



**Universidad Nacional Autónoma de México**  
**Programa de Posgrado en Ciencias de la Administración**

**Un Modelo de Administración del Conocimiento para las  
Pequeñas Organizaciones que Desarrollan Software, aplicable  
al caso de los Programas de Código Abierto.**

**T e s i s**

que para optar por el grado de:

**Doctor en Ciencias de la Administración**

Presenta:

**Octavio Orozco y Orozco**

Comité Tutor:

**Dra. Graciela Bribiesca Correa**

Facultad de Contaduría y Administración

**Dra. Lucía Patricia Carrillo Velázquez**

Centro de Investigaciones Interdisciplinarias en Ciencias y Humanidades

**Dr. Fernando Gamboa Rodríguez**

Coordinación de Universidad Abierta y Educación a Distancia

México, D.F. junio 2013.



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

*Apreciada lectora, apreciado lector, si en vísperas del examen de grado lees esta tesis, probablemente apoyaste el trabajo de investigación que culmina al publicarla y por ello te doy de nuevo las gracias. De ser otro el caso, te invito a reflexionar y trascender lo que expreso en sus páginas.*

# Resumen

La dificultad para adecuar los programas de código abierto limita la utilización de este bien público en las pequeñas organizaciones que desarrollan software, pues esta actividad requiere de un uso intensivo del conocimiento organizacional que se considera necesario administrar. En esta tesis argumento que la concepción y modelo de administración vigentes son insuficientes en este caso. Así, a partir de concebir al conocimiento organizacional como un sistema complejo de acciones en ejecución, conjeturo que para atender la necesidad de estas organizaciones es pertinente un modelo de administración del conocimiento conformado por tres sistemas complejos de acciones –denominados Uso, Adecuación y Contribución. Para probar la pertinencia del modelo, realizo el diseño mixto de un método de investigación-acción de sistemas complejos y lo utilizo para estudiar el complejo empírico. Asimismo, presento los resultados de intervenir con el método una de estas organizaciones en México para implementar el modelo. Concluyo, a partir de las explicaciones del funcionamiento del sistema complejo, que el modelo que propongo es pertinente para atender la necesidad de administrar el conocimiento en la organización de estudio.

# Abstract

The difficulty in adapting open source software limits the use of this public good in small software development organizations, as this activity requires intensive organizational knowledge that is deemed necessary to be managed. In this thesis I argue that the current conception of organizational knowledge and methods for its management are insufficient in such case. Thus, from conceiving organizational knowledge as a complex system of actions being executed, I hypothesize that to attend the need of these organizations it is pertinent to apply a model to manage organizational knowledge which consists of three complex systems of actions –by the name of Usage, Adaptation and Contribution. To test the pertinence of the model I design a mixed methods research based on action-research and complex systems theory, and apply this research method to study the empirical complex. I then present the results of intervening, with the use of this method, one of those organizations in Mexico to implement the model. My conclusion is, based on the explanation of the inner workings of the complex system, that the model I propose is pertinent to fulfill the need of knowledge management in the subject organization.

# Índice general

<b>Resumen</b>	<b>iii</b>
<b>Abstract</b>	<b>iv</b>
<b>Introducción</b>	<b>1</b>
<b>1. Problema Práctico</b>	<b>4</b>
1.1. El software de código abierto . . . . .	5
1.2. La administración del conocimiento: una necesidad en las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto . . . . .	5
<b>2. Problema de Investigación</b>	<b>8</b>
2.1. Pregunta central de la investigación . . . . .	8
2.2. Objetivos de la Investigación . . . . .	8
<b>3. Estado del Arte</b>	<b>10</b>
3.1. Antecedentes. La concepción de conocimiento vigente en la administración del conocimiento . . . . .	12
3.2. El conocimiento como recurso tácito y explícito en la concepción vigente del pro- ceso de administración del conocimiento . . . . .	13
3.3. Dificultades y reflexiones sobre el proceso de administración del conocimiento cuando éste se concibe como recurso . . . . .	18
3.4. La teoría de creación del conocimiento organizacional permite explicar el proceso de la administración del conocimiento, en las organizaciones que desarrollan software	21
3.4.1. Necesidad y disyuntiva en las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto . . . . .	22
3.5. Conclusión . . . . .	27

<b>4. Marco Teórico Conceptual</b>	<b>28</b>
4.1. El conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas y su consecuente modelo de administración . . . . .	28
4.2. Antecedentes de una nueva concepción de conocimiento organizacional . . . . .	29
4.3. Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución . . . . .	44
4.4. Modelo para la administración del conocimiento en las pequeñas organizaciones que desarrollan software cuando pretenden adecuar software de código abierto (MACOSC) . . . . .	46
4.4.1. El MACOSC en la práctica administrativa . . . . .	47
<b>5. Hipótesis</b>	<b>53</b>
<b>6. Marco Metodológico</b>	<b>54</b>
6.1. Perspectiva sobre diversas alternativas de diseño de la investigación . . . . .	54
6.2. Criterios utilizados para el diseño de la investigación . . . . .	61
6.3. El diseño de la investigación-acción de sistemas complejos (IASC) . . . . .	64
<b>7. Estudio Empírico</b>	<b>72</b>
7.1. Perspectiva general de la pequeña organización sujeto de estudio . . . . .	72
7.2. Perspectiva general de CORBA . . . . .	73
7.3. Perspectiva general de la investigación . . . . .	74
7.3.1. Contrastación y Explicación Causal de la Hipótesis . . . . .	75
7.3.2. Plan general de acción para la contrastación y explicación causal . . . . .	76
7.3.3. Diagnóstico general de la oportunidad . . . . .	76
7.3.4. Planeación general de las acciones . . . . .	79
7.4. Primera iteración de la investigación-acción de un sistema complejo: El conocimiento organizacional como un sistema complejo de acciones en ejecución para crear SCAR con JacORB en el subproceso-estadio <b>Uso</b> . . . . .	80
7.4.1. Diagnóstico de la oportunidad . . . . .	80
7.4.2. Planeación de la acción . . . . .	81
7.4.3. Intervención . . . . .	82
7.4.4. Aprendizaje Reflexivo . . . . .	84
7.5. Segunda iteración de la investigación-acción de un sistema complejo: El conocimiento organizacional como un sistema complejo de acciones en ejecución para realizar con git el control de versiones distribuido, en el subproceso-estadio <b>Uso</b> . . . . .	85
7.5.1. Diagnóstico de la oportunidad . . . . .	85

7.5.2.	Planeación de la acción . . . . .	86
7.5.3.	Intervención . . . . .	86
7.5.4.	Aprendizaje Reflexivo . . . . .	88
7.6.	Tercera iteración de la investigación-acción de un sistema complejo: El conocimiento organizacional como un sistema complejo de acciones en ejecución para realizar el control de versiones de JacORB con git, en el subproceso-estadio <b>Ade-</b> <b>cuación</b> . . . . .	89
7.6.1.	Diagnóstico de la oportunidad . . . . .	89
7.6.2.	Diagnóstico de la oportunidad de utilizar instrumentos adicionales de investigación-acción de sistemas complejos . . . . .	91
7.6.3.	Planeación de la acción . . . . .	107
7.6.4.	Intervención . . . . .	108
7.6.5.	Aprendizaje Reflexivo . . . . .	108
	<b>Conclusiones</b>	<b>111</b>
	<b>Bibliografía</b>	<b>112</b>
	<b>Índice alfabético</b>	<b>122</b>



# Índice de figuras

4.1. Los Subprocesos-Estadios del MACOSC . . . . .	47
4.2. MACOSC en la Organización . . . . .	51
4.3. Tránsito por los estadios del MACOSC . . . . .	51
6.1. Gráfica detallada del IASC . . . . .	69
6.2. Ubicación del método IASC . . . . .	70
6.3. Cruce del MACOSC con el IASC . . . . .	71
7.1. Acciones en el cruce MACOSC-IASC . . . . .	77
7.2. Plan 1a iteración cruce MACOSC-IASC . . . . .	82
7.3. Resultados 1a iteración cruce MACOSC-IASC . . . . .	83
7.4. Resultados 1a iteración Comportamiento de Servidores CORBA . . . . .	84
7.5. Plan 2a iteración cruce MACOSC-IASC . . . . .	87
7.6. Resultados 2a iteración cruce MACOSC-IASC . . . . .	89
7.7. Plan 3a iteración cruce MACOSC-IASC . . . . .	107
7.8. Resultados 3a iteración cruce MACOSC-IASC . . . . .	110

# Índice de tablas

7.1. Plantilla de Proyecto de Software (PPS) . . . . .	92
7.2. Plantilla de Método (PM) . . . . .	93
7.3. Plantilla de Práctica (PP) . . . . .	93
7.4. Plantilla del Tablero de Estado de Ejecución de Método (PTEEM) . . . . .	94
7.5. Plantilla del Tablero de Estado de Ejecución de Práctica (PTEEP) . . . . .	95
7.6. Proyecto de Software (PS) - SCAR . . . . .	97
7.7. Método (M) - Adecuación . . . . .	100
7.8. Práctica (P) - Diagnóstico de la oportunidad . . . . .	101
7.9. Práctica (P) - Plan de acción . . . . .	102
7.10. Práctica (P) - Intervención . . . . .	103
7.11. Práctica (P) - Aprendizaje reflexivo . . . . .	103

# Palabras Clave y Abreviaturas

## Palabras clave

- Conocimiento organizacional, administración del conocimiento, sistemas complejos, pequeñas organizaciones que desarrollan software, software de código abierto.

## Keywords

- Organizational knowledge, knowledge management, complex systems, small software development organizations, open source software.

## Abreviaturas

**CORBA** Common Object Request Broker Architecture - corredor de invocaciones entre objetos. Véase la sección [7.2. Perspectiva general de CORBA](#) en la página [73](#).

**IASC** investigación-acción de sistemas complejos - diseño mixto de investigación. Véase la sección [6.3. El diseño de la investigación-acción de sistemas complejos \(IASC\)](#) en la página [64](#).

**MACOSC** Modelo para la administración del conocimiento en las pequeñas organizaciones que desarrollan software cuando pretenden adecuar software de código abierto, que parte de concebir al conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización. Véase la sección [4.4. Modelo para la administración del conocimiento en las pequeñas organizaciones que desarrollan software cuando pretenden adecuar software de código abierto](#) en la página [46](#).

**PCA** programas o software de código abierto. Véase la sección [1.1. El software de código abierto](#) en la página [5](#).

# Introducción

Cualquier persona que sea usuario de Internet o de alguno de la gran mayoría de teléfonos móviles avanzados es, también y quizá sin saberlo, usuario de programas de código abierto<sup>1</sup>. El amplio alcance de esta tecnología —considerada como un bien público, al que es posible acceder de manera libre— y el impacto que tiene en la sociedad son innegables porque los programas de código abierto son parte de los cimientos<sup>2</sup> sobre los que operan cotidianamente Internet y dichos teléfonos.

Sin embargo, la dificultad para adecuar los programas de código abierto limita utilizar ese bien público en las pequeñas organizaciones que desarrollan software, pues es una actividad que requiere de un uso intensivo del conocimiento organizacional y que se considera necesario administrar<sup>3</sup>. Así, estas pequeñas organizaciones en el caso de que se planteen como objetivo adecuar software de código abierto porque lo requieren para prestar sus servicios o proveer los bienes que son

---

<sup>1</sup>Véase (Orozco, 2011).

<sup>2</sup>En el caso de Internet estos cimientos consisten de tres tecnologías esenciales: protocolos básicos de comunicación *TCP/IP*, sistema de nombres de dominio *DNS* y correo electrónico *email*; esos cimientos continúan su evolución con sendos programas o software de código abierto: *BSD Sockets*, *BIND* y *sendmail*. Para el caso de la mayoría de los teléfonos móviles avanzados, en la década de los años ochenta —con el proyecto GNU (Free Software Foundation, 2010b)— y un poco más tarde, en la década de los años noventa, con el proyecto Linux se crearon los cimientos de sus sistemas operativos con software de código abierto. Sus sistemas operativos se basan en Linux —el caso de Android de Google— o se basan en FreeBSD (BSD Berkeley Unix) —en el caso de iOS de Apple. Si caracterizamos el alcance de esta tecnología con porcentajes, tenemos que: en el mercado de equipos de cómputo, un 14.7 % de las ventas de servidores fue con Linux para el cuarto trimestre del 2009 en todo el mundo (\$1,900 millones de dólares en dicho trimestre) (IDC, 2010). El 89.2 % de las supercomputadoras utilizan un sistema operativo con Linux (TOP500.Org, 2010). En 2011, el 52.5 % de los teléfonos celulares avanzados (smartphones) utilizaron un sistema operativo con Linux (Econsultancy, 2011); en 2012, 497 millones de smartphones —el 69 % del mercado global de 722 millones de teléfonos celulares avanzados— utilizan Android (IDC, 2013), que es un sistema operativo de código abierto basado en Linux. En la sección 1.1. *El software de código abierto* en la página 5, se caracteriza con mayor detalle el software de código abierto.

<sup>3</sup>Asumo que los administradores tienen a su cargo administrar el conocimiento organizacional porque la existencia de la organización depende de la eficacia con la cual dicho conocimiento se utiliza, también que el conocimiento organizacional es fundamental para prestar los servicios o proveer los bienes que son una parte esencial de la razón de ser de la organización.

una parte esencial<sup>4</sup> de su razón de ser, confrontan la necesidad<sup>5</sup> de administrar el conocimiento organizacional para lograr ese objetivo. No obstante, situado en el campo de las Ciencias de la Administración, argumento en esta tesis que la concepción y modelo de administración vigentes<sup>6</sup> son insuficientes<sup>7</sup> en este caso. Por ello, a partir de concebir<sup>8</sup> al conocimiento organizacional como un sistema complejo de acciones en ejecución<sup>9</sup>, conjeturo<sup>10</sup> que para atender la necesidad de estas organizaciones es pertinente un modelo de administración del conocimiento<sup>11</sup> conformado por tres subsistemas de acciones —que denomino Uso, Adecuación y Contribución. Para probar la pertinencia del modelo, realizo el diseño mixto de un método de investigación-acción de sistemas complejos<sup>12</sup> y lo utilizo para estudiar<sup>13</sup> el complejo empírico<sup>14</sup>. Asimismo, presento los resultados<sup>15</sup> de intervenir con el método una de estas organizaciones en México para implementar

---

<sup>4</sup>Es decir, el objetivo de adecuar el software de código abierto es a la vez un sub-objetivo de uno más amplio, en el que la organización como organismo viviente se propone el objetivo de su auto-producción. Plantea Morin: “Organizando la producción de objetos y servicios, la empresa se auto-organiza, se auto-mantiene, si es necesario se auto-repara y, si las cosas van bien, se auto-desarrolla desarrollando su producción” (Morin, 2004, 122-124).

<sup>5</sup>En la industria de desarrollo de software, la necesidad de administrar el conocimiento surge cuando es indispensable, debido a las características del software a desarrollar, involucrar a varias personas. Para vislumbrar la dificultad relacionada con el tamaño del software y que origina la necesidad de involucrar a varias personas, recurro al análisis que he realizado en el área de sistemas operativos Linux: Una versión reciente, hecha pública en 2009, del kernel Linux conocida como linux-2.6.29, contiene: 7 707 587 líneas de código abierto en lenguaje “C”. El desarrollo y continua adecuación de esta cantidad de código requiere del involucramiento de múltiples personas y de hecho es realizada cotidianamente por un número considerable de personas alrededor del mundo. Véase la descripción, como problema práctico, en el capítulo 1. *Problema Práctico* que inicia en la página 4 y la sección 1.2. *La administración del conocimiento, una necesidad en las pequeñas organizaciones que desarrollan software, cuando se proponen adecuar software de código abierto* en la página 5, en la que se plantea con mayor detalle esta necesidad. Véase la descripción, como problema teórico, en la sección 3.4.1. *Necesidad y disyuntiva en las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto* en la página 22, en la que se plantea la disyuntiva que confrontan estas organizaciones por la insuficiencia del modelo de administración del conocimiento vigente.

<sup>6</sup>Desde el campo de las Ciencias de la Administración, se planteó el problema de comprender y explicar los enfoques con los cuales se aborda la concepción vigente de conocimiento organizacional y el proceso metodológico para administrarlo, con la pretensión de atender esa necesidad. En el capítulo 3. *Estado del Arte* en la página 10, se describe con mayor detalle la concepción vigente de conocimiento organizacional y el proceso metodológico para administrarlo.

<sup>7</sup>Lo son porque se necesita administrar el conocimiento organizacional y no es posible hacerlo vía el modelo vigente de conversión de conocimiento: a) Tácito a tácito (socialización), b) Tácito a explícito (externalización), c) Explícito a explícito (combinación), d) Explícito a tácito (internalización). En la sección 3.4.1. *Necesidad y disyuntiva en las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto* en la página 22, se argumenta con detalle esta insuficiencia.

<sup>8</sup>Esta concepción se desarrolla con el análisis e interpretación integradores de los planteamientos que se detallan en el capítulo 4. *Marco Teórico Conceptual* en la página 28.

<sup>9</sup>En la sección 4.3. *Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución* en la página 44, se plantea con detalle esta concepción.

<sup>10</sup>Véase la hipótesis en el capítulo 5. *Hipótesis* en la página 53.

<sup>11</sup>Este modelo de administración del conocimiento se describe en la sección 4.4. *Modelo para la administración del conocimiento en las pequeñas organizaciones que desarrollan software, cuando pretenden adecuar software de código abierto (MACOSC)*, en la página 46.

<sup>12</sup>Véase el capítulo 6. *Marco Metodológico* en la página 54.

<sup>13</sup>Véase el capítulo 7. *Estudio Empírico* en la página 72.

<sup>14</sup>Véase en el capítulo 6. *Marco Metodológico* en la página 66, mi interpretación de complejo empírico.

<sup>15</sup>Véase el capítulo 7. *Estudio Empírico* a partir de la sección 7.4. *Primera iteración de la investigación-acción de*

el modelo. Finalmente, concluyo —a partir de las explicaciones del funcionamiento del sistema complejo— que el modelo que propongo es pertinente para atender la necesidad de administrar el conocimiento en la organización de estudio<sup>16</sup>.

---

*un sistema complejo: El conocimiento organizacional como un sistema complejo de acciones en ejecución para crear SCAR con JacORB en el subproceso-estadio Uso* en la página 80.

<sup>16</sup>Véanse las *Conclusiones* en la página 111.

