

Sistemas, Software y Arquitectura

Una Perspectiva del Estado del Arte

Dario Palminio, David Alderete

2015

Índice

1	Introducción	9
1.1	Título de la sección primera	9
1.1.1	Título de la sub-sección primera	9
2	Sistemas	11
2.1	Introducción al sistemismo	12
2.2	Noción de Agregado	12
2.3	Noción de Sistema	12
2.3.1	Estructura	12
2.3.2	Organización	12
2.3.3	Evolución	12
2.4	Noción de niveles	12
2.5	Sistemización	12
2.5.1	Sistema mecánico	12
2.5.2	Sistema Orgánico	12
2.5.3	Sistemisidad	12
2.5.4	Acoplamiento	12
2.5.5	Cohesión	12
2.6	Noción de emergentismo	12
2.6.1	El concepto de emergencia	12
2.6.2	Características constitutivas	12
2.6.3	Emergencia de Sistemas	12
2.6.4	Inteligencia de enjambre	12
2.7	Cibernética	12
3	Modelado de Sistemas	13
3.1	Sistema de modelado	13
3.2	Herramientas conceptuales de modelado	14
3.2.1	Herramientas conceptuales de Diagramado	14
3.3	Modelos de un sistema general	16
3.4	Sistema general de modelado	16
3.5	Modelo de sistemas generales básicos	16
3.5.1	Sistemas termodinámicos	16

3.5.2	Sistema aislado en equilibrio termodinámico	16
3.5.3	Sistema aislado tendiente al equilibrio	16
3.5.4	Sistema abierto en equilibrio estacionario	16
3.5.5	Sistema Sistemas en Equilibrio Dinámico	16
3.5.6	Sistemas Sostenibles y Sostenibilidad	16
3.5.7	Sistema Mínimo de Vida	16
3.5.8	Sistema Agente	16
3.5.9	Sistema Autopoiético Mínimo	16
3.6	Modelos de Sistemas Cibernético	16
3.6.1	Sistema Cibernético General	16
3.6.2	Modelos de organizaciones	16
4	Ingeniería	17
4.1	Título de la sección primera	17
4.1.1	Título de la sub-sección primera	17
5	Sistemas Software	19
5.1	Título de la sección primera	19
5.1.1	Título de la sub-sección primera	19
6	Sistema de Ideas	21
6.1	Modelos mentales	21
6.2	Principios	21
6.2.1	Principio de Simplicidad	22
6.2.2	Principios Sistémicos	22
6.2.3	Principios Lean	22
6.2.4	Manifiesto Ágil	22
6.2.5	Principios Scrum	22
6.2.6	Manifiesto Craftsmanship	22
6.2.7	Principios POO	22
6.2.8	DRY	22
6.2.9	Manifiesto SOA	22
6.3	Paradigmas	22
6.3.1	Paradigma Estructurado	22
6.3.2	Paradigma Orientado a Objetos	22
6.3.3	Paradigma Orientado a Servicio	22
6.4	Filosofías	22
7	Metodologías	23
7.1	Título de la sección primera	23
7.1.1	Título de la sub-sección primera	23

8	Arquitectura	25
8.1	Generalidades	25
8.1.1	Arquitectura de Software	26
8.1.2	Arquitectura de Negocio	26
8.1.3	Arquitectura de Sistemas	26
8.1.4	Arquitectura de Sistemas de Sistemas	26
8.1.5	Objetivo de la Arquitectura	26
8.1.6	Necesidad de Arquitectura de Sistemas	26
8.1.7	Necesidad de Arquitectura de Software	26
8.1.8	Arquitectura como Arte	26
8.1.9	Arquitectura como Ciencia	26
8.1.10	Arquitectura como Tecnología	26
8.1.11	Arquitectura como Ingeniería	26
8.1.12	Arquitectura como filosofía	26
8.1.13	Clarificando la dualidad del término Arquitectura	26
8.1.14	Modelado Arquitectónico	26
8.1.15	Vista Arquitectónica	26
8.1.16	Niveles de abstracción	26
8.1.17	Diagramación Arquitectural	26
8.1.18	Diseño Arquitectónico	26
8.1.19	Estilos y Patrones Arquitectónicos	26
8.1.20	Evaluación Arquitectónica	26
8.2	Arquitectura de Sistemas	26
8.2.1	Título de la sub-sección primera	27
8.3	Arquitectura de Software	27
8.3.1	Título de la sub-sección primera	27
9	Glosario y Acrónimos	29

