactamente en la forma en la que se encuentran en ella, sino que ya incluye un enfoque de datos y de procesos a la vez. Al representar un objeto se están modelando procesos - correspondientes al comportamiento del objeto- y datos -que provienen de las características de los objetos-. Estos datos y procesos no se encuentran desunidos, sino absolutamente relacionados entre sí por ser parte esencial de una unidad conceptual, el objeto, como establece Aristóteles. No debe confundirse, entonces, el enfoque de objetos como la suma aritmética de los datos y de los procesos, sino, como la **suma sinérgica** de estos aspectos a causa de un estudio de la realidad totalmente integrador. Al considerar el comportamiento y las características de un objeto, no sólo se logra un análisis completo, sino que se está obteniendo un plus que corresponde a la relación que poseen esos elementos por ser parte de una única entidad del mundo real.

Además de tratarse de un enfoque diferente, se corresponde completamente con la forma en la que el hombre logra el conocimiento, pues interpreta la realidad de una manera ontológica. El hombre "no sólo percibe las cosas exteriores..., sino que las capta como tales, como objeto o realidad distinta de la suya, como un ser distinto, puesto delante de su propio ser"²⁸.

8. Nace la orientación a objetos

Al igual que en las otras metodologías, la **orientación a objetos** también comenzó primero en la programación, luego en el diseño y por fin en el análisis.

A fines de los sesenta, Alan Kay -a quien muchos atribuyen el establecimiento de la denominación orientación a objetos-, comenzó junto a Adele Goldberg y Daniel Ingalls en XEROX PARC (Palo Alto Research Center), el desarrollo de Smalltalk, un lenguaje concebido para un proyecto denominado DYNABOOK, "que podría ser muy groseramente descrito como un computador personal en la franja de procesamiento de un micro de 32 bits, la funcionalidad

²⁵ BALLESTEROS, Juan Carlos Pablo. La Filosofía del Padre Castellani. Buenos Aires, Gladius, 1990. p.24.

²⁶ ARISTÓTELES. Op.cit. p.284.

²⁷ Ibidem

²⁸ DERISI, Octavio. Filosofía de la Cultura y de los Valores. Buenos Aires, Emecé, 1963, p.40. El destacado es del autor.



Análisis de Sistemas con UML para Estudiantes
Versión 1.1 – Diciembre de 2004

de un McIntosh, y el tamaño de un libro"²⁹. Este lenguaje, destinado principalmente a representar conceptos, todavía hoy es el lenguaje orientado a objetos por excelencia.

Junto con otros aportes, especialmente de LISP (1958) y de SIMULA (1967), **Smalltalk** fue el primero de una serie de lenguajes que comenzaron a aparecer soportando el modelo de objetos.

El primer método de diseño se debe a Grady Booch. Fue presentado en 1986 y apuntaba al empleo del lenguaje Ada que, curiosamente, no es orientado a objetos, sino basado en objetos. Luego fue mejorando el método a lo largo de versiones posteriores -la más difundida es la que corresponde al año 1991-, y hoy existe, por un lado, su edición de análisis y diseño (1994), y junto a Rumbaugh y Jacobson, otros trabajos sobre la definición de un lenguaje de modelación. Poco a poco siguieron apareciendo otros métodos de diseño, entre los que se encuentran, principalmente, un método de análisis lingüístico de la especificación en lenguaje natural, propuesto por Abbott en 1983; el método General de Diseño Orientado a Objetos (General Object-Oriented Design - GOOD), también aparecido en 1983; y el método de Diseño Jerárquico Orientado a Objetos (Hierarchical Object-Oriented Design - HOOD), publicado en 1989.

En cuanto al análisis orientado a objetos, en 1988 aparece el primer libro cuyo título da a entender que se trata de este tema -"Análisis Orientado a Objetos: Modelando el mundo en datos"-, con autoría de Sally Shlaer y Stephen Mellor, y aunque no puede considerarse estrictamente como orientación a objetos, puesto que apuntaba más al modelado de datos que de objetos, es un libro muy importante en el desarrollo de esta disciplina por ser uno de los primeros. Posteriormente, en 1991, estos autores presentaron un nuevo método ya orientado a objetos, y durante un tiempo se mantuvieron a la vanguardia, con su nueva versión denominada OOA/RD.

Un trabajo que había sido presentado por Ivar Jacobson en OOPSLA '87, bajo el título "Object-Oriented Development in an Industrial Environment", podría considerarse como el primer método orientado a objetos propuesto. Coad y Yourdon irrumpen con su método en 1990 y su revisión en 1991. A partir de este momento comenzaron a difundirse muchos otros, quizás, más de los que deberían haberse publicado.

²⁹ TAKAHASHI, Tadao. Op.cit. p.21.



9. Cómo estar seguros de haber aprendido bien el tema

a) Aprender los conceptos claves

Lo habremos logrado si estamos en condiciones de explicar **con nuestras propias palabras**, sin consultar el material escrito, los siguientes conceptos:

- Lenguajes de primera generación
- Lenguajes de segunda generación
- Lenguajes de tercera generación
- Requisitos del sistema en lenguaje natural
- Descomposición funcional
- Enfoque top-down
- Análisis de sistemas estructurado
- Crisis del software
- Diagrama de Flujo de Datos (DFD)
- Orientación a procesos
- Orientación a funciones
- Bases de datos relacionales
- Teoría relacional
- Modelado de la información
- Orientación a datos
- Orientación a la estructura de los datos
- Visión dinámica del sistema
- Visión estática del sistema
- Ventajas del enfoque estático sobre el dinámico
- Necesidad de un enfoque complementario de datos y procesos
- Paradigma de objetos
- Enfoque sinérgico de datos y procesos
- Orientación a objetos
- Smalltalk

Si no podemos explicarnos, deberíamos remitirnos nuevamente a la lectura del capítulo.

b) Lectura de textos

Es fundamental conocer los distintos métodos y autores, por lo que deberíamos, al menos, recorrer los textos de los autores de métodos tradicionales, con la finalidad de:

- ampliar los conocimientos adquiridos sobre el tema



Análisis de Sistemas con UML para Estudiantes
Versión 1.1 – Diciembre de 2004

- ampliar nuestro vocabulario y hacerlo más técnico y preciso
- enfrentarnos a términos más complejos que nos ayuden a familiarizarnos con ellos para así no tener dificultad en lecturas de otros libros o material que se nos presente más adelante
- mejorar nuestra lectura y comprensión de material técnico en inglés
- plantear al profesor las inquietudes que pudieran surgir de tales lecturas

c) Discutir con los compañeros

Es importante que se discutan los temas con los propios compañeros a fin de que cada uno se convenza más de lo que dice o se lo replantee al tener que explicarlo a un compañero.

De la misma manera, sería muy oportuno el intentar plantearse mutuamente situaciones que cada uno deba resolver, tratando de “poner en aprietos” al compañero con la complejidad de las preguntas. Así se logrará familiaridad con el tema y mayor agudeza en los razonamientos.

10. Bibliografía recomendada para conocer a los autores

Recomendamos, al menos, la revisión de los textos que aparecen en las citas. Sería conveniente releer la parte del capítulo en la que aparece la cita y, a la luz de esa lectura, analizar al autor citado, tratando de encontrar los fundamentos planteados en la lectura o de discutirlos si se considera que nuestra apreciación no fue lo suficientemente acertada.