# Capítulo 1

## Problema Práctico

La necesidad de administrar el conocimiento surge en cualquier organización que se proponga adecuar programas y sistemas de código abierto —con el objetivo de transformarlos y aplicarlos para aprovechar oportunidades y resolver problemas concretos de su contexto social, histórico y cultural particular— al percatarse del tamaño de esos programas y sistemas:

- En el área de sistemas operativos, una versión reciente, hecha pública en 2009, del kernel Linux conocida como linux-2.6.29 contiene: 7 707 587 líneas de código abierto en lenguaje “C”.
- En el área de arquitecturas estándares de cómputo distribuido para la integración de aplicaciones empresariales, con el estándar de la industria CORBA, la versión JacORB 2.3 tiene 201 168 líneas de código abierto en lenguaje “Java”.

El kernel Linux, a pesar de ser una parte central, requiere de mucho más código para poderse llamar sistema operativo y ser de utilidad a la mayoría de las organizaciones: La versión del sistema operativo, basado en Linux, llamado Debian contiene más de 52 millones de líneas de código abierto ([Geeknet, 2010](#)). De forma similar el ORB<sup>1</sup> y los elementos adicionales que se incluyen en la implementación del estándar CORBA hecha en JacORB, a pesar de ser también una parte central, requieren de más código para permitir la integración de aplicaciones en cualquier organización que desee utilizar dicho estándar.

Si se considera que la forma principal de adecuar los programas de código abierto, implica utilizar estas cantidades de código<sup>2</sup>, entonces es posible vislumbrar el porqué es indispensable

---

<sup>1</sup>El ORB (*Object Request Broker*) o “Corredor de Invocaciones entre Objetos” es un elemento central de la implementación del estándar CORBA. Véase en la página 73 la sección 7.2 *Perspectiva General de CORBA*.

<sup>2</sup>El código fuente para una obra se refiere a la manera preferida de la obra para hacerle modificaciones a la misma. Para una obra ejecutable, el código fuente completo significa todo el código fuente para todos los módulos que contiene (*Source code for a work means the preferred form of the work for making modifications to it. For an executable file, complete source code means all the source code for all modules it contains*) ([Free Software Foundation, 2010b](#)).

involucrar a varias personas y por qué para la pequeña organización que desarrolla software, cuyo objetivo es adecuar dichos programas a sus condiciones particulares<sup>3</sup>, le es necesario administrar el conocimiento para lograr en grado elevado su objetivo.

## **1.1. El software de código abierto**

La relevancia de este software estriba en que permite que cualquier organización se beneficie de su uso y contribuya a incrementar su valor. Utilizaremos el nombre “software de código abierto” de manera consistente en este trabajo para denominar, con un atributo común, a un amplio conjunto de programas que es amplio y relevante pues un usuario de servicios de correo electrónico, mensajes cortos, o de búsqueda en Internet, un usuario de sistemas de servicios de telefonía o teléfonos celulares —que les proveen empresas y organizaciones como Google, Apple, . . . , UNAM, etc.— es probablemente sin saberlo un usuario de software de código abierto —vía programas cuyos nombres quizá no le sean conocidos, como Linux, SendMail, FreeBSD, Apache, MySQL, PHP, Asterisk, etc.— porque dichas empresas y organizaciones utilizan estos programas para prestar sus servicios (Orozco, 2011). El atributo común con el que caracterizamos este software es que permite ejecutar, estudiar, modificar —como se juzgue pertinente— y redistribuir el código en el que esta escrito el software. Este atributo se especifica usualmente en una parte de la licencia de uso del software; sin embargo, conscientes de las importantes diferencias existentes en términos de licencias de uso y de las múltiples opciones en este ámbito remitimos al lector interesado en dar sus primeros pasos para profundizar en dicho ámbito a dos referentes importantes: la licencia de software de código libre (Free Software Foundation, 2010b), y la de software de código abierto (Open Source Initiative, 2013).

## **1.2. La administración del conocimiento: una necesidad en las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto**

La necesidad de administrar el conocimiento, que es importante en las pequeñas organizaciones que desarrollan software, se torna acuciante<sup>4</sup> cuando el desarrollo involucra adecuar software de

---

<sup>3</sup>Para prestar los servicios o proveer los bienes que son una parte esencial de su razón de ser, que en este caso al hacerlo, además, produce bienes aprovechables socialmente en entornos de redes sociales de innovación.

<sup>4</sup>Davin y Austin expresan los retos que se enfrentan al administrar las actividades que hacen uso intensivo del conocimiento, en las organizaciones del siglo XXI: La mayoría de las mujeres y hombres que realizan dichas actividades, las valoran en sí mismas y no sólo por el beneficio económico que les reportan; frecuentemente conocen más, acerca de lo que hacen, que sus administradores; cuando se agrupan, las novedades que producen con su actividad



código abierto debido a que, usualmente, éste es de gran tamaño.

La actividad de desarrollar este tipo de software requiere de varias personas y es intensiva en cuanto a la creatividad y conocimiento individual y colectivo (Crawford *et al.*, 2007, 605-606). La dificultad cognitiva de su desarrollo, y a partir de la cual se requiere involucrar a varias personas, se relaciona con el tamaño del software; su tamaño simbólico puede expresarse en *líneas de código (LC)* consideradas como *unidades*. El límite o umbral en el tamaño del software,  $T_0$ , se ha definido empíricamente como se muestra en la relación siguiente:

$$T_0 \geq 5\,000\ LC \quad (1.1)$$

Esto equivale a un desarrollo de software que consiste de 100 páginas de código fuente, en un lenguaje de programación de alto nivel y toda la documentación asociada necesaria. También es equivalente a una carga de trabajo, para un solo desarrollador de software, de más de un año de acuerdo con la productividad promedio en la industria como refieren Bohem (1987), Dale y Zee (1992), Jones (1981/1986), Livermore (2005) citados en (Wang, 2008, 12-14). El tamaño del software —si utilizamos la relación (1.1)— que hace evidente la dificultad cognitiva de su desarrollo y a partir del cual se requiere involucrar a varias personas, tiene como rango inicial de 5 000 a 10 000 *LC*. Por software de gran tamaño consideramos, entonces, el que va de cientos de miles a millones de *LC*. Así, en la actividad de desarrollo de software de gran tamaño nadie conoce completamente el sistema (Wang, 2008, 13-14).

Wang afirma que las infraestructuras organizacional y administrativa para el desarrollo de la actividad de ingeniería de software<sup>5</sup> han sido ignoradas, a pesar de ser necesarias, y que tanto los administradores de proyectos como los administradores de la organización juegan roles centrales en esta actividad. Argumenta que así como los programadores necesitan tecnologías que les auxilien en la programación, los administradores de la organización —que desarrolla y mantiene el software— necesitan métodos organizacionales y para la toma de decisiones; asimismo, los administradores de estos proyectos necesitan métodos de administración y control de calidad que les auxilien a balancear demandas encontradas de alcance del proyecto, de tiempo, de riesgo, de costo y de calidad. Todo ello para satisfacer los requerimientos y los objetivos de los diversos grupos de interés o *stakeholders* relacionados con la organización, quienes usualmente tienen requerimientos y expectativas diferentes (Wang, 2008, 857-859). De esta forma, si el desarrollo de software

---

grupal tienden a proliferar de manera exponencial e irrumpen en áreas impredecibles; la sinergia de su agrupación puede llevar a los administradores a un intenso estado de ansiedad (Devin y Austin, 2008, 490-491).

<sup>5</sup>La ingeniería de software es una disciplina de la ingeniería que estudia la naturaleza del software, los enfoques y metodologías del desarrollo de software de gran tamaño, las teorías y leyes del comportamiento del software, y de las prácticas de esta disciplina (*Software engineering is an engineering discipline that studies the nature of software, approaches and methodologies of large-scale software development, and the theories and laws behind software behaviors and software engineering practices*) (Wang, 2008, 2).

es una actividad colectiva que hace uso intensivo del conocimiento, podemos entender porqué se considera que éste necesita ser administrado.

Afirmamos entonces que la necesidad de administrar el conocimiento es acuciante para la pequeña organización que desarrolla software, en el caso en el que se proponga como objetivo adecuar software de código abierto, porque usualmente este excede —con creces— el umbral de las 5 000 *LC*. También, que para hacerlo ella requiere utilizar métodos administrativos y organizacionales pertinentes.

# Capítulo 2

## Problema de Investigación

Se considera que los programas de código abierto<sup>1</sup> cumplen con un conjunto de criterios específicos que deliberadamente eliminan cualquier límite para compartirlos, hacen posible que cualquier organización<sup>2</sup> se beneficie de su uso y además participe, si desarrolla software, para incrementar el valor<sup>3</sup> de dichos programas. También, que el desarrollo de software es una actividad que hace uso intensivo del conocimiento, que este requiere ser administrado y para ello se requieren métodos administrativos y organizacionales pertinentes.

Desde este enfoque:

### 2.1. Pregunta central de la investigación

¿Qué modelo para administrar el conocimiento es pertinente para la pequeña organización que desarrolla software, en el caso que se proponga adecuar programas de código abierto?

### 2.2. Objetivos de la Investigación

- Realizar una investigación documental del estado del arte del concepto de conocimiento organizacional y métodos para su administración, en las organizaciones que desarrollan software desde el campo de las Ciencias de la Administración, enfocando la necesidad de administrar el conocimiento en las pequeñas organizaciones que desarrollan software para el caso en que se proponen adecuar programas de código abierto.

---

<sup>1</sup>Linux, FreeBSD, Asterisk, Apache, JacORB, etc.

<sup>2</sup>Organización que pretenda aprovechar oportunidades al usar los programas.

<sup>3</sup>Lo incrementa al usar los programas y reportar errores que se encuentran, así como al adecuar y contribuir las mejoras.

- Proponer un modelo para la administración del conocimiento en las pequeñas organizaciones que desarrollan software para el caso en que se propongan adecuar programas de código abierto.
- Crear un diseño de investigación que permita probar la pertinencia del modelo para la administración del conocimiento en alguna organización sujeto de estudio.
- Aplicar el diseño de investigación para realizar el estudio empírico que permita probar la pertinencia del modelo para la administración del conocimiento en una organización sujeto de estudio.
- Reportar a la comunidad científica los resultados de la investigación.
- Propiciar la solución del problema práctico de administrar el conocimiento, para el caso en que la organización sujeto de estudio se proponga adecuar programas de código abierto.

# Capítulo 3

## Estado del Arte

El tema central de esta tesis me lleva a considerar como conceptos fundamentales al conocimiento organizacional y a la administración del conocimiento. Es por ello que en las secciones siguientes se describe el estado del arte en torno a estos dos conceptos y teorías que los originan en el campo de las Ciencias de la Administración. Se enfoca la necesidad de administrar el conocimiento en la pequeña organización que desarrolla software, para el caso de los programas de código abierto, a partir de las consideraciones expresadas en la sección 1.2. *La administración del conocimiento, una necesidad en las pequeñas organizaciones que desarrollan software, cuando se proponen adecuar software de código abierto* en la página 5:

- La actividad de desarrollar software de gran tamaño requiere de varias personas y es intensiva en cuanto a la creatividad y conocimiento individual y colectivo.
- La dificultad cognitiva de su desarrollo y a partir del cual se requiere involucrar a varias personas, se relaciona con el tamaño del software.
- El límite o umbral en el tamaño del software,  $T_0$ , se ha definido empíricamente como se muestra en la relación (1.1)  $T_0 \geq 5\,000\ LC$  en la página 6.
  - Esto equivale a un desarrollo de software que consiste de 100 páginas de código fuente, en un lenguaje de programación de alto nivel, y toda la documentación asociada necesaria.
  - También es equivalente a una carga de trabajo, para un solo desarrollador de software, de más de un año de acuerdo a la productividad promedio en la industria.
- El tamaño del software —si utilizamos la relación (1.1)— que hace evidente la dificultad cognitiva de su desarrollo y a partir del cual se requiere involucrar a varias personas, tiene como rango inicial de 5 000 a 10 000 LC.

- Por software de gran tamaño consideramos entonces el que va de cientos de miles a millones de *LC*.
  - Así, en la actividad de desarrollo de software de gran tamaño nadie conoce completamente el sistema.
- Las infraestructuras organizacional y administrativa para el desarrollo de la actividad de ingeniería de software han sido ignoradas, a pesar de ser necesarias, y que tanto los administradores de proyectos, como los administradores de la organización juegan roles centrales en esta actividad.
    - Así como los programadores necesitan tecnologías que les auxilien en la programación, los administradores de la organización —que desarrolla y mantiene el software— necesitan métodos organizacionales y para la toma de decisiones;
    - Asimismo, los administradores de estos proyectos necesitan métodos de administración y control de calidad, que les auxilien a balancear demandas encontradas de alcance del proyecto, de tiempo, de riesgo, de costo y de calidad.
  - Si el desarrollo de software es una actividad colectiva que hace uso intensivo del conocimiento organizacional, entonces podemos entender porqué se considera que éste necesita ser administrado y por qué para ello se necesita de métodos administrativos y organizacionales pertinentes.
  - Esta necesidad, que es importante en las pequeñas organizaciones que desarrollan software, se torna acuciante cuando se proponen adecuar software de código abierto porque este excede —con creces— el umbral de las 5 000 *LC*.

A continuación, en las secciones **3.1** y **3.2** se explora cómo se comprende y explica la concepción vigente de conocimiento organizacional y el proceso metodológico para administrarlo, en el campo de las Ciencias de la Administración. En la sección **3.3** se plantean diversas reflexiones sobre las dificultades asociadas a la concepción vigente de conocimiento y su administración. En la sección **3.4** se enfoca a las organizaciones que desarrollan software y se describe la disyuntiva que enfrentan las pequeñas organizaciones para el caso que involucra adecuar software de código abierto. Se argumenta que la concepción vigente de conocimiento organizacional y su método de administración, son insuficientes en dicho caso. Finalmente, la sección **3.5** concluye que es recomendable formular la concepción de conocimiento organizacional y su método de administración, a fin de que contemplen la necesidad de las pequeñas organizaciones que desarrollan software, cuando se proponen adecuar software de código abierto.

### 3.1. Antecedentes. La concepción de conocimiento vigente en la administración del conocimiento

El problema de la definición del conocimiento en occidente es registrado, quizá por vez primera, por Platón<sup>1</sup> en el diálogo de Sócrates con Teeteto<sup>2</sup>. En ese escrito realiza tres intentos por definirlo: El conocer es percepción, el conocer es opinión verdadera y el conocer es explicación acompañada de opinión verdadera<sup>3</sup> (Véase [Platón, 1988](#), 260, 267 y 313).

Aproximadamente 2000 años después, en 1641, Descartes<sup>4</sup>, en su búsqueda de un fundamento universal y necesario para el conocer, caracteriza al conocimiento como una experiencia de la cosa pensante, “que aumenta cada vez más sin llegar nunca hasta tal punto que no pueda hacerlo ya” ([Descartes, 1997](#), 42-43). Recurre a Dios como la cosa pensante, metafísica, “que en su unidad y simplicidad infinita es la causa última que evita que pueda darse un proceso al infinito en esta experiencia” ([Descartes, 1997](#), 45). En 1739, Hume<sup>5</sup> reflexiona en torno al análisis de la causa y efecto para acotar la capacidad cognitiva humana a lo que somos capaces de experimentar de manera directa: “Parece, según esto, que de las tres relaciones que no dependen de las meras ideas, la única que puede ser llevada más allá de los sentidos y que nos informa de existencias y objetos que no podemos ver o sentir es la *causalidad*.” Y más adelante: “Los objetos no poseen una conexión entre sí que pueda descubrirse. Y sólo partiendo de la costumbre que actúa sobre la imaginación podemos efectuar una inferencia desde la manifestación del uno a la existencia del otro.” Véase de la Parte III sendas secciones: Sección II, De la probabilidad y de la idea de causa y efecto; Sección VIII, De las causas de la creencia ([Hume, 1998](#), 75;104).

En la segunda mitad del siglo XX, en 1966, Polanyi<sup>6</sup> reconsidera la concepción del conocimiento humano a partir del hecho de que “podemos conocer más de lo que podemos decir” (*We can know more than we can tell*). Al analizar este tipo de conocimiento lo denomina como *tácito*<sup>7</sup> y lo asocia con el conocimiento de tipo práctico para diferenciarlo del conocimiento de tipo

---

<sup>1</sup>Vivió ca. 428/427 a.C. – 347 a.C.

<sup>2</sup>Que escribe ca. 369 a.C.

<sup>3</sup>Ante sendos problemas que enfrentan estas definiciones al considerar: el cambio, el regreso al infinito y el dar vueltas en círculo y sin fin, Platón —quizá de forma prudente— permite que Sócrates abandone el diálogo, para ir a responder la acusación de Meleto en el Pórtico del Rey.

<sup>4</sup>Vivió 1596-1650.

<sup>5</sup>Vivió 1711-1776.

<sup>6</sup>Vivió 1891-1976.

<sup>7</sup>Polanyi afirma que su análisis de este tipo de conocimiento está muy ligado a la demostración que la psicología Gestalt hizo sobre el conocimiento que podemos tener de una fisonomía, al integrar nuestra percepción de sus rasgos particulares sin ser a la vez capaces de identificarlos por separado. Como resultado de su análisis, caracteriza al conocimiento *tácito* con cuatro aspectos: funcional, fenoménico, semántico y ontológico ([Polanyi and Sen, 2009](#), 13). Así, afirma que sin atender directamente al aspecto funcional —los rasgos mismos en el caso de una cara— podemos estar conscientes de este aspecto, en su contribución para conformar nuestra comprensión —en el caso, reconocer la cara vía los otros tres aspectos— de la entidad total ([Polanyi and Sen, 2009](#), 18).

intelectual<sup>8</sup> o proposicional (Polanyi and Sen, 2009, 4-7).

Ubicado en el campo de la administración interpreto que Nonaka y Takeuchi conocían sobre el análisis tripartito del conocimiento que define al conocimiento como **creencia verdadera justificada**; así como el planteamiento de los contraejemplos que Gettier en un conciso documento (Gettier, 1963) propone para este análisis, en cuanto a que la creencia verdadera justificada es condición para conocer necesaria, más no suficiente (veáse Nonaka y Takeuchi (1995, 21), así como las notas 2 y 3 Nonaka y Takeuchi (1995, 50,51)). Considero que por ello “expanden” en una dirección más práctica la propuesta filosófica de Polanyi (“*While Polanyi argues the contents of tacit knowledge further in a philosophical context, it is also possible to expand his idea in a more practical direction*”) (Nonaka y Takeuchi, 1995, 60). Para definir el conocimiento lo dividen en aspectos intangibles y tangibles, lo que incide en la génesis de uno de los recientes enfoques de las Ciencias de la Administración denominado administración del conocimiento (*knowledge management* o gestión del conocimiento). Proponen, por un lado, que el conocimiento intangible o tácito es personal y contextualizado, por ende difícil de formalizar y comunicar<sup>9</sup>. El conocimiento tangible o explícito, por otro lado, se refiere a conocimiento que es transmisible en un lenguaje formal, sistemático (Nonaka y Takeuchi, 1995, 58-62).