List of Figures

1.1	Ejemplo de imagen	9
3.1	DFD Notación Gane-Sarson [Gane/Sarson, 1979]	15
3.2	DFD Notación DeMarco-Yourdon [Dixit, 2007]	15

Capítulo 1

Introducción

Texto del Capítulo 1



Figura 1.1: Ejemplo de imagen

1.1 Título de la sección primera

Texto de la sección 1

1.1.1 Título de la sub-sección primera

Texto de la sub-sección 1

Capítulo 2

Sistemas

Texto del Capítulo 2

2.1 Introducción al sistemismo

2.2 Noción de Agregado

2.3 Noción de Sistema

2.3.1 Estructura

2.3.2 Organización

2.3.3 Evolución

2.4 Noción de niveles

2.5 Sistemización

2.5.1 Sistema mecánico

2.5.2 Sistema Orgánico

2.5.3 Sistemisidad

2.5.4 Acoplamiento

2.5.5 Cohesión

2.6 Noción de emergentismo

2.6.1 El concepto de emergencia

Emergencia en niveles

Definición de emergencia

Las cosas que emergen

2.6.2 Características constitutivas

2.6.3 Emergencia de Sistemas

2.6.4 Inteligencia de enjambre

Inteligencia de la multitud

2.7 Cibernética

Capítulo 3

Modelado de Sistemas

El Modelado de Sistemas es una actividad esencial de Ingeniería de Sistemas y de cualquier pensador sistémico. Pues, se hacen modelos como simplificación abstracta de la realidad [Meadows, 2009] que forma una representación del sistema real [Fiuba, 2005] o sistema origen a estudiar.

3.1 Sistema de modelado

Es que para entender la realidad se hace necesario percibirla y modelarla de algún modo. En este proceso existe un orden real de las cosas y un orden percibido y reflejado en un modelo por alguien, un observador. Como dice Capra “toda la percepción de un modelo es, de alguna manera, la percepción de algún orden” [Fritjof Capra, 1975] y la percepción de un orden es hecha por alguien. Un principio cibernético dice que todo fenómeno observado es observado por alguien (Heinz Von Foerster, 1979) que en otras palabras quiere decir que “todo conocer depende de la estructura del que conoce” [Maturana, 1984]. Por este motivo en el sistema de modelado existe el observador como un elemento principal del mismo.

Table 3.1: Herramientas conceptuales de Diagramado

Acrónimo	Significado en inglés
ADL	Architecture Description Language
BPMN	Business Process Modeling Notation
CD	Conceptual Diagram or ConceptDraw
CLD	Causal Loop Diagram
ERD	Entity Relationship Diagram
FC	Flow Charts (for control flow)
DFD	Data Flow Diagram
MMD	Map Mind Diagram
SC	Structure Chart
SFD	Stock and Flow Diagrams
SSADM	Structured Systems Analysis and Design Method
UML	Unified Modeling Language

3.2 Herramientas conceptuales de modelado

3.2.1 Herramientas conceptuales de Diagramado

ADL

BPMN

CD

CLD

ERD

FC

El Diagrama de Flujo (Flow Charts) es usado para diagramar flujos de control.

DFD

El Diagrama de Flujo de Datos o DFD fue introducido y popularizado en 1970 para el análisis y diseño estructurado [Gane/Sarson, 1979] para diagramar procesos de sistemas. Un DFD muestran el flujo de datos dentro del sistema desde entidades externas, mostrando como los datos fluyen entre diferentes procesos y qué almacenes intervienen para que los datos sean guardados y recuperados [Scott Ambler, 2004]. Para este tipo de diagramas hay dos estándares de notaciones: la notación DeMarco-Yourdon (DeMarco, Yourdon y Constantine 1979 [Dixit, 2007]) y la notación Gane-Sarson.

En este tipo de diagramas hay solo cuatro elementos [Dixit, 2007]:

- **Entidad externa:** son fuentes externas. En ambas notaciones se representa con cuadrados.
- **Proceso:** son los subsistemas o procesos del sistema. En notación Gane-Sarson se representa con rectángulos redondeados y según DeMarco-Yourdon con círculos o elipses.
- **Flujo de dato:** flujo de datos electrónicos o físicos. En ambas notaciones se representa con flechas.
- **Almacen:** almacén de información. En notación Gane-Sarson se representa con rectángulos abiertos y según DeMarco-Yourdon con rectángulos sin líneas laterales. Por practicidad se puede implementar con un rectángulo con algún distintivo, siempre cuando se aclare que la notación representa un almacén de datos.