### **3.2. El conocimiento como recurso tácito y explícito en la concepción vigente del proceso de administración del conocimiento**

Nonaka y Takeuchi plantean el proceso de creación del conocimiento organizacional a partir de que el conocimiento es creado sólo por individuos y que la organización soporta y provee el contexto de dicha creación. La asunción crítica de dicho proceso, sostienen, es que el conocimiento organizacional se crea y expande a través de la interacción social entre el conocimiento tácito y el explícito. A partir de reconocer esa interacción crean un modelo de conversión de conocimiento dinámico, interactivo y en espiral que consta de cuatro formas diferentes de conversión del conocimiento: a) Tácito a tácito (socialización), b) Tácito a explícito (externalización), c) Explícito a explícito (combinación), d) Explícito a tácito (internalización). Sostienen que los manuales son una de las formas más concretas de capturar el conocimiento en su faceta tangible, explícita (Nonaka y Takeuchi, 1995, 58-62; 218).

Así, proponen su teoría de la creación del conocimiento organizacional con la finalidad de

---

<sup>8</sup>Que Gilbert Ryle identifica, nos dice Polanyi, como *know how* y *know that* respectivamente.

<sup>9</sup>Caracterizan este conocimiento al decir que tiene que ver con creencias y compromisos; que es una función de una particular perspectiva o intención; que tiene que ver con la acción, que siempre es conocimiento para un fin; que tiene que ver con significado, que es específico a un contexto y que es relacional.



desarrollar capacidades para innovar, pues consideran que cuando una organización innova, procesa información que llega de fuera con el fin de resolver problemas y adaptarse a entornos cambiantes. Al innovar crea nuevo conocimiento e información, de adentro hacia afuera, con el fin de redefinir tanto los problemas como las soluciones, y durante ese proceso re-crear su entorno. Plantean, además, que las organizaciones deben satisfacer cinco condiciones para la creación del conocimiento: Intención, autonomía, fluctuación y caos creativo, redundancia, y variedad o diversidad interna. Así como realizar cinco fases para la administración del proceso de conversión de conocimiento tácito en explícito en la organización de manera interactiva y en espiral: Compartir conocimiento tácito, crear conceptos, justificar conceptos, construir un arquetipo, y expansión vertical y horizontal del conocimiento (Nonaka y Takeuchi, 1995, 56-58 y 73-90).

Lehaney et al. proponen la siguiente definición de administración del conocimiento: “La administración de conocimiento se refiere a la sistemática organización, planeación, programación *scheduling*, monitoreo e implantación de personas, procesos, tecnología y entorno con objetivos apropiados y mecanismos de retroalimentación bajo el control de un sector público o privado que impulsa y facilita explícita y específicamente la creación, retención, compartición, identificación, adquisición, utilización y medición de información y nuevas ideas con el fin de lograr objetivos estratégicos como son el aumento y mejora de la competitividad y el rendimiento sujetos a restricciones financieras, legales, de recursos, políticas, técnicas, culturales y sociales. Tres áreas con sendos sistemas organizacionales interactúan en la administración del conocimiento: personal, estructuras y tecnología. Los procesos de conocimiento que estas áreas realizan incluyen la creación, retención, compartición, identificación, adquisición, utilización y medición de información y nuevas ideas. Los objetivos estratégicos que persiguen son el aumento y mejora de la competitividad y el rendimiento. El conocimiento que utilizan, crean y recrean se clasifica por tipo de habilidad para saber-qué, saber-quién, saber-cómo, saber-dónde, saber-porqué, saber-cuándo” (Lehaney et al., 2004, 3-10). Interpretamos en Lehaney et al. una concepción del conocimiento dual que identifica al conocimiento explícito con información, al tácito con las ideas.

(Bjørnson y Dingsøyr, 2008) aceptan que la administración del conocimiento es un método que simplifica el proceso de compartir, crear, distribuir, capturar y entender el “conocimiento” de una empresa<sup>10</sup>. Asimismo, interpretamos en esta propuesta una concepción del conocimiento dual que hace clara referencia al proceso de conversión de conocimiento que plantean Nonaka y Takeuchi.

Carrillo analiza los antecedentes teóricos de la administración del conocimiento e inicialmente plantea a la gestión del conocimiento como “un proceso de administración de las actividades y tareas dirigidas a adquirir, desarrollar y usar el conocimiento organizacional”. Desde esta perspectiva, plantea una metodología para instrumentar el proceso de administración del conocimiento,

---

<sup>10</sup>A method that simplifies the process of sharing, distributing, creating, capturing and understanding of a company’s “knowledge” (Bjørnson y Dingsøyr, 2008, 1055).

que se describe más adelante y construye un modelo conceptual para explicar este proceso como un sistema complejo<sup>11</sup> conformado por tres procesos particulares: crear, evaluar y comunicar conocimiento, en el que considera la dimensión individual y social como componentes del sistema (Carrillo, 2008, 70-71; 90-108). Para probar la utilidad de la metodología y el modelo, selecciona al conocimiento caracterizado de acuerdo con los usos de la disciplina contable y al que se le denomina Capital Intelectual (Carrillo, 2008, 61-62; 74-75). Por otra parte, observa diversas entidades académicas para el estudio de caso y aplica su definición de la siguiente forma: “Gestión del conocimiento es una tendencia de la práctica administrativa dirigida a incidir en el proceso de creación y reconocimiento de capital intelectual, que se adopta como estrategia organizacional con la finalidad de incrementar la capacidad de competencia” (Carrillo, 2008, 68-69). Clasifica el Capital Intelectual (CI) por el grado de codificación del conocimiento<sup>12</sup>: “Al que reside en la mente de las personas, se le denomina CI, está basado en conocimiento tácito y se sustenta en el conocimiento difícil de formalizar y codificar, pues requiere de la voluntad del individuo. Al que se sustenta en conocimiento susceptible de ser codificado y transferible a través del lenguaje formal y sistematizado es reconocido como CI basado en conocimiento explícito.” (Carrillo, 2008, 79-80). Carrillo realiza una propuesta metodológica para la construcción, *ad hoc* a la organización, de un Modelo –conceptual– de Gestión del Conocimiento. En una primera experiencia empírica construye el modelo para crear capital intelectual a efecto de incrementar la capacidad de competencia en la organización que analiza. En dicho estudio Carrillo relaciona al individuo con representaciones –objeto– de conocimiento manifiestas en la realidad. En particular esta propuesta metodológica reconoce a las organizaciones como sistemas complejos en constante equilibrio dinámico por lo cual plantea que la construcción del modelo de gestión del conocimiento es en sí mismo un proceso de gestión del conocimiento que cada organización lleva a cabo a fin de construir el modelo *ad hoc* para reducir períodos de ajuste o resiliencia (Carrillo, 2012, 4). En la metodología y modelo conceptual propuestos es clara la concepción del conocimiento dual que emplea Carrillo, planteada por Nonaka y Takeuchi.

Solleiro *et al.*, para estudiar los procesos de gestión del conocimiento en centros de investigación y desarrollo de México, Brasil y Chile recurren a la definición de gestión del conocimiento como el “proceso constante de identificar, encontrar, clasificar, proyectar, presentar y usar de un modo más eficiente el conocimiento y la experiencia del negocio, acumulada en la organización” (Solleiro *et al.*, 2009, 23). Lo consideran como un tema de alta relevancia en la gestión de organizaciones porque les permite “ser más competitivas”. Así, proponen identificar los activos intelectuales para proteger, de forma rigurosa, aquellos que sean sujetos de registro de propiedad intelectual (Solleiro *et al.*, 2009, 24). Para ello, asumen como capital intelectual “la combinación del Capital

---

<sup>11</sup>(Véase Carrillo, 2008, 70-74, notas 29-33).

<sup>12</sup>De acuerdo, afirma, con la propuesta de Lubit (2001).

Humano, Organizativo y Relacional de una organización”<sup>13</sup> y lo ven como “el capital invisible que poseen las organizaciones” (Solleiro *et al.*, 2009, 33). Describe que los modelos más avanzados para medir el capital intelectual responden a la necesidad de “recoger en un esquema fácilmente comprensible los elementos intangibles que aportan o agregan valor para la empresa a través de un proceso de identificación, selección y medición de activos, basados en los tres capitales citados: capital humano<sup>14</sup>, capital estructural<sup>15</sup> y relacional<sup>16</sup>” (Solleiro *et al.*, 2009, 35).

Por otro lado, de Gortari, para caracterizar lo que denomina como la nueva administración del conocimiento parte de su concepción dual vigente<sup>17</sup>, observa la relevancia del conocimiento en las empresas que estudia —Cemex y Vitro— para concluir que en ambas “para su uso y aprovechamiento fue necesario un proceso de reestructuración de la empresa, que permitiera que sus estructuras organizacionales —en particular las direcciones de tecnología— puedan actuar como mecanismos de mediación de manera que se pueda generar una dinámica de interacción y combinación del conocimiento tácito y codificado de los individuos para transformarlo en colectivo para integrar y acumular conocimiento que sirva de base para la construcción de las ventajas competitivas de las empresas” (de Gortari, 2007, 381).

Finalmente, Suárez, Carreto y Ruiz, al plantear un modelo de administración del conocimiento aplicado a enseñanza-aprendizaje por medio de dispositivos móviles, identifican a la “gestión o administración del conocimiento como un concepto aplicado en las organizaciones que pretende transferir el conocimiento y experiencia existente entre sus miembros, de modo que pueda ser utilizado como un recurso disponible para otros en la organización. La administración del conocimiento implica la conversión del conocimiento tácito (el que sólo sabe alguien) en explícito (conocimiento documentado y replicable) para convertirlo en un activo estratégico de la organización.”(Suárez, Carreto y Ruiz, 2012, 152).

Interpretamos que en Suárez, Carreto y Ruiz, de Gortari y en Solleiro *et al.* es evidente la influencia de la concepción del conocimiento dual que plantean Nonaka y Takeuchi. También que en los autores latinoamericanos estudiados es preponderante la utilización de criterios de análisis como la ventaja competitiva y el capital intelectual<sup>18</sup>

Del estado del arte que presentan en sendos trabajos (Desouza, 2000, 99) y (Bjørnson y Dingsøyr,

---

<sup>13</sup>Definición que asumen a partir del trabajo del proyecto (MERITUM, 2001).

<sup>14</sup>“Se refiere al conocimiento (tácito y explícito) que poseen las personas, así como la capacidad de poder regenerarlo —la capacidad para aprender”.

<sup>15</sup>“Representa el conocimiento propio de la organización, es decir el que poseen las personas y los equipos de la entidad, sea explícito, codificado o sistematizado mediante un proceso formal —a través de rutinas organizativas o pautas de acción”.

<sup>16</sup>“Se refiere al valor que tiene para la organización el conjunto de relaciones —económicas, políticas e institucionales— que mantiene con los diferentes agentes sociales”.

<sup>17</sup>Esta dualidad la refiere como conocimiento tácito y como conocimiento codificado, para el explícito.

<sup>18</sup>Acordes estos criterios a la Gestalt con la que enfocan sus respectivos casos de estudio (véase Khun 2007, 246; Khun 2007, 213).

2008, 1055) se identifican dos clases de estrategias<sup>19</sup> para la respectiva administración del conocimiento explícito y tácito:

- La codificación: Se sistematiza y almacena información que constituye el conocimiento de la empresa, para que este disponible para el personal de la empresa. Esto implica separar el conocimiento de su creador.
- La personalización: Se soporta el flujo de información en la empresa con un almacén central de la fuentes de conocimiento, como una sección amarilla *yellow pages* sobre **quién sabe qué** en la empresa. Se considera opuesto a la codificación, e implica compartir el conocimiento vía la interacción y diálogo entre personas. El conocimiento no se separa de su fuente, es necesario identificar a la persona para solicitarlo.

De la misma forma se identifican escuelas de la administración del conocimiento tácito y explícito. Por ejemplo, (Bjørnson y Dingsøyr, 2008, 1055) clasifican<sup>20</sup> los trabajos en la administración del conocimiento como sigue:

- Tecnocrática:
  - Sistemas: Se enfoca en la tecnología para compartir el conocimiento, utilizando repositorios de conocimiento.
  - Cartográfica: Se enfoca en crear mapas de conocimiento y directorios de conocimiento.
  - Ingeniería: Se enfoca en los procesos y flujos de conocimiento en las organizaciones.
- Económica<sup>21</sup>: Se enfoca en la relación entre el conocimiento como activo *asset* y los ingresos en las organizaciones.
- Del comportamiento:
  - Organizacional: Se enfoca en las redes para compartir conocimiento.
  - Espacial: Se enfoca en el diseño de los espacios de oficina que promueven que se comparta el conocimiento.
  - Estratégica: Se enfoca en la forma en que el conocimiento se puede ver como la esencia de la estrategia de una empresa.

---

<sup>19</sup>Se reconoce que hay una multitud de estrategias que pueden ser categorizadas en dos.

<sup>20</sup>De acuerdo, nos dicen, con Earl (2001).

<sup>21</sup>No tiene subclasificaciones a la fecha.

El enfoque de la administración del conocimiento, que concibe al conocimiento como un fenómeno socialmente construido, está en proceso de desarrollo. Al caracterizar así al conocimiento, su administración debe considerar los sistemas humanos complejos, comunidades de práctica, zonas de conocimiento y estructuras de soporte orgánico. Esta tendencia se identifica como una segunda generación de administración del conocimiento y se considera que en ésta el cambio es guiado más que planeado (Bjørnson y Dingsøy, 2008, 1057).

Jennex después de hacer una revisión general, basada en la concepción dual de conocimiento organizacional, concluye que la administración del conocimiento es una disciplina vital; que ayuda a las organizaciones en la utilización del conocimiento de sus miembros y en la construcción del conocimiento organizacional; que impacta el rendimiento organizacional, y que las organizaciones que utilizan bien su conocimiento tienden a mejorar su rendimiento y a ser más exitosas (Jennex, 2008, 345).

De esta forma la vigente definición dual del conocimiento, planteada por Nonaka y Takeuchi, es un elemento dominante en los esfuerzos por administrarlo. Se reconoce además que es difícil separar el conocimiento de su creador, por ende que es difícil de formalizar y codificar ese “capital invisible de las organizaciones”.

### **3.3. Dificultades y reflexiones sobre el proceso de administración del conocimiento cuando éste se concibe como recurso**

Así, en la administración del conocimiento cuando éste se considera como un recurso, propiedad de la organización y algo que puede ser capturado, transmitido y almacenado, el objetivo ha sido desarrollar<sup>22</sup> los recursos informáticos necesarios para crear, capturar y comunicar dicho conocimiento organizacional<sup>23</sup>. Sin embargo, esta concepción genera diversas dificultades que impactan la administración del conocimiento.

En una revisión y extracto de lo que publicaron en 1998 sobre la administración del conocimiento, Davenport y Prusak advierten que la tecnología informática es sólo el conducto y almacén para el intercambio de conocimiento, que no crea conocimiento y que no puede garantizar o aún promover la generación de conocimiento o compartición de conocimiento en una cultura corporativa que no favorece esas actividades. Respecto del conocimiento, Davenport y Prusak plantean que su definición no es clara ni simple y redundan al describir su naturaleza dual. Respecto del valor del conocimiento, afirman que se ubica cerca de la acción, que puede y debe ser evaluado por las acciones a las que conduce, pero que puede ser difícil trazar la ruta entre conocimiento y

---

<sup>22</sup>Aprovechando las oportunidades de los decrecientes costos de las computadoras y redes de comunicación.

<sup>23</sup>Véase por ejemplo: Carrillo 2008, 172-173 y de Gortari 2007, 381.

acción (Davenport y Prusak, 2000).

Herrera, Ramírez y May al proponer un modelo e indicadores de capital intelectual identifican bienes o activos intelectuales<sup>24</sup>, sin sustancia física, útiles para la producción o abastecimiento de bienes, prestación de servicios o para propósitos administrativos, que generarán beneficios económicos futuros controlados por la organización. Notan, sin embargo, que “surge la problemática de cómo se puede evaluar un beneficio si anteriormente no había un adecuado sistema de registro y cálculo de los activos señalados; incluso en lo relacionado con la presentación de los intangibles en los informes financieros se dificulta y en algunos aspectos diverge el razonamiento económico con respecto a los postulados contables básicos, por ejemplo, la disminución del valor de los recursos intangibles o su amortización contable cuando en lugar de restar su generación de beneficios se incrementa financieramente *a posteriori*. Todo ello repercute definitivamente en la presentación razonable y objetiva de los datos contables y en la toma de decisiones.” (Herrera, Ramírez y May, 2012, 262). Concluyen respecto del capital intelectual que “aunque incorpóreo, es efectivamente un activo generador de bienes” (Herrera, Ramírez y May, 2012, 273).

Por su parte, Desouza advierte que las organizaciones que desarrollan software tienen que reconocer que las tecnologías de información y comunicaciones son sólo un medio para promover el conocimiento, que el conocimiento en la ingeniería de software tiene una naturaleza tácita y contextual que en buena medida no puede ser articulado o expresado de manera explícita (véase Desouza, 2000, 100).

Por otro lado, Quintanilla concibe a la gestión del conocimiento como esencial para la gestión de la innovación en las empresas, porque plantea que “la empresa debe adoptar estrategias de gestión integral del conocimiento.” Sin embargo, afirma que “para tener una idea adecuada del papel del conocimiento en la innovación tecnológica, conviene tener en cuenta dos tipos de criterios para la clasificación de las formas del conocimiento”. Los criterios que propone son de contenido (representacional-operacional)<sup>25</sup> y de forma (explícito-tácito)<sup>26</sup>. Advierte además: “Con frecuencia se tiende a confundir los dos criterios de distinción de tipos de conocimiento, suponiendo que todo el conocimiento representacional es o puede ser explícito, mientras el conocimiento operacional o *know how* tiene al menos un componente tácito irreducible. En realidad, los dos criterios de clasificación son independientes”, y concluye que son cuatro<sup>27</sup> tipos de conocimiento: Intui-

---

<sup>24</sup>Los denominan también como capital intelectual o capital del conocimiento y orientan su trabajo hacia las organizaciones artísticas en México.

<sup>25</sup>“El conocimiento representacional o *know that* consiste en la representación y explicación de las propiedades y regularidades características de las entidades y procesos. El conocimiento operacional o *know how* se refiere a propiedades y reglas características de acciones u operaciones de transformación de cosas o procesos” (Quintanilla, 2005, 240).