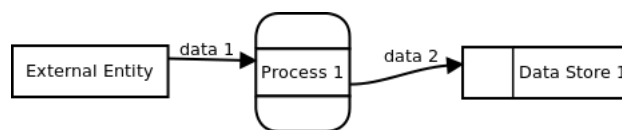


Figura 3.1: DFD Notación Gane-Sarson [Gane/Sarson, 1979]



Figura 3.2: DFD Notación DeMarco-Yourdon [Dixit, 2007]

MMD

SC

SFD

SSADM

UML

3.3 Modelos de un sistema general

3.4 Sistema general de modelado

3.5 Modelo de sistemas generales básicos

3.5.1 Sistemas termodinámicos

3.5.2 Sistema aislado en equilibrio termodinámico

3.5.3 Sistema aislado tendiente al equilibrio

3.5.4 Sistema abierto en equilibrio estacionario

3.5.5 Sistema Sistemas en Equilibrio Dinámico

3.5.6 Sistemas Sostenibles y Sostenibilidad

3.5.7 Sistema Mínimo de Vida

3.5.8 Sistema Agente

3.5.9 Sistema Autopoiético Mínimo

3.6 Modelos de Sistemas Cibernético

3.6.1 Sistema Cibernético General

Ejemplo 1: Sistema regulador de tanque de agua

Ejemplo 2: Sistema regulador de Watt

Sistema Cibernético Auto-Aprendiente

Sistema Cibernético Auto-Organizado

Sistema Cibernético Auto-Aprendiz y Auto-Organizado

3.6.2 Modelos de organizaciones

Viable System Model

Capítulo 4

Ingeniería

Texto del Capítulo 2

4.1 Título de la sección primera

Texto de la sección 1

4.1.1 Título de la sub-sección primera

Texto de la sub-sección 1

Capítulo 5

Sistemas Software

Texto del Capítulo 2

5.1 Título de la sección primera

Texto de la sección 1

5.1.1 Título de la sub-sección primera

Texto de la sub-sección 1

Capítulo 6

Sistema de Ideas

Texto del Sistema de Ideas

6.1 Modelos mentales

Texto de la sección Modelos mentales

6.2 Principios

Texto de la sección Principios

6.2.1 Principio de Simplicidad

Principio de la Navaja de Ockham

Principio KISS

Principio de Simplicidad de Einstein

6.2.2 Principios Sistémicos

6.2.3 Principios Lean

6.2.4 Manifiesto Ágil

6.2.5 Principios Scrum

6.2.6 Manifiesto Craftsmanship

6.2.7 Principios POO

Principios SOLID

6.2.8 DRY

6.2.9 Manifiesto SOA

6.3 Paradigmas

Texto de la sección Paradigmas

6.3.1 Paradigma Estructurado

6.3.2 Paradigma Orientado a Objetos

6.3.3 Paradigma Orientado a Servicio

6.4 Filosofías

Texto de la sección Filosofías

Capítulo 7

Metodologías

Texto del Capítulo 2

7.1 Título de la sección primera

Texto de la sección 1

7.1.1 Título de la sub-sección primera

Texto de la sub-sección 1

Capítulo 8

Arquitectura

Generalidades, Arquitectura de Sistemas y Arquitectura de Software

8.1 Generalidades

Texto de la sección Generalidades

- 8.1.1 Arquitectura de Software
- 8.1.2 Arquitectura de Negocio
- 8.1.3 Arquitectura de Sistemas
- 8.1.4 Arquitectura de Sistemas de Sistemas
- 8.1.5 Objetivo de la Arquitectura
- 8.1.6 Necesidad de Arquitectura de Sistemas
- 8.1.7 Necesidad de Arquitectura de Software
- 8.1.8 Arquitectura como Arte
- 8.1.9 Arquitectura como Ciencia
- 8.1.10 Arquitectura como Tecnología
- 8.1.11 Arquitectura como Ingeniería
- 8.1.12 Arquitectura como filosofía
- 8.1.13 Clarificando la dualidad del término Arquitectura
 - Arquitectura como actividad
 - Arquitectura como propiedad
- 8.1.14 Modelado Arquitectónico
- 8.1.15 Vista Arquitectónica
- 8.1.16 Niveles de abstracción
- 8.1.17 Diagramación Arquitectural
- 8.1.18 Diseño Arquitectónico
- 8.1.19 Estilos y Patrones Arquitectónicos
 - Estilo Arquitectónico
 - Patrón Arquitectónico
- 8.1.20 Evaluación Arquitectónica
 - Mala Arquitectura
 - Buena ArquitecturaDRY
- 8.2 Arquitectura de Sistemas

Texto de la sección Arquitectura de Sistemas

8.2.1 Título de la sub-sección primera

Texto de la sub-sección 1

8.3 Arquitectura de Software

Texto de la sección Arquitectura de Software

8.3.1 Título de la sub-sección primera

Texto de la sub-sección 1

Capítulo 9

Glosario y Acrónimos

Nombre	Detalle
ADL	Architecture Description Language
BPMN	Business Process Modeling Notation
CD	Conceptual Diagram or ConceptDraw
CLD	Causal Loop Diagram
ERD	Entity Relationship Diagram
FC	Flow Chart Diagram for control flow
DFD	Data Flow Diagram
MMD	Map Mind Diagram
SAD	Software Architecture Documentation
SC	Structure Chart
SFD	Stock and Flow Diagrams
SSADM	Structured Systems Analysis and Design Method
SW	Software
UML	Unified Modeling Language

Ejemplos de referencias: [AGdE, 2004] [Albin, 2003] [Andrés Mejía D., 2007] [Bass Clements Kazman, 1998] [BAWG, 2008] [Beck, 2001] [Bertalanffy, 1968] [Billy Koen, 2000] [BIZBOK, 2014] [Boehm, 1995] [Booch, 2007] [Bredemeyer, 2002] [Bunge, 1979] [Bunge, 2003] [Burd, 2011] [Buschmann, 1996] [Capretz, 1996] [CCIT, 2014] [Claude Perrault, 1673] [Clements Northrop, 1996] [Cohn, 2004] [Larman/Vodde, 2008] [Dan North, 2009] [Dan North, 2012] [Donella Meadows, 1982] [Donella Meadows, 2009] [DSA, 2002] [Edwards Deming, 2000] [Esterking, 2007] [Evans, 2004] [Ferguson, 2014] [Finkelstein, 1996] [Fiuba, 2005] [FOLDOC, 2015] [Francesco Milizia, 1781] [Giraldo, 2006] [PMBOK, 2004] [Maturana, 1988] [Ian Sommerville, 2006] [IEEE, 2000b] [IEEE, 1993] [IEEE, 1990] [IEEE, 2000a] [Joaquín Arnau, 2000] [Joaquín Arnau, 2000] [Kazman, 1996] [Beck, 2003] [Beck, 2005] [Kruchten, 1999] [Kurzweil, 2005] [Le Corbusier, 1923] [Malveau, 1997] [Malveau, 2004] [Marco Vitruvio, 25ac] [Martin Alaimo, 2014] [Martin Fowler, 1999] [Martin Fowler, 2002] [Meadows, 2009]

[Montrose, 1995] [Moore, 1975] [Cosmos, 2014] [Oxford, 2015] [Perry, 1997]
[Peter Senge, 1990] [Poppendieck, 2003] [Pressman, 2002] [Robert Martin, 2009]
[Giachetti, 2010] [Rozanski/Woods, 2012] [Ackoff, 1981] [Sarabia, 1995] [SBOK, 2013]
[SCEA, 2010] [Scott Bellware, 2008] [SEBoK, 2014] [SEI, 2003] [Shaw/Garlan, 1996]
[SLI, 2009] [SOLID, 2009] [Stafford Beer, 1977] [Stephen Burd, 2011] [Stephen Hawking, 2010]
[SWEBOK, 2014] [UBA, 2015] [UIT, 2014] [UNTREF, 2014] [US, 2015] [Velthuis/Parra, 2007]
[Vilana Arto, 2011] [Wirf-Brock/McKean, 2015] [Scott Ambler, 2004] [Gane/Sarson, 1979]
[Dixit, 2007]