<sup>26</sup>“El *conocimiento explícito* es el que se puede formular adecuadamente mediante un conjunto de enunciados. El *conocimiento tácito* es el conocimiento personal no formulado explícitamente mediante un conjunto de enunciados” (Quintanilla, 2005, 241).

<sup>27</sup>De estos cuatro tipos pone énfasis en dos, para ofrecer sendas acepciones de la noción de conocimiento técnico,

ción, Habilidad, *Know how* y *Know that*; en lugar de los dos denominados explícito e implícito (Quintanilla, 2005, 239-241).

Le Blanc y Ermine notan que en los conceptos de sociedad del conocimiento, economía del conocimiento, administración del conocimiento e ingeniería del conocimiento, las tecnologías de información y comunicaciones están siempre presentes de manera preponderante. Sin embargo, que éstas se dedican al proceso de información (Le Blanc y Ermine, 2007, 52). Así se preguntan si es posible medir el conocimiento de un sistema y se esfuerzan en buscar un formalismo similar al que se utiliza para medir la cantidad de información propuesto por Shannon<sup>28</sup>. Proponen el modelo *AIK* de un sistema *S* de administración de conocimiento<sup>29</sup> que utiliza la concepción dual de Nonaka y Takeuchi. Se plantean que en una base de conocimiento o *knowledge corpus* se puede, a partir de la información que contiene —que se representa sobre tres ejes: información<sup>30</sup>, sentido y contexto— calcular sendos tipos de entropía (...) pero que no les es aún claro cómo combinarlas para medir la cantidad de conocimiento del sistema (*The combination of those three entropies is not yet clear, and is currently under research, to define a global knowledge entropy of the corpus*) (Le Blanc y Ermine, 2007, 64).

En variados ámbitos en que se considera necesaria la administración del conocimiento, al pretender aplicar la definición dual del conocimiento organizacional, planteada por Nonaka y Takeuchi, la administración se enfrenta con problemas, como apreciamos en esta sección que registra la correspondiente reflexión de diversos investigadores. Estos problemas impactan la administración del conocimiento, por lo que a continuación describimos la situación que prevalece para el caso que nos concierne.

---

distinguiendo entre conocimiento técnico en sentido primario que corresponde a la habilidad y en sentido secundario que corresponde al *know how*.

<sup>28</sup>Shannon a su vez reconoce la influencia de la mecánica estadística —aunque no refiere directamente a Ludwig Boltzmann (1844-1906) y su fórmula  $S = k \log W$ — al plantear su propuesta: “Quantities of the form  $H = -\sum p_i \log p_i$  (the constant  $K$  merely amounts to a choice of a unit of measure) play a central role in information theory as measures of information, choice and uncertainty. The form of  $H$  will be recognized as that of entropy as defined in certain formulations of statistical mechanics where  $p_i$  is the probability of a system being in cell  $i$  of its phase space” (Shannon, 1948, 11).

<sup>29</sup>En el modelo *AIK*, dentro del sistema *S* está la *A* que representa las redes de trabajadores del conocimiento o *knowledge workers networks* y el sistema de información *I*. Notablemente fuera del sistema *S* se encuentra *K* el capital intelectual o *knowledge capital*. Con este modelo formal intentan tener una base sólida para definir el capital intelectual de un sistema organizado (Le Blanc y Ermine, 2007, 53).

<sup>30</sup>Nótese el círculo sobre información.



### **3.4. La teoría de creación del conocimiento organizacional permite explicar el proceso de la administración del conocimiento, en las organizaciones que desarrollan software**

La dificultad recurrente en el desarrollo de software ha llevado a considerar a la administración del conocimiento como un elemento sustantivo para solucionar dicha dificultad.

Ainamo, al caracterizar los proyectos globales y concluir que son una solución organizacional, con creciente aceptación en diversos sectores y sociedades, afirma que los administradores de muchas empresas globales en los sectores de las tecnologías de la información y comunicaciones invierten actualmente en sistemas de administración del conocimiento<sup>31</sup> con el objetivo de desarrollar competencias organizacionales distintivas (Ainamo, 2008, 488).

Para la ingeniería de software, Crawford *et al.* plantean que en el desarrollo de software es posible identificar dos tipos de conocimiento: a) Conocimiento inmerso en los productos o artefactos que son fruto de actividades altamente creativas, b) Conocimiento sobre productos y procesos (al cual le llaman meta-conocimiento). Y que algunas de las fuentes del conocimiento explícito (artefactos, objetos, componentes, patrones, esquemas y contenedores) se almacenan electrónicamente. Sin embargo, también consideran que la mayoría de dicho conocimiento es tácito y esta incorporado en las personas que conforman los equipos de desarrollo. A partir de esto, nos dan la perspectiva siguiente sobre la administración del conocimiento en la ingeniería de software: “La administración del conocimiento se enfoca en el conocimiento corporativo como un activo crucial de la empresa y persigue la utilización y desarrollo óptimos de este activo, ahora y en el futuro.” Proponen que el principal argumento para administrar el conocimiento en la ingeniería de software es que ésta es una actividad de uso y creatividad intensos de conocimiento. Que el desarrollo de software es un proceso en el que cada persona involucrada tiene que tomar gran cantidad de decisiones y el conocimiento individual debe ser compartido y llevarse a los niveles de proyecto y organizacional; esto es exactamente, nos dicen, lo que la administración del conocimiento propone. Las personas en estos grupos tienen que –crear— colaborar, comunicar y coordinar su trabajo lo cual hace necesaria la administración del conocimiento (Crawford *et al.*, 2007, 605-606).

De los diversos métodos planteados recientemente para el desarrollo de software, destacan los que denominamos como proyectados y los ágiles. Por un lado, los administradores en los equipos proyectados se enfocan en seguir de manera preferente las prácticas establecidas en un estándar con reconocimiento nacional o internacional. Estas prácticas típicamente incluyen la administración del desarrollo como un proyecto —con sus fases de inicio, planeación, ejecución y cierre— y una relación con el cliente regida de forma contractual, el establecimiento de procesos y control de

---

<sup>31</sup>Sistemas de administración del conocimiento que Ainamo describe con la concepción dual vigente.



calidad, así como documentación comprensiva previa al desarrollo del software. En estos proyectos las prácticas se rigen por estándares o normas y en ellos se considera importante la administración del conocimiento y usualmente se utiliza una base de datos de conocimiento *knowledge database*. Esto se constata cuando se recurre la norma nacional mexicana aplicable a pequeñas empresas (véase Oktaba (2006, 93-101); Oktaba *et al.* (2007, 21-28)) y también cuando se recurre a la norma internacional aplicable a pequeñas empresas que desarrollan software (véase ISO-IEC 29110-5-1-2 (2011) y Morales, Ventura, Oktaba y Torres (2012)). Por otro lado, los administradores en los equipos ágiles se enfocan en las acciones de los miembros del equipo<sup>32</sup>, para facilitar un ambiente auto-organizado que avance (en la solución)<sup>33</sup> (véase Nerur y Balijepally, 2007, 83).

Podemos interpretar que la administración del conocimiento en ambos casos hace clara referencia a la concepción de conocimiento —como recurso— explícito y tácito, en relación con la teoría de creación del conocimiento organizacional planteada por (Nonaka y Takeuchi, 1995), también que los métodos proyectados ponen énfasis en la administración del conocimiento explícito y los métodos ágiles en la del tácito.

### 3.4.1. Necesidad y disyuntiva en las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto

Cuando las pequeñas organizaciones que desarrollan software se proponen adecuar software de código abierto<sup>34</sup> enfrentan la necesidad de administrar el conocimiento organizacional y confrontan una disyuntiva al pretender hacerlo a partir de la concepción dual vigente.

La necesidad se hace evidente al considerar el tamaño del software de código abierto. Si utilizamos la relación<sup>35</sup> (1.1) para caracterizar el tamaño, tendremos por ejemplo que en el área de sistemas operativos una versión reciente, hecha pública en 2009, del kernel Linux conocida como linux-2.6.29 tiene un tamaño,  $T_l$ , de  $T_l = 7\ 707\ 587$  líneas de código abierto; de la misma forma, pero ahora en el área de arquitecturas estándares de cómputo distribuido para la integración de aplicaciones empresariales, con el estándar de la industria CORBA, la versión JacORB 2.3.1 tiene un tamaño,  $T_j$ , de  $T_j = 201\ 168$  líneas de código abierto en lenguaje “Java”. El kernel Linux, a pesar de ser una parte central, requiere de mucho más código —las líneas de código que se indican arriba

---

<sup>32</sup>El Manifiesto por el Desarrollo Ágil de Software establece: *Individuos e interacciones* sobre procesos y herramientas. *Software funcionando* sobre documentación extensiva. *Colaboración con el cliente* sobre negociación contractual. *Respuesta ante el cambio* sobre seguir un plan. Los autores notan que aunque valoran los elementos de la derecha, valoran más los de la izquierda —que están en itálicas. (Véase Beck *et al.*, 2001).

<sup>33</sup>*Managers in agile teams lend focus to team members' actions, facilitating a thriving self-organizing environment.*

<sup>34</sup>Porque lo requieren para prestar los servicios o proveer los bienes que son una parte esencial de su razón de ser, y lo hacen con el objetivo de transformarlo y aplicarlo para aprovechar oportunidades o resolver problemas concretos de su contexto social, histórico y cultural particular.

<sup>35</sup>Véase la sección 1.2. *La administración del conocimiento, una necesidad en las pequeñas organizaciones que desarrollan software, cuando se proponen adecuar software de código abierto* en la página 5.

contabilizan únicamente el código de todos los programas escritos en el lenguaje de programación denominado “C” que integran dicha versión— para poderse llamar sistema operativo y ser de utilidad a la mayoría de las organizaciones. Por ejemplo, la versión del sistema operativo basado en Linux y llamado Debian tiene un tamaño de prácticamente 68 millones de líneas de código abierto, escritas en múltiples lenguajes de programación (Geeknet, 2010). De forma similar, el ORB<sup>36</sup> y los elementos adicionales que se incluyen en la implementación del estándar CORBA hecha en JacORB, a pesar de ser también una parte central, requieren de más código para permitir la integración de aplicaciones en cualquier organización que desee utilizar dicho estándar. Por la relación (1.1) y el rango de tamaño usuales en el software de código abierto que usualmente exceden  $T_0$ , ejemplificados por  $T_i$  y  $T_j$ , se considera necesario administrar el conocimiento organizacional en las pequeñas organizaciones que desarrollan software, cuando el desarrollo requiere adecuar este software. Sin embargo, en ese caso confrontan una disyuntiva<sup>37</sup> estas organizaciones porque necesitan administrar el conocimiento organizacional y por tres razones no es posible hacerlo con la concepción y modelo de conversión vigentes<sup>38</sup>.

La primera, porque usualmente no se cuenta con el conocimiento explícito, en la forma de documentación del software, que propicie la adecuación del código en el que fue escrito. La segunda, porque no se cuenta también usualmente —de manera vinculante y directa— con los programadores que tienen el conocimiento tácito correspondiente para adecuarlo. Al expresar estas dos razones con el modelo de conversión vigente tenemos:

- La socialización es insuficiente para el caso en que es necesario realizar adecuaciones o localizaciones específicas. Este caso se explica porque no es el interés inmediato de los desarrolladores del software de código abierto resolver los problemas de la pequeña organización.

No obstante, se reconoce que la socialización es útil para para el caso que es necesario resolver fallas o problemas en el software de código abierto<sup>39</sup>. Este caso se explica porque resolver los errores de los programas usualmente si es de interés para los desarrolladores del software de código abierto.

Una alternativa reciente consiste en contratar —disponible en ciertos casos de software de

---

<sup>36</sup>Object Request Broker o “Corredor de Invocaciones entre Objetos”.

<sup>37</sup>En esta situación se considera que no se incurre en un falso dilema porque hay varias alternativas posibles. Véase, para mayor detalle sobre esta falacia Herrera, A. y Torres J. A. 2007, 59.

<sup>38</sup>Vía el modelo de conversión de conocimiento: a) Tácito a tácito (socialización), b) Tácito a explícito (externalización), c) Explícito a explícito (combinación), d) Explícito a tácito (internalización). Véase la sección 3.2. *El conocimiento como recurso tácito y explícito en la concepción vigente del proceso de administración del conocimiento organizacional*.

<sup>39</sup>*Les communautés de logiciels libres en sont un très bon exemple. Elles associent des développeurs, des utilisateurs experts et des novices. Les premiers apportent leur aide et leurs conseils aux deux autres, mais à l'inverse ces derniers apportent aux développeurs des informations sur les défauts et les inadaptations du logiciel (Flichy, 2008, 3)*. Sin olvidar que estas soluciones, que describe Flichy, ocurren con tiempos de respuesta que dependen de los desarrolladores del software y no del que confronta directamente el problema, en el que se detectó el error.

código abierto— alguna empresa que se dedique a proveer servicios especializados, que van desde resolver de manera oportuna los errores reportados hasta realizar adecuaciones o localizaciones específicas. Por los costos involucrados, esta opción es, sin embargo, usualmente sólo viable para organizaciones de mayor tamaño.

Se podría caracterizar la socialización en este caso como una relación en la que se observa también una disyuntiva de asimetría de información<sup>40</sup>.

- La externalización es insuficiente porque los integrantes de la pequeña organización no son los creadores del software y —sobre todo al inicio del proyecto— no tienen los recursos para intentar documentarlo. Como ya se ha afirmado, los creadores del software, por su parte, usualmente no lo documentan.

No obstante, se reconoce que hay algunos casos de software de código abierto en que se ha generado documentación, creada por algunos desarrolladores o por terceros, y que falta buen trecho por recorrer. Se puede afirmar que las adecuaciones más sofisticadas implican cambios a la arquitectura del software. También se registran esfuerzos en esta área para algunos casos de software de código abierto en que se ha generado alguna documentación sobre su arquitectura. (Véase por ejemplo [Brown y Wilson, 2012](#)). En este terreno, y respecto de la necesidad de conducir estudios empíricos para explorar la relación entre la arquitectura del software y los requerimientos de comunicación de los grupos de trabajo, en el contexto de la administración del conocimiento puede consultarse a [Ali, Beecham y Mistrik \(2010, 351\)](#).

- La combinación y la internalización son insuficientes por el tamaño y la falta de documentación del software.

No obstante, se reconoce que hay algunos casos en que la documentación creada por terceros para productos similares podría ser utilizada, con el debido cuidado en el nivel de similitud y diferencia existentes. Véase por ejemplo lo propuesto por [Brown y Wilson \(2011\)](#). Sin embargo, es evidente que aquí también falta buen trecho por recorrer.

La tercer razón tiene que ver con dos aspectos que surgen recurrentemente en la práctica administrativa del conocimiento organizacional con la concepción vigente. Primero, los creadores de esta concepción tomaron un curso que omitió la acción y privilegió una posición epistémica y ontológica; segundo, en la práctica se asocia al conocimiento organizacional con aspectos intangibles

---

<sup>40</sup>*Information asymmetry dilemma*. Para un esfuerzo en este sentido al interior de las organizaciones —como grupos de trabajo— véase [Nielsen 2008](#), 5-11. Sin embargo, consideramos que también en este trabajo permea la concepción dual vigente y, además, debemos notar que, para el caso del software de código abierto, la pequeña organización que desarrolla software —y sus grupos de trabajo— son sólo una parte de una comunidad mucho más amplia que cruza horizontalmente a las organizaciones y esto obliga a considerar aspectos adicionales a los ahí mencionados.

y con el criterio de rendimiento económico. Estos aspectos se sustentan —ceñidos a los autores estudiados— de la siguiente manera:

- Nonaka y Takeuchi reconocen la importancia de la acción y su relación esencial con el conocimiento<sup>41</sup>, sin embargo la omiten y se vuelcan en lo que denominan la dimensión epistemológica<sup>42</sup> del conocimiento que sustentan en la “expansión práctica” del trabajo de Polanyi<sup>43</sup>. También plantean el proceso de creación del conocimiento organizacional, considerando que el conocimiento es creado sólo por individuos y que la organización soporta y provee el contexto de dicha creación. Se percatan de esta peculiaridad del conocimiento y la llaman la dimensión ontológica<sup>44</sup>. Davenport y Prusak respecto del valor del conocimiento, afirman que se ubica cerca de la acción, que puede y debe ser evaluado por las acciones a las que conduce, pero que puede ser difícil trazar la ruta entre conocimiento y acción (Davenport y Prusak, 2000). Así, consideramos que se reconoce la importancia de la acción y que su omisión —y consecuente dificultad para enlazarla al conocimiento— muestran la insuficiencia de la concepción dual vigente.
- En la administración, se asocia al conocimiento con aspectos intangibles: “aunque incorpóreo, es efectivamente un activo generador de bienes” (Herrera, Ramírez y May, 2012, 273); “el capital invisible que poseen las organizaciones” (Solleiro *et al.*, 2009, 33). Le Blanc y Ermine intentan trascender este aspecto intangible y medir el capital intelectual o *knowledge capital*. De las hipótesis que han propuesto eligen la semiótica, en la que conjeturan que el conocimiento se percibe como un signo. Afirman entonces que el conocimiento es información que hace sentido en un contexto dado (Le Blanc y Ermine, 2007, 54). Esto, dicen, les permite resolver lo que identifican como una ambigüedad entre conocimiento e información. Con esta definición vinculan —vía su modelo formal *AIK*— el capital intelectual —*K*—, que reside fuera del sistema de administración del conocimiento —*S*—, con la base de conocimiento o *knowledge corpus* —*I*. Sin embargo, un primer problema es que a pesar de redefinir conocimiento, recurren de nueva cuenta a la concepción dual de Nonaka y Takeuchi donde *I* es conocimiento explícito; *K*, tácito. Este último vuelve a ser intangible, porque *K* queda fuera de *S* (Le Blanc y Ermine, 2007, 53). Un segundo problema es que para formalizar uno

---

<sup>41</sup>*knowledge is essentially related to human action* (Nonaka y Takeuchi, 1995, 58,59), véase también su respectiva nota de pie 2: (Nonaka y Takeuchi, 1995, 90).

<sup>42</sup>Al plantear que el conocimiento explícito es de tipo proposicional.

<sup>43</sup>*While Polanyi argues the contents of tacit knowledge further in a philosophical context, it is also possible to expand his idea in a more practical direction.* Y plantean que en el proceso de creación de conocimiento organizacional la “articulación de modelos mentales tácitos” es un factor clave: *the articulation of tacit mental models, in a kind of “mobilization” process, is a key factor in creating new knowledge.* (Véase Nonaka y Takeuchi, 1995, 60).

<sup>44</sup>Sin embargo, asumen que como proceso, la creación del conocimiento organizacional, amplifica el conocimiento creado por los individuos y lo “cristaliza como parte de la red de conocimiento de la organización” (Nonaka y Takeuchi, 1995, 59).

de los componentes críticos —el contexto— lo reducen, con el uso de grafos y la teoría de redes sociales, a la conexión entre los usuarios de la base de conocimiento *I* (Le Blanc y Ermine, 2007, 63-64). Beyerlein y Kennedy parten de la concepción dual para administrar el capital intangible; sin embargo, afirman que es imperativo encontrar maneras de desarrollar las diversas formas del capital intangible<sup>45</sup>; que un primer paso es hacerlos visibles vía mecanismos de registro y reporte<sup>46</sup>; que la importancia de hacer visible las diversas formas de capital intangible es aparente cuando las empresas reducen su tamaño: A mayor capital intelectual y social de la persona que deja la organización el último día de empleo, mayor la pérdida para la empresa (Beyerlein y Kennedy, 2008, 409). Considero que ignorar o reducir deliberadamente la individualidad, la cultura organizacional, el contexto y la situación social al recurrir a aspectos inobservables, intangibles, muestra la insuficiencia de la concepción dual, para la administración del conocimiento organizacional.

- Prevalece en la concepción vigente el criterio de rendimiento económico<sup>47</sup>, de competitividad excluyente, de capital intelectual, que privatiza el conocimiento como recurso. Para los esfuerzos administrativos de privatizar el conocimiento como capital intelectual, es notable que el productor del capital intelectual pueda salir caminando de la organización, inclusive que no sea un activo de la organización<sup>48</sup>. En este contexto, sin embargo, para el desarrollo de software de código abierto la administración del conocimiento no es guiada por el criterio de capital intelectual<sup>49</sup>. Así, para nuestro caso de interés, el criterio de rendimiento económico

---

<sup>45</sup>Ellos caracterizan seis: Procesos internos y externos de conversión —*The way assets are used, grown, stored, and combined determines how much value is produced*—, intelectual —*What you know*—, social —*Who you know*—, organizacional —*How you do it*—, colaborativo —*How you work together*—, humano —*How well you can do it*— (Beyerlein y Kennedy, 2008, 402-403).

<sup>46</sup>Como se registra en este trabajo, esto usualmente se realiza con una perspectiva dual que aborda al conocimiento —en su faceta explícita— como información y consecuentemente utiliza de manera preponderante las tecnologías de información y comunicaciones para vincularlo con su faceta tácita, intangible.

<sup>47</sup>En otro contexto, de gestión ambiental que podríamos relacionar como evidencia de que el criterio económico no es necesariamente el de mayor relevancia, Rivas propone hacer una distinción de términos que sea más clara: “de un modo simple podríamos decir: administración es aplicar racionalmente el proceso administrativo para conseguir rentas empresariales. Gestión es lograr los objetivos conciliando intereses divergentes en ambientes conflictivos y complejos” (Rivas *et al.*, 2009, 15).

<sup>48</sup>Drucker lo expresa así: *An accounting system which would show people as “capital investments” would therefore make a major difference. But it is not easy to see how one could show people as assets on the books. An asset is by definition something one can sell and something that has a value when a company goes into liquidation. But a company does not own people. And an asset which can give notice and leave is not an asset in any sense of the word. There are equally serious practical objections. How does one, for instance, measure the return on training? Still, the idea has merit. It clearly would be highly desirable for managers to be led by their own measurements and their own controls to acting on their profession that “people are our greatest asset”* (Drucker, 1974, 308).

<sup>49</sup>Para organizaciones de mayor tamaño, Federman afirma: *Radical changes on some of the basic assumptions of conventional practices may be required to translate Free/Libre/Open Source Software (FLOSS) approaches to corporate management* (Federman, 2006, 89); Goldman y Gabriel reflexionan que para estas organizaciones la idea de dar el trabajo realizado por sus desarrolladores a cualquier persona —posiblemente incluyendo a sus competidores— parece violar el sentido común (Goldman y Gabriel, 2005, 1).

vigente en la práctica es insuficiente pues es necesario utilizar criterios adicionales, entre los que se encuentran promover como un bien público el conocimiento y el software de código abierto<sup>50</sup>.

La pequeña organización enfrenta una disyuntiva a la hora de administrar el conocimiento organizacional con la concepción y métodos vigentes, por las razones expuestas sobre su insuficiencia. Considero que por ello no puede administrar con éstos, luego que ha de hacerlo con otros.

### 3.5. Conclusión

En el análisis de primer nivel he descrito la concepción vigente de conocimiento organizacional en el campo de la administración; he expuesto que se concibe este conocimiento como tácito y explícito, y que en esta concepción se basa la metodología de la teoría de creación del conocimiento organizacional. Finalmente, he planteado que esta metodología es frecuentemente adoptada en las organizaciones que desarrollan software para satisfacer su necesidad de administrar el conocimiento.

En un segundo nivel interpreté y describí las dificultades, reflexiones y planteamientos que desde diversos campos disciplinarios se han hecho sobre el conocimiento organizacional y su administración.

Finalmente, al analizar el caso de las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto, argumenté que la concepción vigente de conocimiento organizacional y su consecuente método de administración son insuficientes.

Concluyo como resultado de esta investigación sobre el estado del arte, que es recomendable formular una nueva concepción de conocimiento organizacional y su consecuente modelo de administración del conocimiento, a fin de que se contemple la necesidad de las pequeñas organizaciones que desarrollan software, cuando pretenden adecuar software de código abierto.

---

<sup>50</sup>Recordemos que éste es el atributo común con el que caracterizamos este software. Véase la sección [1.1 El software de código abierto](#) en la página [5](#). Por lo que la administración establece, en este caso, como uno de sus objetivos organizacionales dicha promoción. Tenemos en este caso criterios que pueden, de forma pertinente, complementar e inclusive reemplazar al de carácter económico. Por citar algunos tenemos los de carácter moral, los de justicia, equidad, democracia, estética y libertad.

# Capítulo 4

## Marco Teórico Conceptual

La aplicación de la definición dual del conocimiento organizacional planteada por Nonaka y Takeuchi enfrenta diversos problemas que impactan la administración del conocimiento. En particular, al analizar el caso de las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto, argumenté que enfrentan una disyuntiva en su necesidad de administrar el conocimiento organizacional con la concepción y métodos vigentes; y considero, por la razones expuestas sobre la insuficiencia de estos últimos, que esa es la razón por la que no pueden administrar con ellos, luego entonces que han de hacerlo con otros. En este contexto, formulo<sup>1</sup> una nueva concepción de conocimiento organizacional y su consecuente modelo de administración del conocimiento, a fin de que se contemple la necesidad de las pequeñas organizaciones que desarrollan software, cuando pretenden adecuar software de código abierto.

### **4.1. El conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas y su consecuente modelo de administración**

En el capítulo anterior se hace evidente que en las Ciencias de la Administración prevalece una concepción dual del conocimiento organizacional, al diferenciarlo en tácito y explícito. Se constata que, por una parte, al tácito se le caracteriza como implícito, intangible, difícil de separarlo de su creador y, por tanto, hace difícil formalizar y codificar ese “capital invisible que poseen las organizaciones”; mientras que por la otra, se exalta al conocimiento explícito como la expresión de ventajas competitivas y como sustento de la privatización del conocimiento, vía el capital inte-

---

<sup>1</sup>En octubre de 2012 publiqué, con el invaluable apoyo de la Dra. Lucía Patricia Carrillo como coautora, mi avance en la investigación del marco teórico en una ponencia para el *XVII Congreso Internacional en Contaduría, Administración e Informática*. El texto completo de la ponencia se encuentra en ([Orozco y Carrillo, 2012](#)).



lectual de la organización. Se ha mostrado así que esta concepción permea el enfoque vigente del proceso de la administración del conocimiento.

En este capítulo, se presenta los antecedentes y una propuesta para concebir al conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización. Considero que esta concepción apoyará la reformulación de métodos, para la administración del conocimiento, por dos razones. Primero, porque permite conjeturar que el conocimiento organizacional es observable como un sistema complejo de acciones eficaces, eficientes y reiteradas, que emerge al ser articulado y puesto en ejecución por los diversos agentes que conforman la organización vista esta última también como un agente. Segundo, porque considero que concebir y observar así al conocimiento organizacional soporta la creación de diversos modelos y métodos de administración del conocimiento que se pueden adecuar de mejor forma al dominio particular en el que opera la organización.

La sección 4.2 presenta los antecedentes de una nueva concepción de conocimiento organizacional. A continuación, la sección 4.3 contiene la nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización. Dicho sistema complejo emerge cuando el sistema de acciones en ejecución cumple las condiciones e incluye las causalidades que ahí se describen. Finalmente, la sección 4.4 aplica esta concepción de conocimiento organizacional en la formulación de un modelo para la administración del conocimiento en las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto.

## 4.2. Antecedentes de una nueva concepción de conocimiento organizacional

El problema de la definición del conocimiento en relación con la acción nos remite de nueva cuenta a Platón como primera fuente occidental, pero ahora al diálogo de Sócrates con Menón<sup>2</sup>, escrito que enfoca la naturaleza de la virtud y a si ésta se adquiere por la práctica, si es enseñada o si está en la naturaleza de las personas. Describe al conocimiento en función de la opinión verdadera<sup>3</sup>: “La opinión verdadera libre no es valiosa por lo escurridiza; cuando se ata a la causa, es conocimiento. Por ello —dice Platón en boca de Sócrates— es máspreciado el conocimiento

---

<sup>2</sup>Que escribe ca. 386-382 a.C.

<sup>3</sup> [...] *and thus are of no great value until one makes them fast with causal reasoning. And this process, friend Meno, is recollection, as in our previous talk we have agreed. But when once they are fastened, in the first place they turn into knowledge, and in the second, are abiding. And this is why knowledge is more prized than right opinion: the one transcends the other by its trammels* (véase Plato, 1967, 98a).



que la opinión verdadera”<sup>4</sup>. Finalmente, lo relaciona con la virtud en tanto que ambos pueden guiar la acción virtuosa. Concluye, sin embargo, que la virtud no es conocimiento porque no es algo que pueda enseñarse (Veáse [Plato, 1967](#), 98a-99a).

Afirma Polanyi que Platón, en el “Menón” muestra de manera concluyente que si todo el conocimiento es explícito, *i.e.*, clara y proposicionalmente establecido, entonces no podemos conocer un problema o buscar su solución. También afirma que de esta paradoja, de la existencia de los problemas y de los descubrimientos que se pueden realizar para resolverlos, se sigue que “podemos conocer más de lo que podemos decir”. Y que el tipo de conocimiento tácito que resuelve esta paradoja descansa en el aspecto funcional —que podríamos no ser capaces de especificar— y en los otros tres aspectos —fenoménico, semántico y ontológico— en una interrelación que no podemos definir ([Polanyi and Sen, 2009](#), 24).

Posteriormente, Polanyi en su texto *La Revolución Científica* caracteriza al conocimiento así: Hay dos tipos de conocimiento que invariablemente participan en cualquier acto para conocer una entidad totalmente: 1) Un conocer al atender a la entidad en su totalidad, y 2) Un conocer al atender a sus aspectos particulares. Y que la oscilación que va del detalle de las particularidades a la integración de la totalidad es el camino principal (*royal road*) para profundizar nuestro entendimiento, nuestra comprensión de la entidad. Este, nos dice, es el acto central de conocer. Lo que no se comprende no se conoce. Nos da a continuación el ejemplo de un experto que puede identificar 800 000 especies de insectos y que debe utilizar un número muy amplio de pistas que él no puede identificar por separado en sí mismas. De ahí que la zoología y la botánica no puedan ser aprendidas a partir de páginas impresas, como tampoco puede serlo la medicina, ya que se requieren muchas horas de enseñanza práctica para realizar el proceso de asimilar el conocimiento a nuestro cuerpo (*indwelling*). Entonces, nos dice más adelante, es posible identificar el conocimiento de algo como el tipo de conocimiento que tenemos de nuestro propio cuerpo, viviendo en él. Este tipo de conocimiento no es una relación Yo-eso (*I-it*), más bien es una forma de existir, una manera de ser. Lo podemos llamar una relación Yo-conmigo (*I-Myself, I-Me*) ([Polanyi, 1997](#), 332-337).

Identifica Polanyi como una cuestión crucial el “objetivo declarado de la ciencia moderna para establecer un conocimiento estrictamente objetivo en el que se eliminen todos los elementos personales.” Para ello muestra que “el proceso de formalizar todo el conocimiento para excluir cualquier conocimiento tácito se autorefuta”, pues una teoría puede ser construida únicamente si se utiliza conocimiento tácito previo y puede funcionar como teoría solo dentro del acto de conocer tácitamente ([Polanyi and Sen, 2009](#), 20-21).

Recientemente, y desde una perspectiva filosófica, Villoro se propone analizar, clarificar y sis-

---

<sup>4</sup>Da pie, así, a la denominada “Paradoja de Menón” que Amartya Sen expresa de la siguiente forma: “La búsqueda de ese tipo de conocimiento es un absurdo, debido a que ya conoces de antemano lo que buscas —y en ese caso no necesitas buscar— o no conoces lo que estas buscando —y en ese caso no puedes esperar encontrarlo” ([Polanyi and Sen, 2009](#), xi). Paradoja que Platón resuelve al recurrir al recuerdo *recollection* (de vidas pasadas).

tematizar el concepto de conocimiento<sup>5</sup> (Villoro, 2008, 12). Establece que en castellano existen dos verbos que no suelen usarse con el mismo significado: “conocer” y “saber”; que esta distinción semántica orienta para descubrir otra de importancia epistemológica, que no se ha presentado con claridad debido a que los principales análisis de los conceptos epistémicos han sido escritos en inglés, lengua en que esa distinción se ha perdido. Mientras que asocia al saber con predicar, al conocer lo asocia con la experiencia directa, con múltiples experiencias variadas, capaces de ser integradas en una unidad. Por ello, el conocimiento puede ser más o menos complejo, más o menos rico (Villoro, 2008, 197-199). Así, afirma: El conocimiento en general es un estado disposicional a actuar, adquirido, determinado por un objeto o situación objetiva aprehendidos, que se acompaña de una garantía segura de acierto (Villoro, 2008, 221). Lo asocia con la práctica al decir que es la razón que funda nuestros saberes y que funciona como motivo que determina nuestros procesos de justificación. Define a la práctica como: 1) acción intencional objetiva; 2) actividad material transformadora y adecuada a fines; 3) actividad material transformadora, adecuada a fines, condicionada por las relaciones sociales y que responde a intereses igualmente sociales (Villoro, 2008, 251-252).

También, afirma que en la realización práctica de los fines que movieron a conocer algo se comprueba ese conocimiento; que la práctica es entonces criterio de verdad para probar, para contrastar el conocimiento<sup>6</sup> (Villoro, 2008, 253). Sustenta entonces que la práctica es el único criterio de verdad y que asociar los conceptos de práctica y teoría como “práctica teórica” no es útil porque elimina una distinción importante: ¿en qué sentido podríamos llamar “práctica” a una actividad privada, inobservable por otros sujetos, que sólo acontece en la mente de quien conoce? Respecto de los objetivos del conocer, establece que el conocimiento no requiere de la supresión de los fines interesados que lo motivan, sino sólo de su no interferencia en el proceso de justificación. Así, sólo el interés por conseguir eficacia para nuestras acciones y encontrar sentido a la vida puede explicar el conocimiento (Villoro, 2008, 258, 261, 262).

Concluye Villoro: La concepción del conocimiento como un proceso interesado, ligado a fines prácticos, motivado por deseos y condicionado por situaciones sociales, introduce la voluntad en él. El conocimiento no es un fin en sí mismo. Responde a la necesidad de hacer eficaz nuestra acción en el mundo y darle un sentido. Su logro es una meta regulativa en la realización y el

---

<sup>5</sup>Reconoce la influencia de Platón, en particular del diálogo de Sócrates con Menón, para definir de manera más amplia al conocimiento (Villoro, 2008, 20).

<sup>6</sup>Concuerda aquí con la tesis de Marx: “El problema de si al pensamiento humano se le puede atribuir una verdad objetiva, no es un problema teórico, sino un problema práctico. Es en la práctica donde el hombre tiene que demostrar la verdad, es decir, la realidad y el poderío, la terrenalidad de su pensamiento. El litigio sobre la realidad o irrealidad de su pensamiento que se aísla de la práctica, es un problema puramente escolástico” (Marx, 1975, 24). Quizá también con la de Tsetung: “Practicar, conocer, practicar otra vez y conocer de nuevo. Esta forma se repite en infinitos ciclos, y, con cada ciclo, el contenido de la práctica y del conocimiento se eleva a un nivel más alto. Esta es en su conjunto la teoría materialista dialéctica del conocimiento, y ésta es la teoría materialista dialéctica de la unidad entre el saber y el hacer” (Tsetung, 1975, 37).

perfeccionamiento de todo hombre y la especie<sup>7</sup> (Villoro, 2008, 268, 296-297).

Por otra parte, Olivé afirma que el conocimiento se refiere a los hechos, que la moral está conformada por normas y valores para la evaluación de acciones. Las une en lo que denomina una práctica cognitiva: Es un sistema dinámico constituido por un conjunto de seres humanos quienes a su vez dan lugar a un complejo de acciones orientadas por representaciones —que van desde modelos y creencias hasta complejas teorías científicas— y que tienen una estructura axiológica, es decir, normativo-valorativa. Esta estructura axiológica no está formada por un conjunto rígido de normas ya constituidas que los agentes deben entender y en su caso “internalizar” para actuar conforme a ellas. Estas prácticas cognitivas se manifiestan entonces en una serie de acciones (investigar, observar, etc.) y se requiere valorar tanto a éstas como a sus resultados. Así, una comunidad de expertos, junto con los valores y normas bajo los cuáles evalúan y toman decisiones acerca de cómo actuar, aunados al sistema de acciones que de hecho realizan, constituyen una práctica cognitiva (Olivé, 2008, 93). A esta comunidad de expertos, Olivé la denomina como una red cognitiva o epistémica que caracteriza como dinámica y que incluye como elementos a) agentes intencionales con capacidades y con propósitos comunes; b) un medio del cual forma parte la red y en donde los agentes interactúan con otros objetos y otros agentes; y c) un conjunto de objetos (incluidos otros seres vivos) que forman parte también del medio (Olivé, 2008, 130). Asimismo, defiende la tesis del conceptualismo naturalista en la que la percepción tiene “un contenido organizado conceptualmente” y, por tanto, que “toda observación —incluida la percepción sensorial— sólo es posible mediante la aplicación de un sistema de conceptos”<sup>8</sup>, y propone<sup>9</sup> la noción de *concepto empírico primitivo*: “Diremos que un organismo dispone de un concepto, del concepto del objeto correspondiente, si el organismo tiene la capacidad de *representarse* ese objeto. Cuando se trata de conceptos de objetos que el tipo de organismo en cuestión puede percibir por medio de los sentidos, diremos que se trata de un *concepto empírico primitivo*. Así, los conceptos empíricos primitivos tienen su origen en el funcionamiento adecuado de los sistemas de percepción sensorial y, por consiguiente, *son anteriores a las creencias*. No es que las creencias conceptualicen la experiencia; más bien *las creencias son posibles porque la experiencia esta conceptualizada*, tiene ya un contenido conceptual” (Olivé, 2008, 207)<sup>10</sup>. Respecto de este representarse, lo plantea como una relación tripartita entre lo representado —hechos, procesos—, el representante —teorías, modelos—, y los agentes

---

<sup>7</sup>Propone para ello una ética que permite liberar de la sujeción a intereses, de personas o de grupos, las creencias colectivas. Que es a la vez condición de un pensamiento racional (Villoro, 2008, 268-296).