Bibliografía

- [Ackoff, 1981] Ackoff (1981). *Creating the Corporate Future: plan or be planned for. By Russell Ackoff*. México: Editorial Limusa, 1995.
- [AGdE, 2004] AGdE (2004). *Arquitectura de Software, Guía de estudio. Erika Camacho, Fabio Cardeso, Gabriel Nuñez, Revisada Por: Prof. Maria A. Perez De Ovalles, Prof. Anna Grimán, Prof. Luis E. Mendoza*. Compartida.
- [Albin, 2003] Albin, S. (2003). *The Art of Software Architecture: Design methods and techniques. By Albin Stephen*. Nueva York, Wiley.
- [Andrés Mejía D., 2007] Andrés Mejía D. (2007). *Tres esferas de acción del pensamiento crítico en ingeniería*. Universidad de Los Andes, Bogotá, Colombia. Revista Iberoamericana de Educación (ISSN: 1681-5653).
- [Bass Clements Kazman, 1998] Bass Clements Kazman (1998). *Software Architecture in Practice. By Len Bass, Paul Clements y Rick Kazman*. Reading, Addison-Wesley.
- [BAWG, 2008] BAWG (2008). *What Is Business Architecture? By Object Management Group OMG, Business Architecture Working Group*. Bawg.omg.org. URL: www.businessarchitectureinstitute.org.
- [Beck, 2001] Beck (2001). *Agile Manifesto By Beck, Kent*. URL: www.agilemanifesto.org, 2001, como estaba en Octubre de 2012.
- [Beck, 2003] Beck (2003). *Test-Driven Development By Example. Por Kent Beck*. Three Rivers Institute. Addison-Wesley Professional.
- [Beck, 2005] Beck (2005). *Extreme Programming Explained: Embrace Change, 2nd Edition. Por Kent Beck, Cynthia Andres*. Addison-Wesley. Completely rewritten, 2005.
- [Bertalanffy, 1968] Bertalanffy (1968). *Teoría General de los Sistemas. Por Ludwig Von Bertalanffy*. Fondo de cultura económica, México. Reimpresión Argentina, 2007 (1ra. edición en Ingles 1968). ISBN 987-950-557-096-6.

- [Billy Koen, 2000] Billy Koen (2000). *El método de Ingeniería. Por Koen V. Billy*. Facultad de Ingeniería, Santiago de Cali 2000.
- [BIZBOK, 2014] BIZBOK (2014). *Business Architecture Guild, A Guide to the Business Architecture Body of Knowledge. By BIZBOK Guide Version 2.0, 2012 and BIZBOK Guide Version 4.1. 2014*. BIZBOK.
- [Boehm, 1995] Boehm (1995). *Reasoning about the Composition of Heterogeneous Architecture. By Boehm, B., Abd-Allah, A.* USC Center for Software Engineering Technical Report. University of Southern California, Los Angeles.
- [Booch, 2007] Booch (2007). *The Irrelevance of Architecture. By Grady Booch, IBM*. IEEE, Software, May/June 2007 (Vol. 24, No. 3), 2007.
- [Bredemeyer, 2002] Bredemeyer (2002). *The Visual Architecting Process. By Bredemeyer, D., Malan, R.* White Paper.
- [Bunge, 1979] Bunge (1979). *Ontología II: Un mundo de sistemas. Tratado de filosofía. Por Mario Bunge*. Filosofía de la ciencia. Editorial Gedisa.
- [Bunge, 2003] Bunge (2003). *Emergencia y convergencia: novedad cualitativa y unidad del conocimiento. Por Mario Bunge*. Filosofía de la ciencia. Editorial Gedisa.
- [Burd, 2011] Burd (2011). *Systems Architecture, Sixth Edition. By Stephen D. Burd*. University of New Mexico. Course Technology, Cengage Learning.
- [Buschmann, 1996] Buschmann (1996). *Pattern - Oriented Software Architecture. A System of Patterns. By Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P., Stal, M.* John Wiley and Sons, Inglaterra.
- [Capretz, 1996] Capretz (1996). *Object-Oriented Software: Design. By Luiz Fernando Capretz*. 1996.
- [CCIT, 2014] CCIT (2014). *Coyuntura TIC Avances y retos de la defensa digital en Colombia. By Luiz Fernando Capretz*. Reporte TIC de CCIT elaborado por Fedesarrollo.
- [Claude Perrault, 1673] Claude Perrault (1673). *Les dix livres d'Architecture de Vitruve By Claude Perrault*. 1673.
- [Clements Northrop, 1996] Clements Northrop (1996). *Software architecture: An executive overview. By Paul Clements y Linda Northrop*. Technical Report, CMU/SEI-96-TR-003, ESC-TR-96-003. Febrero de 1996.
- [Cohn, 2004] Cohn (2004). *User Stories Applied: For Agile Software Development. By Mike Cohn*. Addison Wesley.