<sup>8</sup>Reconoce Olivé: “La herencia, por supuesto, es de añeja prosapia kantiana: no es que la percepción sin conceptos sea ciega, es que no hay observación ni percepción sin conceptos. No es que los conceptos sin percepción sean vacíos, es que no es posible que existan los conceptos si no hay percepción” (Olivé, 2008, 205).

<sup>9</sup>No sin reconocer que hay poco acuerdo sobre la respuesta a la pregunta ¿Cómo entender las nociones de concepto, de poseer un concepto, y de contenido conceptual y contenido no conceptual? (Olivé, 2008, 206).

<sup>10</sup>El énfasis adicional es mío, para contrastar con la propuesta sobre el lenguaje de Maturana y Varela que describo más adelante, en la página 36 y la nota de pie asociada.

con sus actitudes y creencias. Así entiende por agente perceptual a cualquier sistema capaz “de percibir algunos objetos, es decir, de tener una *representación de un objeto* como algo distinto de otras entidades en el mundo<sup>11</sup> [...] “El agente perceptual es una entidad más en el mundo, de manera que interactúa causalmente con el resto de los objetos en él. La *representación de un objeto* o de un estado de cosas en el mundo incluye la relación entre el objeto representado y el estado del agente perceptual que indica la presencia de un cierto objeto o estado de cosas” (Olivé, 2008, 209). ... “Podemos distinguir entre las representaciones en donde el estado que indica la presencia del objeto lo produce causalmente el objeto en el mundo y aquellas donde ese estado es producido por estímulos internos del agente, por ejemplo por la memoria. En el primer caso hablamos de la percepción del objeto y en el segundo de la sensación del objeto” (Olivé, 2008, 210). [...] “En condiciones normales una persona aprende a distinguir las representaciones de un objeto donde el estímulo proviene por ejemplo de su memoria, de las representaciones del objeto donde el estado indicativo correspondiente lo causa la presencia del objeto real. Pero a veces hay duda y la persona debe realizar una serie de acciones (oler, tocar, manipular) para cerciorarse que la representación la generó la presencia del objeto real, es decir, que se trata de una percepción del objeto y no sólo de una sensación de él. Éste es uno de los puntos de origen de las normas epistémicas para cualquier práctica cognitiva que involucre conceptos empíricos primitivos” (Olivé, 2008, 210). Así, afirma: “Que un agente perceptual de la clase *X* disponga de un concepto empírico primitivo, digamos del concepto de *O*, entonces, es tener la capacidad de representarse correctamente a un objeto de la clase *O*. Y tener de hecho una representación (correcta o incorrecta) de un objeto *o* es *aplicar el concepto de O*” [...]. “La cuestión de cómo los perros, o los humanos, aprenden a discriminar unas representaciones de otras (identificar objetos y por tanto aplicar conceptos) es una cuestión de investigación empírica” (Olivé, 2008, 214). “Cuando aparece el lenguaje proposicional, evolutivamente hablando, los hablantes utilizan determinados términos para referirse a los miembros de una cierta clase de objetos (*O*) de los que pueden tener representaciones. Pero esa clase existe antes del uso de los términos.” [...] “Aunque es cierto que con el lenguaje proposicional se crean conceptos (empíricos y no empíricos) mucho más complejos que los primitivos conceptos empíricos a los que aquí nos referimos, los conceptos empíricos primitivos no han surgido con el lenguaje proposicional, más bien constituyen condiciones de posibilidad para que surja ese lenguaje” (Olivé, 2008, 214).

Respecto de las prácticas, normatividad y el acceso epistémico a la realidad, afirma: “El mundo no es como lo concibe la tradición epistemológica moderna que supone que somos sujetos que nos enfrentamos a una realidad conformada por objetos que queremos conocer y manipular, pero

---

<sup>11</sup>Nos dice: “Entendemos por agentes perceptuales cualesquiera sistemas capaces de recibir información de su entorno (estímulos), de procesar esa información, de reaccionar a ella y de interactuar con el entorno” (Olivé, 2008, 209).

que existen como objetos independientes de nosotros, de nuestros recursos conceptuales y de nuestras prácticas. No, esos objetos forman parte de un complejo sistema en donde estamos nosotros también y donde ambos nos afectamos recíprocamente y donde ellos nos constituyen a nosotros y nosotros a ellos en un constante intercambio, en un flujo y reflujo. Esta interacción e interdependencia no es sólo epistémica sino también ontológica, lo que somos depende de nuestro medio y de sus objetos, y el medio y sus objetos también dependen de nosotros” (Olivé, 2008, 215).

Advierte, sin embargo, que las condiciones materiales y sociales en las que se constituyen los medios varían enormemente, lo que hace diferente una red cognitiva de otra. Esto lo conduce a afirmar que las representaciones, las normas y los valores que forman parte de una práctica adecuada a un medio, así como la estructura axiológica que permite calificar como racional o no una decisión o una acción, tanto como los objetos relativos a esa práctica, difieren de las que constituyen otras prácticas adecuadas a distintas situaciones (Olivé, 2008, 225).

Plantea que lo que puede explicar esta diversidad axiológica, este carácter plural de la racionalidad es —desde una perspectiva naturalizada— el modelo siguiente: Si, las normas y valores que sirven como estándares para elegir creencias y teorías (normas epistémicas) tienen su origen en los procedimientos y acciones que los agentes miembros de la práctica deben realizar para saber que una determinada representación lo es de un objeto o situación en el mundo que da lugar a una creencia objetiva la racionalidad teórica y la práctica, entonces eso es lo que justifica su creencia en cierto estado de cosas en el mundo, es decir, su disposición a actuar como si ese estado de cosas en el mundo realmente existiera (Olivé, 2008, 224-225). Así, concluye respecto del conocimiento que la concepción naturalista permite reconocer la pluralidad de formas de conocer y de interactuar con la realidad, pero de que no “todo vale” porque la realidad impone constreñimientos, también porque las prácticas imponen una restricción social (Olivé, 2008, 227).

Desde una perspectiva biológica del entendimiento humano, Maturana y Varela proponen: “Conocer es acción efectiva, es decir, efectividad operacional en el dominio de existencia del ser vivo” (Maturana y Varela, 2003, 15) y que “reconocemos conocimiento cada vez que observamos una conducta efectiva (o adecuada) en un contexto señalado, es decir, en un dominio que definimos con una pregunta (explícita o implícita) que formulamos como observadores” (Maturana y Varela, 2003, 115). Lo que explica esta propuesta se soporta a su vez en cuatro conceptos que denominan como: La organización autónoma del ser vivo<sup>12</sup>, la deriva filogenética<sup>13</sup>, la deriva ontogenéti-

---

<sup>12</sup>En la que “los fenómenos que generan en su operar dependen de su organización y de cómo ésta se realiza y no del carácter físico de sus componentes que sólo determinan el espacio de su existencia”. Esto se considera su *fenomenología biológica* o *autopoiesis*. Que posteriormente utilizan para caracterizar las unidades o sistemas autopoiéticos (Maturana y Varela, 2003, 32).

<sup>13</sup>La filogenia es la sucesión de formas orgánicas emparentadas secuencialmente por relaciones reproductivas (Maturana y Varela, 2003, 69).

ca<sup>14</sup> y el acoplamiento estructural<sup>15</sup>. De esta propuesta deducen diversos fenómenos<sup>16</sup> y realizan observaciones adicionales sobre: fenómenos sociales<sup>17</sup>, comunicación<sup>18</sup>, dominios lingüísticos<sup>19</sup>, lenguaje<sup>20</sup> y autoconciencia<sup>21</sup> (Maturana y Varela, 2003, 15, 121-153).

Maturana y Varela plantean que “un ser vivo se conserva como unidad bajo continuas perturbaciones del medio y de su propio operar”, que en los organismos “el sistema nervioso genera una dinámica conductual a través de generar relaciones de actividad neuronal interna en su clausura operacional”, que “el sistema vivo, a todo nivel, está organizado de manera de generar regularidades internas”. Por ello afirman que “en el dominio del acoplamiento social y la comunicación<sup>22</sup>, se produce el mismo fenómeno, sólo que la coherencia y la estabilización de la sociedad como unidad se producirán esta vez mediante los mecanismos hechos posibles por el operar lingüístico y su am-

---

<sup>14</sup>La ontogenia es la historia de cambio estructural de un ser vivo particular. Sin que este pierda su organización (Maturana y Varela, 2003, 49).

<sup>15</sup>Acoplamiento estructural es lo que permite al ser vivo, en ambas derivas, conservar la adaptación al medio ambiente. Es un proceso continuado de cambios de estado gatillados por mutuas perturbaciones entre la unidad autopoiética y el medio, en la que todos los cambios —que se dan en forma continua— están determinados estructuralmente. Es esencial que el observador los distinga como operacionalmente distintos, independientes y entre los cuales se da la congruencia estructural necesaria (Maturana y Varela, 2003, 64, 67).

<sup>16</sup>Que resultan de la “coordinación conductual en las interacciones recurrentes entre seres vivos y de la coordinación conductual recursiva sobre la coordinación conductual” (Maturana y Varela, 2003, 15, 137).

<sup>17</sup>Que caracterizan como acoplamientos de tercer orden (Maturana y Varela, 2003, 121). En que las unidades autopoiéticas resultantes, aunque transitorias, generan una fenomenología particular en la que “los organismos participantes satisfacen sus ontogenias individuales fundamentalmente mediante acoplamientos mutuos en la red de interacciones recíprocas que conforman al constituir las” (Maturana y Varela, 2003, 129). Enfatizan que los organismos y los sistemas sociales son casos opuestos en la serie de metasistemas —autopoiéticos— formados por la agregación de sistemas celulares de cualquier orden: “El organismo restringe la creatividad individual de las unidades que lo integran, pues éstas existen para éste; el sistema social humano amplía la creatividad individual de sus componentes, pues éste existe para éstos” (Maturana y Varela, 2003, 132).

<sup>18</sup>“Como observadores, designamos como *comunicativas* las conductas que se dan en un acoplamiento social, y como *comunicación*, la coordinación conductual que observamos como resultado de ella” (Maturana y Varela, 2003, 129). Definen conducta cultural como la estabilidad transgeneracional de configuraciones conductuales adquiridas ontogénicamente en la dinámica comunicativa de un medio social (Maturana y Varela, 2003, 133).

<sup>19</sup>Un dominio lingüístico de un organismo lo conforman todas sus conductas lingüísticas. Éstas son conductas comunicativas ontogénicas, que se dan en un acoplamiento estructural ontogénico entre organismos. Se pueden describir en términos semánticos por un observador: “Toda vez que un observador describe las conductas de interacción entre organismos como si el significado que él asume que ellas tienen para los participantes determinasen el curso de tales interacciones. [...] Los dominios lingüísticos son, en general, variables y cambian a lo largo de las ontogenias de los organismos que los generen” (Maturana y Varela, 2003, 138).

<sup>20</sup>Es una deriva cultural, una armazón que se constituye con lo que se dispone en cada momento. “Operamos en el lenguaje cuando un observador ve que tenemos como objetos de nuestras distinciones lingüísticas elementos de nuestro dominio lingüístico”. Es recursivo —reflexión lingüística— porque permite describirnos a nosotros mismos y a nuestra circunstancia (Maturana y Varela, 2003, 139).

<sup>21</sup>La reflexión lingüística —el operar recursivo del lenguaje— es lo que permite dar coherencia descriptiva a nuestra deriva ontogénica. Da origen a la conciencia al mantener “una continua recursión descriptiva que llamamos ‘yo’, que nos permite conservar nuestra coherencia operacional lingüística y nuestra adaptación en el dominio del lenguaje” (Maturana y Varela, 2003, 152-153).

<sup>22</sup>Que denominan como “trofolaxis lingüística” en los humanos. Con este término establecen un símil con el acoplamiento conductual observable en los insectos sociales, vía el intercambio de sustancias químicas, denominado trofolaxis (Maturana y Varela, 2003, 124-125).



pliación en el lenguaje”. Las palabras, entonces, se consideran también como acciones y no cosas que “pasan de aquí para allá”. De esta forma “el lenguaje no fue nunca inventado por un sujeto solo en la aprehensión de un mundo externo, y no puede, por lo tanto, ser usado como herramienta para revelar un tal mundo” [...] “nos realizamos en un mutuo acoplamiento lingüístico, no porque el lenguaje nos permita decir lo que somos, sino porque somos en el lenguaje en un continuo ser en los mundos lingüísticos y semánticos que traemos a la mano con otros” y “nos encontramos a nosotros mismos en este acoplamiento, no como el origen de una referencia ni en referencia a un origen<sup>23</sup>, sino como un modo de continua transformación en el devenir del mundo lingüístico que construimos con los otros seres humanos” (Maturana y Varela, 2003, 154-155).

Consideran también que es importante no confundir dos niveles<sup>24</sup> fenoménicos para la cabal comprensión de los fenómenos sociales: El primero es que los individuos como componentes de la unidad grupal<sup>25</sup> son irrelevantes en su individualidad y todos ellos son, en principio, reemplazables por otros que puedan cumplir sus mismas relaciones; el segundo, es que para los componentes como seres vivos, en cambio, su individualidad es su condición de existencia.

Afirman, finalmente, que la identidad entre acción y conocimiento tiene un carácter ético porque tiene lugar en el dominio social: “cualquier cosa que destruya o limite la aceptación de otro junto a uno, desde la competencia hasta la posesión de la verdad, pasando por la certidumbre ideológica, destruye o limita el que se dé el fenómeno social, y por tanto lo humano, porque destruye el proceso biológico que lo genera” (Maturana y Varela, 2003, 163-164).

García hace un planteamiento constructivista del problema del conocimiento y sin definirlo lo

---

<sup>23</sup>Consideramos que hace sentido —para contrastar esta posición— citar aquí a Descartes, en la segunda de sus *Meditaciones metafísicas*, cuando establece el fundamento de todo el conocimiento en una proposición necesariamente verdadera: “esta proposición *Yo soy, yo existo*, es necesariamente verdadera cada vez que la profiero o que la concibo”; “Yo soy, yo existo; es cierto. Pero ¿durante cuánto tiempo? Ciertamente, mientras pienso”, y concluye: “Soy, pues, una cosa verdadera, y verdaderamente existente; pero ¿qué clase de cosa? Dicho esta: una cosa pensante” (Descartes, 1997, 22;24). Posición usualmente abreviada como *Cogito, ergo sum* y a la que adiciona a Dios, en su retorno hiperbólico, para fundamentar el conocimiento como describimos brevemente en la página 12. También hace sentido citar a Olivé que en su proceso argumental para proponer entender los conceptos naturalistamente, cuando refiere la propuesta de Heck, R. G. que reza “tenemos que *aprender a vivir* con la incómoda idea de que si bien nuestros sistemas perceptuales nos abren el mundo cuando todo va bien, el que lo hagan es algo contingente —de lo cual debemos agradecer a Dios o a la evolución, o a ambos [...] (Heck, 2000, 522)”. Afirma: “Mantengamos a Dios y el naturalismo se va a pique, por lo menos el naturalismo en la línea de Darwin y en la forma en que se desarrolló en la epistemología en los últimos 50 años. Quitemos a Dios y tenemos una pura y limpia posición que podemos llamar ‘fundamentista naturalista’, digna de principios del siglo XXI” (Olivé, 2008, 203). Posteriormente, sustenta como complemento, un realismo débil sobre su propuesta de “conceptos empíricos primitivos”, en una posición que denomina como conceptualismo naturalista (Olivé, 2008, 205-210).

<sup>24</sup>Y que en esta consideración no hay contradicción entre los niveles porque los componentes se realizan, en su individualidad, como miembros del grupo. Que “la existencia de lo vivo en la deriva natural, tanto onto como filogénica, no se da en la competencia sino en la conservación de la adaptación, en un encuentro individual con el medio que resulta en la sobrevivencia del apto”. Para el caso del dominio humano es pertinente recordar, que el lenguaje es el modo de acoplamiento social (Maturana y Varela, 2003, 131).

<sup>25</sup>“Para la cual, en su dinámica como unidad, la conservación de la adaptación es también necesariamente válida en su dominio de existencia” (Maturana y Varela, 2003, 131).

caracteriza como una totalidad relativa<sup>26</sup>. Lo propone como un proceso que toma sentido en un contexto social y cuyos “grados” o “niveles” también adquieren significado en dicho contexto. Afirma que este proceso constructivo del conocimiento es inseparable de los mecanismos con los cuales se desarrolla. Por ello no da una definición general de lo que es “conocimiento” (García, 2000, 39-40; 48). Sin embargo, ocho años después, define al conocimiento como “un fenómeno social intersubjetivo —y afirma— que conocer es establecer relaciones en una materia prima que es provista por la experiencia pero cuya organización depende del sujeto cognoscente” (García, 2008, 43).

En el dominio de los sistemas complejos, García establece dos condiciones que caracterizan a un sistema como complejo: “a) Las funciones de los elementos (subsistemas) no son independientes; esto determina la interdefinibilidad de los componentes; b) el sistema como totalidad es abierto, es decir, carece de fronteras rígidas; está inmerso en una realidad más amplia con la cual interactúa por medio de flujos de materia, energía, recursos económicos, políticas regionales, nacionales, etc.” (García, 2008, 143-144).

Casi diez años antes, Piaget y García establecen que “una acción en sí misma no es ni verdadera ni falsa, y no se evalúa sino en términos de eficacia o utilidad con respecto a un objetivo.” Y que “existe una lógica de significaciones que precede la lógica formal de los enunciados; dicha lógica de significaciones está fundada sobre implicaciones entre significaciones o, lo que es lo mismo, sobre implicaciones entre acciones” (Piaget, García *et al.*, 1997, 149).

Morin afirma<sup>27</sup>: “Consideremos una organización tal como una empresa que se sitúe en el mercado. Produce objetos o servicios, cosas que se vuelven exteriores y entran en el universo del consumo.” [...] “Organizando la producción de objetos y de servicios, la empresa se auto-organiza, se auto-mantiene, si es necesario se auto-repara y, si las cosas van bien, se auto-desarrolla desarrollando su producción.” [...] “La complejidad aparece en ese enunciado: se producen cosas y se auto-produce al mismo tiempo; el productor mismo es su propio producto.” identifica tres causalidades<sup>28</sup> que “se reencuentran en todos los niveles de organización complejos”. Y concluye: “La empresa, organismo viviente, se auto-organiza, y realiza su auto-producción” (Morin, 2004, 122-124).

Respecto de la sabiduría la caracteriza como reflexiva, al conocimiento lo caracteriza como organizador y a la información como designable con unidades binarias (*bits*). Abunda: “El conocimiento supone una relación de apertura y de clausura entre el conocedor y lo conocido. El

---

<sup>26</sup>Su método para esta caracterización consiste en “hacer un recorte de los datos concernientes a la actividad humana” que es “socialmente considerada como teniendo carácter cognoscitivo” (García, 2000, 39).

<sup>27</sup>Desde una perspectiva que Sotolongo identifica como “pensamiento complejo” para distinguirlo de las “ciencias de la complejidad” (Sotolongo, 2012).

<sup>28</sup>Lineal: Tal cosa produce tales efectos; circular retroactiva: que estimula o disminuye los efectos; recursiva: los efectos son necesarios para la causa.



problema del conocimiento, así como el de la organización viviente, es el de ser, a la vez, abierto y cerrado. Es el problema del cómputo-auto-exe-referente. Es el problema de la frontera que aísla a la célula y que, al mismo tiempo, la hace comunicarse con el exterior. El problema es el de concebir la apertura que condiciona a la clausura y viceversa. [...] Es sorprendente que el conocimiento emerge de un iceberg de desconocimiento prodigioso en nuestra relación con nosotros mismos. Lo desconocido no es solamente el mundo exterior, es, sobre todo, nosotros mismos. Así es que, vemos como el conocimiento supone la separación entre el conocedor y lo conocido, y supone la separación interna con nosotros mismos”. Y concluye: “Conocer es producir una traducción de las realidades del mundo exterior [...] somos coproductores del objeto que conocemos; cooperamos con el mundo exterior y es esa coproducción la que nos da la objetividad del objeto. Somos productores de la objetividad [...]. La objetividad concierne igualmente a la subjetividad. Creo que podemos hacer una teoría objetiva del sujeto a partir de la auto-organización propia del ser celular y esa teoría objetiva del sujeto nos permite concebir los diferentes desarrollos de la subjetividad hasta el hombre sujeto-consciente. Pero esa teoría objetiva no anula el carácter subjetivo del sujeto” (Morin, 2004, 152-154). En un texto más reciente, Morin afirma: “Todo conocimiento incluido el científico debe comportar en sí una reflexión epistemológica sobre sus fundamentos, sus principios y sus límites. [...] el hecho de que usted aísla y fragmente el objeto hace que la complejidad ya no exista: por lo que no es un problema científico desde el punto de vista de una disciplina cerrada y de un objeto descontextualizado. Pero desde que comienza a religar estos objetos aislados, usted se encuentra ante el problema de la complejidad” (Morin, 2010, 161).

Respecto de la acción, plantea el *principio de ecología de la acción* de la siguiente forma: “desde el momento en que una acción entra en un medio dado, escapa a la voluntad y a la intención de quién la ha creado, entra en un juego de interacciones y retroacciones múltiples y va a verse por tanto derivada fuera de sus finalidades, y en ocasiones incluso ir en el sentido contrario. La ecología de la acción tiene valor universal” (Morin, 2010, 163)<sup>29</sup>.

Quintanilla al plantear los fundamentos ontológicos de la técnica<sup>30</sup>, propone caracterizar un sistema como abierto si interactúa con su entorno; de lo contrario, es cerrado. Que el entorno de

---

<sup>29</sup>Sánchez, la editora del texto, en nota al pie, aclara que este principio es propuesto por Morin en *El Método II*, Parte Primera, Capítulo V, “El pensamiento ecologizado”: Toda iniciativa humana, toda acción “se introduce de manera aleatoria en un juego extraordinariamente múltiple y complejo de interretroacciones de los que el actor a menudo no tiene ni la menor sospecha” (pág. 105). Nos dice además que Morin extrae dos principios, el primero (y cita Morin a Lise Laférière): “El nivel óptimo de eficacia de una acción se sitúa al comienzo de su desarrollo”; el segundo es un principio de incertidumbre: “las últimas consecuencias de un acto dado no son predecibles” (pág. 106). Finalmente, afirma que este principio lo aplicará posteriormente Morin en toda su obra, y refiere como ejemplo *El Método VI* (Morin, 2010, 172).

<sup>30</sup>Ontología en la que analiza conceptos sobre: sistemas, estados y acontecimientos, regularidades, propiedades y clases de acontecimientos, sistemas complejos y acciones entre sistemas, propiedades y tipos de acciones, acción intencional, cooperación y artefactos. Para constituir el marco conceptual que le permite definir la noción de sistema técnico.

un sistema está formado por otros sistemas sobre los que actúa o que actúan sobre él; que un sistema complejo se forma cuando la interacción entre dos sistemas es estable y su regularidad más característica es el *acoplamiento*<sup>31</sup>. También propone utilizar los siguientes conceptos para caracterizar diversos aspectos de la *acción* de un sistema sobre otro: a) Sistema *agente*: el que realiza la acción; b) Sistema *paciente*: sobre el que se realiza la acción; c) *Causa*: un acontecimiento o cambio de estado en el agente que produce otro acontecimiento o cambio de estado en el paciente; d) *Efecto*: el acontecimiento o cambio de estado producido en el paciente por la causa del agente; e) *Condiciones iniciales* de la acción: el estado inicial en que se encuentra el sistema compuesto por el agente y el paciente en el momento de iniciarse la acción; f) *Resultado de la acción*: el estado final en el que se encuentra el sistema compuesto por el agente y el paciente una vez realizada la acción; y g) *Resultado neto*: el estado final del paciente de la acción.

Para definir la noción de *acción* de un sistema sobre otro, nos dice que el sistema *agente actúa* sobre el sistema *paciente* si se dan las siguientes condiciones: i) El sistema agente produce un acontecimiento al que llamamos *causa*; ii) En el sistema paciente se produce un acontecimiento al que llamamos *efecto*; iii) Existen leyes o *regularidades* que permiten afirmar que, en el estado inicial en que se encontraba el sistema paciente, sin presencia de la causa, no se hubiera producido el efecto (Quintanilla, 2005, 73).

Distingue acciones simples y compuestas. La primera es aquélla en la que intervienen solamente un sistema agente y un sistema paciente; de lo contrario es una acción compuesta<sup>32</sup>. Define un *sistema de acciones* como un conjunto de acciones compuestas conectadas entre sí, de tal forma que cualquier acción es causa o efecto de alguna otra dentro del conjunto. Más adelante, afirma que “En filosofía de la técnica nos interesan especialmente las acciones intencionales. Para poder decir que un individuo o sistema actúa intencionalmente es preciso suponer que es capaz de representarse conceptualmente la realidad y de tener creencias acerca de ésta<sup>33</sup>, que es capaz de asignar valores a determinados estados, acontecimientos o cosas y de desear que se realicen los

---

<sup>31</sup>“Decimos que dos sistemas  $S$ ,  $S'$  están acoplados si existe al menos una regularidad estructural que pone en correspondencia los valores de al menos una de las propiedades de  $S$  con los de al menos una de las propiedades de  $S'$ . El acoplamiento así definido no es simétrico. Si hay además un acoplamiento inverso de  $S'$  a  $S$  entonces decimos que los dos sistemas están en realimentación o *feedback*. Desde una perspectiva dinámica, la relación entre dos subsistemas de un sistema complejo es la acción de uno sobre otro (la interacción si la acción es recíproca), que podemos denominar también *acoplamiento dinámico*. Un sistema abierto constituye (parte del) entorno de otro durante un intervalo temporal si está dinámicamente acoplado a él. Y a la inversa, un sistema abierto está dinámicamente acoplado a su entorno si influye sobre éste. Si un sistema no influye sobre otro ni es influido por otro, en ningún intervalo de su duración, entonces es un sistema cerrado” (Quintanilla, 2005, 72-73).

<sup>32</sup>Clasifica por tipos las acciones compuestas: *Suma de acciones* cuando la causa de la acción compuesta es un acontecimiento en un solo sistema y el efecto se produce en varios sistemas. *Producto de acciones* cuando varias causas de diferentes sistemas agentes producen un efecto en un único sistema. *Producto relativo de acciones* cuando el efecto producido en el sistema paciente de una acción es la causa de otra acción de éste sobre un tercero. *Sistema de acciones* un conjunto de acciones compuestas conectadas entre sí, de tal forma que cualquier acción es causa o efecto de alguna otra dentro del conjunto (véase Quintanilla, 2005, 74).

<sup>33</sup>Recordemos aquí la definición filosófica tradicional de conocimiento: Creencia verdadera justificada.

que considera más valiosos, y que es capaz de tomar decisiones espontáneamente basándose en sus representaciones, valoraciones y deseos. Aunque el significado de estas ‘capacidades’ no es en absoluto nítido y hay continuos debates filosóficos en torno de estos conceptos, aquí supondremos por el momento que todas ellas son características precisamente de los individuos humanos, y las entenderemos en su sentido más común, dejando abierta la cuestión de si es posible atribuir las mismas capacidades a otro tipo de entidades.” Véase<sup>34</sup> a [Quintanilla \(2005, 73-74; 78\)](#). Respecto a la cooperación, el mismo autor nos dice: “Dado el producto de dos acciones de los agentes  $X$  y  $Y$  sobre el paciente  $S$ , con objetivos  $O$  y  $O'$ , decimos que se trata de un *producto intencional* de acciones si se cumplen las siguientes condiciones: i) Los dos agentes tienen una representación de la acción compuesta, de sus componentes y de su resultado, y creen que esa acción compuesta es posible; ii) Los dos agentes tienen un objetivo compartido<sup>35</sup>:  $C \subseteq O \cap O'$  que está incluido en los objetivos de sus respectivas acciones componentes, y iii) Los dos agentes creen que  $C$  está incluido en el resultado de la acción compuesta. En tal caso, decimos que el producto de las dos acciones es una *cooperación* de  $X$  y  $Y$  sobre el sistema  $S$  con el objetivo común  $C$ . La cooperación intencional requiere, por lo tanto, de una representación por parte de cada agente del conjunto de las acciones involucradas” ([Quintanilla, 2005, 80-81](#))<sup>36</sup>.

Para proponer tres modelos de gestión del conocimiento en la empresa, Quintanilla analiza dos acepciones de la noción de conocimiento técnico, distinguiendo entre conocimiento técnico en sentido primario y en sentido secundario. Para su análisis clasifica al conocimiento técnico desde dos puntos de vista o criterios: el primero, *contenido del conocimiento*, subdivide en dos el contenido y lo caracteriza como representacional u operacional; el segundo, *forma del conocimiento*, subdivide en dos la forma y lo caracteriza como explícito o tácito<sup>37</sup>. Así, al cruzar el contenido del conocimiento —representacional u operacional— con la forma del conocimiento —explíci-

---

<sup>34</sup>También es interesante y pertinente contrastar aquí la propuesta de Olivé sobre la representación y el conceptualismo naturalista ([Olivé, 2008, 205-210](#)).

<sup>35</sup> $C$  como el subconjunto de la intersección de  $O$  y  $O'$ .

<sup>36</sup>Plantea que lo contrario de la cooperación es la *confrontación*. “Dos sistemas de acciones intencionales están mutuamente confrontados si sus respectivos objetivos son incompatibles, y por lo tanto  $O \cap O' = \emptyset$  (la intersección de  $O$  y  $O'$  es vacía). Dada la definición de producto intencional se sigue que dos acciones confrontadas no pueden constituir un producto intencional y, por lo tanto, tampoco un sistema intencional, aunque sí un sistema no intencional de sistemas intencionales. Una competición o una guerra son sistemas de acciones de este tipo: todas las acciones son intencionales, pero no constituyen un único sistema intencional, sino al menos dos: uno para cada rival. A cada uno de los sistemas intencionales de acciones intencionales que componen un juego de competición los llamamos estrategias. La guerra, que es el paradigma de la confrontación, nos suministra la terminología. Una *estrategia* o sistema intencional de acciones intencionales recibe también el nombre de *plan de acción*. La ejecución de un plan de acción equivale a la ejecución intencional de un sistema de acciones intencionales y requiere, por lo tanto, un agente o conjunto de agentes que adopten el plan como esquema o representación anticipada de su propio sistema de acciones” ([Quintanilla, 2005, 80-81](#)).

<sup>37</sup>Comenta que Polanyi hace esta distinción, aunque no cita y no refiere su trabajo ([Quintanilla, 2005, 180](#)).

to o tácito— obtiene cuatro tipos de conocimiento<sup>38</sup> que denomina: *Know that*<sup>39</sup>, *Know how*<sup>40</sup>, *Intuición*<sup>41</sup> y *Habilidad*<sup>42</sup>. Concluye que “la distinción entre conocimiento técnico primario y conocimiento técnico secundario es importante para la gestión de la innovación en la empresa”. Para abundar en este tema, puede consultarse a [Quintanilla \(2005, 239-243\)](#).

Respecto de la acción racional y la eficiencia —en la que subsume a la eficacia o efectividad— plantea que hay varios modelos y que el económico es el más común: “actuar racionalmente equivale a utilizar los medios más adecuados para el fin propuesto”. Sin embargo, agrega: “El problema con esta definición es que resulta vacía mientras no precisemos el significado de *adecuación de medios a fines*” ([Quintanilla, 2005, 217](#)). Así, propone: “1. La noción intuitiva de adecuación de medios a fines de la acción como criterio de racionalidad práctica se puede ilustrar con modelos ejemplares de distintos ámbitos, especialmente en el ámbito de la economía y la tecnología; 2. El criterio de eficiencia o rendimiento económico y el criterio de eficiencia técnica no son equivalentes; 3. El criterio de eficiencia económica da lugar a consecuencias indeseables si lo adoptamos como paradigma de racionalidad práctica; 4. El criterio de eficiencia técnica es preferible como

---

<sup>38</sup>Refiere a (Ryle, 1949) como quien utiliza inicialmente las expresiones *Know that* y *Know how* para identificar dos tipos de saber o de conocimiento (citado en [Quintanilla 2005, 53](#)). También refiere que esta distinción entre las nociones de “saber cómo se hace algo” y “saber hacer algo” —que generalmente se designan con la misma expresión: *know how*— se basa en los trabajos de Piaget sobre la *la prise de conscience* y en la distinción entre conocimiento tácito y conocimiento explícito de Polanyi ([Quintanilla, 2005, 180](#)).

<sup>39</sup>Cruce entre lo representacional como contenido y lo explícito como forma. Aquí ubica el conocimiento científico, que consiste en la representación y explicación de las propiedades y regularidades características de entidades y procesos. También, a la mayor parte del conocimiento científico, tanto básico como aplicado. Las teorías tecnológicas también las ubica aquí ([Quintanilla, 2005, 240-243](#)).

<sup>40</sup>Cruce entre lo operacional como contenido y lo explícito como forma. Se refiere a propiedades y reglas características de acciones u operaciones de transformación de cosas o procesos. Aquí ubica al conocimiento Técnico Secundario: “Es el conocimiento que un tecnólogo tiene de las propiedades y reglas de operación de un sistema técnico. Incluye conocimientos tanto representacionales como operacionales, en su mayoría explícitos y formalizados (en las teorías tecnológicas, en los planos y diseños de los sistemas técnicos, etc.) que se refieren a la estructura, funcionamiento y producción de sistemas técnicos. En relación con el conocimiento técnico primario, el conocimiento secundario se puede entender como una forma de *metaconocimiento*: incluye normas, reglas y criterios acerca del tipo de conocimientos técnicos primarios que los usuarios, operadores o constructores de un sistema técnico deben tener, pero no se identifica con ellos”. Hay importantes partes del conocimiento científico que se ubican aquí: técnicas de laboratorio. El conocimiento técnico se ubica aquí con los manuales de instrucciones de uso y mantenimiento de un sistema técnico. Las teorías tecnológicas también se ubican aquí ([Quintanilla, 2005, 240-243](#)).

<sup>41</sup>Cruce entre lo representacional como contenido y lo tácito como forma. Aquí ubica el conocimiento artístico. Hay importantes partes del conocimiento científico que comparte con el arte, que se ubican aquí ([Quintanilla, 2005, 240-243](#)).

<sup>42</sup>Cruce entre lo operacional como contenido y lo tácito como forma. Hay importantes partes del conocimiento científico que se ubican aquí: técnicas de laboratorio. Aquí ubica al conocimiento Técnico Primario: “es el conocimiento que poseen los operadores o usuarios de un sistema técnico, y que es necesario para que se pueda utilizar ese sistema en forma adecuada y eficiente. Este conocimiento técnico primario está formado en gran parte por habilidades, es decir, tiene un carácter operacional y tácito, pero no exclusivamente. Buena parte del conocimiento técnico primario es de naturaleza explícita: es el que se recoge y se formula en los manuales de operación y mantenimiento de los sistemas técnicos, que especifican las reglas de funcionamiento, las operaciones que se pueden realizar, en qué orden, etc. Por otra parte, cualquier conocimiento práctico u operacional, tácito o explícito, tiene componentes representacionales ineludibles (identificación del sistema, sus partes, propiedades, etcétera)” ([Quintanilla, 2005, 240-243](#)).

criterio de acción racional instrumental” (Quintanilla, 2005, 218-219). En ese contexto, establece que: “Una acción técnica es una acción que se propone cambiar algo de la realidad de forma eficiente para obtener un resultado que se considera valioso”; mientras que por “de forma eficiente” propone entender “que la acción de que estamos hablando es efectiva (consigue en grado elevado lo que se propone) y ajustada (se minimizan los resultados no propuestos o no deseados)” (Quintanilla, 2005, 221)<sup>43</sup>.