- [Cosmos, 2014] Cosmos (2014). *Cosmos: A SpaceTime Odyssey*. By Neil deGrasse Tyson, Livia Hanich, Steven Holtzman, Ann Druyan and Steven Soter. Cosmos Studios.
- [Dan North, 2009] Dan North (2009). *Agile specifications, BDD and Testing eXchange*. By Dan North. November 2009 in London.
- [Dan North, 2012] Dan North (2012). *Introducing BDD*. By Dan North. Retrieved 12 August 2012.
- [Dixit, 2007] Dixit (2007). *Structured System Analysis and Design*. By J.B. Dixit, Raj Kumar. Laxmi publications. First edition 2007.
- [Donella Meadows, 1982] Donella Meadows (1982). *Whole Earth Models and Systems. Co-Evolution Quarterly*. By Donella h. Meadows. 1982.
- [Donella Meadows, 2014] Donella Meadows (2014). *El Diccionario de la lengua española de la Real Academia Española (DRAE) es la obra de referencia de la Academia. La última edición es la 23*. Octubre de 2014.
- [DSA, 2002] DSA (2002). *Documenting Software Architecture. View and Beyond*. By Paul Clements, Felix Bachmann, Len Bass, David Garlan, James Ivers, Reed Little, Robert Nort and Judith Stafford. Addison Wesley. SEI Series in Software Engineering.
- [Edwards Deming, 2000] Edwards Deming (2000). *System of Profound Knowledge*. By Deming, W. Edwards. The New Economics for Industry, Government, Education, 2nd Edition, The MIT Press.
- [Esterking, 2007] Esterking (2007). *La administración de proyectos en un ámbito competitivo*. Por José D. Esterking. Thomson.
- [Evans, 2004] Evans (2004). *Domain-Driven Design - Tackling Complexity in the Heart of Software*. By Evans Eric. Addison-Wesley.
- [Ferguson, 2014] Ferguson (2014). *BDD in Action: Behavior-driven development for the whole software lifecycle*. By John Ferguson Smart. Paperback, Manning.
- [Finkelstein, 1996] Finkelstein (1996). *The viewpints FAQ, Software Engineering Journal*. By Finkelstein A. and Sommerville.
- [Fiuba, 2005] Fiuba (2005). *Sistemas, Modelos y Simulación. 75.26 – Simulación*. Profesor: Ing. Horacio Rocca. Fiuba Facultad de Ingeniería. Universidad de Buenos Aires. Calendario segundo cuatrimestre 2005.
- [FOLDOC, 2015] FOLDOC (2015). *Diccionario de computación FOLDOC*. Online.

- [Francesco Milizia, 1781] Francesco Milizia (1781). *Principi di Architettura Civile. Francesco Milizia*. Online.
- [Fritjof Capra, 1975] Fritjof Capra (1975). *El TAO de la Física. Una exploración de los paralelismos entre la física moderna y el misticismo oriental. Por Fritjof Capra*. Editorial Sirio. Edición original: 1975, 1983. 9na. edición 2007.
- [Gane/Sarson, 1979] Gane/Sarson (1979). *Structured systems analysis: tools and techniques. By Chris Gane, Trish Sarson*. Prentice-Hall.
- [Giachetti, 2010] Giachetti (2010). *Design of Enterprise Systems: Theory, Architecture, and Methods. By Ronald E. Giachetti*. Taylor and Francis Group.
- [Giraldo, 2006] Giraldo (2006). *Ciencia y método científico en ingeniería. Volumen 1-Año 2006. Por Alexander Pareja Giraldo*. Institución Universitaria Antonio José Camacho. ISSN: 1909-8510.
- [Ian Sommerville, 2006] Ian Sommerville (2006). *Ingeniería del software. Por Ian Sommerville*. Séptima edición, 2005 y 2006.
- [IEEE, 1990] IEEE (1990). *IEEE 610.12.1990*. IEEE.
- [IEEE, 1993] IEEE (1993). *ISO/IEC/IEEE Systems and Software Engineering Vocabulary*. IEEE.
- [IEEE, 2000a] IEEE (2000a). *IEEE Architecture Working Group*. IEEE.
- [IEEE, 2000b] IEEE (2000b). *IEEE Std 1471-2000*. IEEE.
- [Joaquín Arnau, 2000] Joaquín Arnau (2000). *72 voces para un diccionario de arquitectura teórica*. p. 176.
- [Kazman, 1996] Kazman (1996). *Tool Support for Architecture Analysis and Design. By Kazman, R.* Department of Computer Science, University of Waterloo.
- [Kruchten, 1999] Kruchten (1999). *The Rational Unified Process. By Kruchten, P.* Longman, Inc. Reading, MA: Addison Wesley Lane, T. (1990). Studying Soft, 1990.
- [Kurzweil, 2005] Kurzweil (2005). *The singularity is near. When humans transcend biology. By Ray Kurzweil*. Penguin Books.
- [Larman/Vodde, 2008] Larman/Vodde (2008). *Scaling Lean and Agile Development: Thinking and Organizational Tools for Large-Scale Scrum. By Craig Larman, Bas Vodde*. Paperback.

- [Le Corbusier, 1923] Le Corbusier (1923). *Vers une Architecture. Le Corbusier*. 1923.
- [Malveau, 1997] Malveau (1997). *CORBA Design Patterns By Raphael Malveau, Thomas J. Mowbray*. John Wiley and Sons. 1997.
- [Malveau, 2004] Malveau (2004). *Software Architect BOOTCAMP. The completely updated “field manual” for becoming a better software architect! By Raphael Malveau, Thomas J. Mowbray Ph.D.* Prentice All, Second Edition, Pearson Education Inc. ISBN: 0-13-141227-2.
- [Marco Vitruvio, 25ac] Marco Vitruvio (25ac). *De architectura Marco Vitruvio, 25 a. de C, 1.iii.ii.* 25 a. de C.
- [Martin Alaimo, 2014] Martin Alaimo (2014). *Proyectos Ágiles con Scrum. Flexibilidad, apredizaje, innovación y colaboración en contextos complejos. Por Martin Alaimo.* Kleer (Agile Coaching and training).
- [Martin Fowler, 1999] Martin Fowler (1999). *Refactoring: Improving the Design of Existing Code. By Martin Fowler (with Kent Beck, John Brant, William Opdyke, and Don Roberts).* Addison-Wesley.
- [Martin Fowler, 2002] Martin Fowler (2002). *Pattern of Enterprise Application Architecture. By Martin Fowler, David Rice, Matthew Foemmel, Edward Hieatt, Robert Mee, Randy Stafford.* Addison-Wesley.
- [Maturana, 1984] Maturana (1984). *El árbol del conocimiento. Por Humberto Maturana y F. Varela.* Santiago, Chile.
- [Maturana, 1988] Maturana (1988). *Ontología del Conversar. Por Humberto Maturana.* Santiago, Chile.
- [Meadows, 2009] Meadows (2009). *Thinking in Systems. A Primer. By Donella H. Meadows.* Diana Wright, Sustainability Institute London, Sterling, VA. First published by Earthscan in the UK in 2009.
- [Montrose, 1995] Montrose (1995). *Object-Oriented Development Using The Shlaer-Mellor Method. By Rodney C. Montrose.* Project Technology, Inc.
- [Moore, 1975] Moore (1975). *Progress in digital integrated electronics. By Moore Gordon E.* Artículo en el encuentro IEEE International Electron Devices Meeting, IEDM Technical Digest, 1975.
- [Oxford, 2015] Oxford (2015). *Oxford dictionaries, Language matters (spanish).* Oxford Dictionaries.
- [Perry, 1997] Perry (1997). *Software Architecture and its relevance for Software Engineering. By Dewayne Perry.* Coord'97.

- [Peter Senge, 1990] Peter Senge (1990). *La quinta disciplina. El arte y la práctica de la organización abierta al aprendizaje*. By Peter M. Senge. Editorial Granica, 2003. Edición original en inglés, 1990.
- [PMBOK, 2004] PMBOK (2004). *Guía de los Fundamentos de la Dirección de Proyectos Tercera Edición (Guía del PMBOK)*. Project Management Institute, Four Campus Boulevard, Newtown Square, PA 19073-3299 EE.UU.
- [Poppendieck, 2003] Poppendieck (2003). *Lean Software Development: An Agile Toolkit*. By Poppendieck, Mary. Addison Wesley.
- [Pressman, 2002] Pressman (2002). *Ingeniería del software. Un enfoque práctico. Quinta edición*. By Roger S. Pressman. Mc Graw Hill.
- [Robert Martin, 2009] Robert Martin (2009). *Design Principles and Design Patterns*. By Robert C. Martin (Uncle Bob). Objectmentor.com. Last verified 2009-01-14.
- [Rozanski/Woods, 2012] Rozanski/Woods (2012). *Software systems architecture : working with stakeholders using viewpoints and perspectives*. By Nick Rozanski, Eoin Woods. Addison-Wesley, Pearson Education Inc.
- [Sarabia, 1995] Sarabia (1995). *La Teoría General de Sistemas*. Por A. A. Sarabia. Isdefe, Ingeniería de Sistemas, Madrid, ed. 4.
- [SBOK, 2013] SBOK (2013). *Una guía para el conocimiento de Scrum (Guía SBOK) - 2013 Edición. Título original: A Guide to the SCRUM BODY OF KNOWLEDGE (SBOK GUIDE) 2013 Edition*. SCRUMstudy, una marca de VMEdU, Inc.
- [SCEA, 2010] SCEA (2010). *Sun Certified Enterprise Architect for Java EE Study Guide*. By Mark Cade and Humphrey Sheil. Prentice Hall, Second Edition.
- [Scott Ambler, 2004] Scott Ambler (2004). *The Object Primer 3rd Edition. Agile Model Driven Development with UML 2*. By Scott W. Ambler. Cambridge University Press, Paperback, 2004 ISBN: 0-521-54018-6.
- [Scott Bellware, 2008] Scott Bellware (2008). *Behavior-Driven Development*. By Scott Bellware. Code Magazine (June 2008) Retrieved 12 August 2012.
- [SEBoK, 2014] SEBoK (2014). *Guide to the Systems Engineering Body of Knowledge (SEBoK) v. 1.3*. By Body of Knowledge and Curriculum to Advance Systems Engineering (BKCASE) project. Sebokwiki.org, released 30 May 2014.

- [SEI, 2003] SEI (2003). *Software Architecture in Practice*. By Software Engineering Institute (SEI) with L. Bass, P. Clements, R. Kazman. Addison Wesley, 2nd Edition, 2003.
- [Shaw/Garlan, 1996] Shaw/Garlan (1996). *Software Architecture: Perspectives on an emerging discipline*. By Mary Shaw y David Garlan. Upper Saddle River, Prentice Hall, 1996.
- [SLI, 2009] SLI (2009). *Leadership and Systems in The Search for Leadership: An Organisational Perspective (Basic principles of systems thinking as applied to management and leadership)*. By Systemic leadership institute (systemicleadershipinstitute.org). Triarchy Press, May 2009.
- [SOLID, 2009] SOLID (2009). *SOLID Object-Oriented Design*, Sandi Metz (Duke University). Talk given at the 2009 Gotham Ruby Conference in May, 2009. Last verified 2009-01-15.
- [Stafford Beer, 1977] Stafford Beer (1977). *Designing Freedom*. By Stafford Beer. CBC Learning Systems, Toronto, 1974-1977; and John Wiley, London and New York, 1975.
- [Stephen Burd, 2011] Stephen Burd (2011). *Systems Architecture*. By Stephen D. Burd. University of New Mexico, Sixth Edition.
- [Stephen Hawking, 2010] Stephen Hawking (2010). *The Grand Design*. By Stephen Hawking, Leonard Mlodinow. An Bantam Books Trade Paper Back.
- [SWEBOK, 2014] SWEBOK (2014). *Guide to the Software Engineering Body of Knowledge (SWEBOK) By IEEE Computer Society project, Editors Pierre Bourque, École de technologie supérieure (ÉTS) Richard E. (Dick) Fairley, Software and Systems Engineering Associates (S2EA)*. IEEE, 2014. Paperback ISBN-10: 0-7695-5166-1. Paperback ISBN-13: 978-0-7695-5166-1.
- [UBA, 2015] UBA (2015). *Programa de estudio*. By UBA. Facultad de Arquitectura Diseño y Urbanismo. Ciudad Universitaria – Ciudad Autónoma de Buenos Aires.
- [UIT, 2014] UIT (2014). *Base de Datos de los Indicadores Mundiales de las Telecomunicaciones/TIC de la UIT. The World in 2014: ICT Facts and Figures*. By International Telecommunication Union (UIT). Printed in Switzerland. Geneva, April 2014. (Datos y cifras relativos a las TIC en un informe realizado por la División de datos y estadísticas de TIC de la Oficina de Desarrollo de las Telecomunicaciones de la UIT).

- [UNTREF, 2014] UNTREF (2014). *Construcción de software: una mirada ágil. Por Nicolás Paez, Diego Fontdevila, Pablo Suárez, Carlos Fontela, Marcio Degiovannini, Alejandro Molina.* Universidad Nacional de Tres de Febrero (UNTREF).
- [US, 2015] US (2015). *Filosofía Lean aplicada a la Ingeniería del Software. Por Miguel Ángel Moreno Martín.* Universidad de Sevilla.
- [Velthuis/Parra, 2007] Velthuis/Parra (2007). *Fábricas de software: experiencias, tecnologías y organización. Por Mario Piattini Velthuis y Javier Garzás Parra.* Alfaomega Ra-Ma.
- [Vilana Arto, 2011] Vilana Arto (2011). *Fundamentos del Lean Manufacturing. Nota Técnica 3.01. Por Profesor José Ramón Vilana Arto.* Escuela de Organización Industrial, Dirección de Operaciones. Executive MBA.
- [Wirf-Brock/McKean, 2015] Wirf-Brock/McKean (2015). *Filosofía Lean aplicada a la Ingeniería del Software. By Wirf-Brock and McKean.* Addison-Wesley, 2009. Pearson Education Inc., 2003. ISBN: 0-201-37943-0.