Más adelante refiere la obra y propuestas de Kotarbinski (1965) y Bunge (1979, 1989, 1999) como su punto de partida para precisar el concepto de eficiencia técnica<sup>44</sup> “como una función de la efectividad y la adecuación” y para distinguir las propiedades de la acción racional: a) Efectividad, b) Adecuación o ajuste y c) Productividad praxiológica o técnica (Quintanilla, 2005, 227, 228, 229). Define entonces el grado de *eficiencia técnica*, de una acción  $A$  —que realiza el agente  $X$  sobre el sistema  $Y$ — en función de los conjuntos de objetivos  $O$  y los resultados  $R$ , así:

$$efic(A_{XY}) = |O \cap R| / |O \cup R| \quad (4.1)$$

de la que deriva cinco consecuencias.

La primera, el grado de *efectividad* o eficacia de la acción  $A$  del agente  $X$  sobre el sistema  $Y$  mide la *proporción de los objetivos conseguidos respecto a los pretendidos*. Esta función permite asignar un valor cuantitativo a la función de efectividad<sup>45</sup> y la define como:

$$efec(A_{XY}) = |O \cap R| / |O| \quad (4.2)$$

La segunda, el grado de *frustración* de la acción  $A$ , ya que mide *la proporción de los objetivos no*

<sup>43</sup>Se pregunta si tiene algún sentido reclamar esta noción de eficiencia técnica, como modelo o paradigma de la acción racional y para responder propone considerar que: “Desde el punto de vista de una determinada concepción absolutista o dogmática del conocimiento, la operación de proponer el conocimiento científico como paradigma de la racionalidad epistémica en general puede parecer una locura. ¿Qué sucede con la supuesta necesidad de conocimientos absolutos sobre los cuales asentar todo el edificio del saber humano? ¿Es realmente plausible reducir la racionalidad epistémica a los estrechos márgenes del método científico que sólo ofrece conocimientos provisionales, tentativos, parciales, incompletos? No argumentaré nada a este propósito. Si alguien considera que este giro es inaceptable en la esfera de la racionalidad epistémica, considerará también inaceptable mi propuesta para la racionalidad práctica. Pero lo que me interesa sobre todo argumentar es que, si no tenemos objeciones serias a la estrategia racionalista en aquel caso, entonces tampoco deberíamos tenerlas en la presente situación. En otras palabras: si no nos parece inaceptable considerar la ciencia como paradigma de la racionalidad del conocimiento tampoco debería parecer nos inaceptable considerar la tecnología como paradigma de la racionalidad de la acción” (Quintanilla, 2005, 221-222).

<sup>44</sup>En el que subsume el concepto de *efectividad*: “La efectividad de una acción consiste en su capacidad para conseguir su objetivo (puede haber también acciones contraefectivas: las que alejan del objetivo propuesto)”, y afirma —con Kotarbinski y Bunge— que “una acción es *efectiva* si sus objetivos intencionales están incluidos en los resultados efectivamente conseguidos” y advierte: “*la efectividad no es lo mismo que la eficiencia*. Cierta nivel de efectividad parece necesario para alcanzar un nivel mínimo de eficiencia. Sin embargo, una efectividad completa o máxima no garantiza un alto nivel de eficiencia” (Quintanilla, 2005, 225-226).

<sup>45</sup>La considera como una aportación a la definición de Kotarbinski (1965) y Bunge (1989). Refiere que ha sido aceptada por Bunge (2002).

conseguidos respecto de los pretendidos, y la define como:

$$frus(A_{XY}) = |O \setminus R|/|O| = 1 - efec(A_{XY}) \quad (4.3)$$

La tercera, el grado de *ajuste* o *adecuación* de la acción  $A$ , ya que mide *la proporción de los objetivos conseguidos al total de resultados producidos* (idealmente el ajuste sería máximo si se consiguieran todos y sólo los objetivos propuestos, sin desgaste ni residuo ni resultado no querido alguno) y la define como:

$$ajus(A_{XY}) = |O \cap R|/|R| \quad (4.4)$$

La cuarta, el *coste relativo* de la acción  $A$ , ya que mide *la proporción de resultados indeseados respecto al total de los producidos* y la define como:

$$cost(A_{XY}) = |R \setminus O|/|R| = 1 - ajus(A_{XY}) \quad (4.5)$$

La quinta, la *productividad praxiológica* de la acción  $A$ , ya que mide *la proporción de objetivos conseguidos respecto del total de resultados no queridos*, así definida la productividad depende completamente del ajuste o del coste relativo de la acción<sup>46</sup> y la define<sup>47</sup> como:

$$prod(A_{XY}) = |O \cap R|/|R \setminus O| \quad (4.6)$$

Cierra Quintanilla con la afirmación de que “la reivindicación de un concepto objetivo y puramente técnico de eficiencia tiene un interés tanto teórico como práctico. Desde el punto de vista teórico, nos permitirá comprender mejor algunos aspectos del desarrollo de las técnicas y de la estructura de la acción racional. Desde el punto de vista práctico, nos puede ayudar a diseñar políticas de desarrollo tecnológico orientadas a conseguir mayor eficiencia técnica, y no sólo más rentabilidad económica, y a conducir los debates públicos en torno a las opciones de desarrollo tecnológico en términos racionales” (Quintanilla, 2005, 230, 231).

Con ello, propone que la *racionalidad instrumental* de la acción se puede formular así: “La acción  $A(O,R)$  es una acción instrumentalmente racional si es un medio adecuado para conseguir  $O$ ”. Asimismo, precisa entonces el significado de “adecuación” de medios a fines al proponer que “*el grado de racionalidad instrumental de una acción es lo mismo que su grado de eficiencia técnica*” (Quintanilla, 2005, 231).

<sup>46</sup>Afirma Quintanilla que capta el contenido intuitivo de la noción de economía de la acción de Kotarbinski.

<sup>47</sup>Describe aquí una aparente paradoja de esta fórmula para acciones máximamente efectivas y ajustadas, en las que  $O = R$ , la productividad es infinita puesto que  $R \setminus O = \emptyset$ . Esto, dice, significa que debemos considerar  $O = R$  como un caso límite no realizable, ya que, entre otras cosas, violaría las leyes de la termodinámica.



### 4.3. Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución

El análisis<sup>48</sup> e interpretación integradores de estos planteamientos<sup>49</sup> me permiten concebir<sup>50</sup> al conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización:

- Si un agente  $X$  en un dominio  $D$  al ejecutar el sistema de acciones  $A$  consigue en grado elevado el objetivo  $O$  (con  $O \subseteq \omega$ ) que se propone y lo hace en tiempo  $T$  (con  $T \leq \tau$ ) y utiliza recursos  $P$  (con  $P \subseteq \rho$ ) y minimiza los resultados no deseados o no propuestos  $E$  (con  $E \subseteq \epsilon$ ), **entonces**  $X$  actúa eficaz y eficientemente ( $AEE$ ) para lograr  $O$  en  $D$ .
- Si un agente  $X$  actúa eficaz y eficientemente ( $AEE$ ) en  $D$ , un número de veces  $\eta$  (con  $\eta > 1$ ), **entonces**  $X$  actúa eficaz, eficiente y reiteradamente ( $AEER$ ) para lograr  $O$  en  $D$ .
- Si un agente  $X$  actúa eficaz, eficiente y reiteradamente ( $AEER$ ) para lograr  $O$  en  $D$ , **entonces**  $X$  **conoce** como lograr  $O$  en  $D$ .

En este sistema de acciones  $A$ , los valores de:  $\omega$  como el conjunto de objetivos de la organización,  $\tau$  como el tiempo máximo en que se puede lograr el objetivo,  $\rho$  como los recursos máximos que se pueden utilizar para lograr el objetivo,  $\epsilon$  como los resultados no deseados o no propuestos pero tolerables, y  $\eta$  como el número de veces mínimo que se debe ejecutar la acción; se definen al articular y ejecutar la **acción eficaz, eficiente y reiterada**, de los diversos agentes involucrados en la organización.

- Si una organización —considerada como un agente  $X$ — al ejecutar el sistema de acciones  $A$ , actúa eficaz, eficiente y reiteradamente ( $AEER$ ) para lograr el objetivo  $O$  en  $D$ , **entonces** esa organización **conoce** como lograr  $O$  en  $D$ .
- Si un sistema de acciones  $A$  en ejecución, cumple dos condiciones  $C_{o1}$ ,  $C_{o2}$  :

<sup>48</sup>Veáse [Herrera, A. y Torres J. A. 2007](#), 11-23, y [Morado 2009](#), 53-68.

<sup>49</sup>Esta nueva concepción utiliza los componentes esenciales de la acción  $A$  considerada como un *sistema de acciones*. Se construye sobre la propuesta de Quintanilla, quien a su vez construye a partir de Kotarbinski y Bunge: Es efectiva (consigue en grado elevado lo que se propone) y ajustada (se minimizan los resultados no propuestos o no deseados).

<sup>50</sup>De esta concepción se deriva un consecuente modelo de administración del conocimiento y éste se plantea en la siguiente sección.

- $C_{o1}$  - Las funciones de los elementos (las acciones de los diversos agentes —como subsistemas— involucrados en la organización) no son independientes, lo que determina la interdefinibilidad de los componentes
- $C_{o2}$  - El sistema como totalidad es abierto, es decir, carece de fronteras rígidas; esta inmerso en una realidad más amplia con la cual interactúa por medio de flujos de materia, energía, recursos económicos, políticas regionales, nacionales, etcétera

**entonces** el sistema de acciones  $A$  en ejecución cumple las condiciones que caracterizan a un sistema como complejo<sup>51</sup>.

- **Si** un sistema de acciones  $A$  en ejecución, incluye tres causalidades  $C_{a1}$ ,  $C_{a2}$  y  $C_{a3}$  :
  - $C_{a1}$  - Lineal: Tal acción produce tales efectos
  - $C_{a2}$  - Circular retroactiva: que estimula o disminuye los efectos
  - $C_{a3}$  - Recursiva: los efectos son necesarios para la causa

**entonces** el sistema de acciones  $A$  en ejecución, tiene todos los niveles de organización de los sistemas complejos<sup>52</sup>.

- **Si** una organización al ejecutar el sistema de acciones  $A$ , actúa eficaz, eficiente y reiteradamente (*AEER*) y el sistema de acciones  $A$ , cumple las condiciones  $C_{o1}$  y  $C_{o2}$  e incluye las causalidades  $C_{a1}$ ,  $C_{a2}$  y  $C_{a3}$  para lograr el objetivo  $O$  en  $D$ , **entonces conoce** como lograr  $O$  en  $D$  y se considera ese **conocimiento organizacional** como un **sistema complejo**<sup>53</sup>.

Por lo anterior, concibo al conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización. El sistema complejo emerge cuando el sistema de acciones en ejecución cumple las condiciones e incluye las causalidades descritas. La emergencia de este sistema complejo se observa, por ejemplo, cuando la organización logra su objetivo de mantener su acoplamiento estructural

<sup>51</sup>En el dominio de los sistemas complejos, son las dos condiciones que García utiliza para caracterizar a un sistema como complejo (García, 2008, 143-144).

<sup>52</sup>Son las tres causalidades que Morin identifica y que “se reencuentran en todos los niveles de organización complejos” (Morin, 2004, 122-124).

<sup>53</sup>Un caso de sistema complejo que emerge al articularse y ejecutarse de forma interdefinida la *AEER* de cada uno de los agentes involucrados en la organización, como un sistema abierto de acciones  $A$ , y lograr el objetivo  $O$  de mantener su acoplamiento estructural de tercer orden, se observa cuando la organización logra en grado elevado prestar sus servicios o proveer los bienes que son una parte esencial de su razón de ser. Plantea Morin: “La complejidad aparece en ese enunciado: se producen cosas y se auto-produce al mismo tiempo; el productor mismo es su propio producto.” E identifica tres causalidades que “se reencuentran en todos los niveles de organización complejos.” y concluye: “La empresa, organismo viviente, se auto-organiza, y realiza su auto-producción” (Morin, 2004, 122-124).



de tercer orden<sup>54</sup> y logra en grado elevado prestar sus servicios o proveer los bienes que son una parte esencial de su razón de ser vía la ejecución del sistema<sup>55</sup> de acciones, de forma abierta e interdefinida.

#### **4.4. Modelo para la administración del conocimiento en las pequeñas organizaciones que desarrollan software cuando pretenden adecuar software de código abierto (MACOSC)**

En el análisis de primer nivel —en el capítulo anterior— he descrito la concepción vigente de conocimiento en el campo de la administración mediante la exposición de su caracterización como conocimiento tácito y explícito, basado en la metodología de la teoría de creación del conocimiento organizacional, frecuentemente adoptada en las organizaciones que desarrollan software. También he planteado que, sin embargo, cuando el desarrollo requiere adecuar software de código abierto la dualidad de esta concepción no aplica por tres razones. La primera porque usualmente no se cuenta con el conocimiento explícito, en la forma de documentación del software, que propicie la adecuación del código en el que fue escrito. La segunda porque no se cuenta también usualmente —de manera vinculante y directa— con los programadores que tienen el conocimiento implícito correspondiente para adecuarlo. La tercer razón tiene que ver con aspectos que surgen recurrentemente en la práctica administrativa del conocimiento organizacional con la concepción vigente: Los creadores de esta concepción tomaron un curso que omitió la acción; en la práctica se asocia al conocimiento con aspectos intangibles y con el criterio de rendimiento económico.

Ante ello y como resultado del análisis e interpretación en un segundo nivel, de los planteamientos previos que desde diversos campos disciplinarios se han hecho sobre el conocimiento y su administración, en esta tesis propongo concebir al conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización<sup>56</sup> y de considerar que el sistema complejo emerge cuando el sistema de acciones en ejecución cumple las condiciones e incluye las causalidades ya descritas.

Considero que esta nueva concepción, en el dominio de las pequeñas organizaciones que desarrollan software, cuando el desarrollo involucra adecuar software de código abierto, me permite formular un modelo de administración del conocimiento organizacional como sistema complejo (MACOSC) que propicia atender su necesidad.

El modelo para la administración del conocimiento (MACOSC) en las pequeñas organizaciones

---

<sup>54</sup>Véase la definición de acoplamiento estructural en la página 35, y el de tercer orden en la página 35.

<sup>55</sup>Compuesto del conjunto de las acciones de los diversos agentes involucrados.

<sup>56</sup>Debemos notar que la organización misma es vista también como un agente.

Subproceso -Estadio	Acciones
<b>Uso</b>	<i>Observar, Representar, Compilar, Verificar, Ejecutar, Aplicar.</i>
<b>Adecuación</b>	<i>Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar.</i>
<b>Contribución</b>	<i>Observar, Proponer, Negociar, Proveer.</i>

Figura 4.1: Los Subprocesos-Estadios del MACOSC (Fuente: Elaboración propia).

que desarrollan software, cuando pretenden adecuar software de código abierto, se integra por tres subprocesos-estadios: **Uso**, **Adecuación** y **Contribución**. Estos consisten de acciones que se articulan de forma colectiva interdefinida y con interdependencia de funciones como se pretende ilustrar en la figura 4.1.

La emergencia del conocimiento organizacional como sistema complejo se observará en el grado de acoplamiento estructural y dinámico del sistema organizacional en su dominio: en los niveles de efectividad reiterada y en la eficiencia del sistema organizacional mismo, así como en los efectos deseados y no deseados en el dominio<sup>57</sup>.

#### 4.4.1. El MACOSC en la práctica administrativa

En la práctica administrativa el MACOSC<sup>58</sup> se implementa como un sistema de acciones con el propósito de que la organización logre en alto grado su objetivo. Este sistema se compone de tres subsistemas de acciones denominados Uso, Adecuación y Contribución —por eso se consideran como subprocesos— que al ejecutarse conducen a la organización a un estadio correspondiente. Cada subsistema, como subproceso-estadio, se describe a continuación.

El subproceso-estadio **Uso** consiste de las acciones siguientes:

- *Observar*

<sup>57</sup>Al diferenciar objetivos y resultados.

<sup>58</sup>Modelo para la administración del conocimiento en las pequeñas organizaciones que desarrollan software cuando pretenden adecuar software de código abierto.

Se obtiene del sitio original<sup>59</sup> el software en código fuente y se observa. El resultado cuantificable y observable es un inventario detallado.

- *Representar*

Se procesa el software con herramientas que permitan generar diversas representaciones. El resultado cuantificable y observable son las representaciones de las interrelaciones de los componentes en código fuente.

- *Compilar*

Se compila el software para generar los ejecutables y/o bibliotecas. El resultado cuantificable y observable es un programa compilado, ejecutable y/o un conjunto de bibliotecas, listos para ser invocados.

- *Verificar*

Se verifica que el software ejecutable y/o bibliotecas pasen las pruebas esenciales que constaten su funcionalidad “tal y como vienen”. El resultado cuantificable y observable es el número de pruebas que pasa el código al ejecutarlas.

- *Ejecutar*

Se ejecuta el software y/o bibliotecas en un entorno de “prueba” que permita observar su funcionalidad “tal y como vienen”. El resultado observable es que las pruebas previstas, en ese entorno, pasan sin errores.

- *Aplicar*

Se crea una aplicación inicial para confirmar que es posible usar el software “tal y como viene” para tomar la oportunidad o resolver el problema planteado en el objetivo particular de la organización. El primer resultado es que sea posible crear una aplicación sin la funcionalidad requerida. Los resultados subsecuentes agregan de forma iterativa mayor funcionalidad, con maestría creciente, hasta lograr la funcionalidad requerida para un objetivo parcial. Se itera en este estadio para crear aplicaciones cada vez más demandantes que permitan explorar el subestadio *Aplicar* con todo detalle, hasta lograr en alto grado el objetivo propuesto.

El subproceso-estadio **Adecuación** consiste de las acciones siguientes:

- *Observar*

---

<sup>59</sup>También se utilizan los términos: Se “baja”, se hace *download*, etcétera.

Se observa el software, en modo fuente y en modo ejecutable, para decidir cuáles elementos es necesario adecuar para que la organización pueda tomar la oportunidad o resolver el problema planteado de mejor manera. Se establecen como objetivos y se priorizan. El resultado es una lista de objetivos priorizados.

- *Representar*

Se procesa el software con herramientas que permitan generar representaciones del software acordes a los objetivos ordenados y priorizados. El resultado son representaciones ad-hoc al estadio cognoscitivo del agente involucrado.

- *Modificar*

Se modifica el software para alcanzar el objetivo siguiente de acuerdo a su prioridad. Se recorre parte del subproceso-estadio **Uso** una vez realizada la modificación. El resultado observable es el número de pruebas que el software modificado pasa<sup>60</sup>.

- *Compilar*

Se compila el software para generar los ejecutables y/o bibliotecas. El resultado cuantificable y observable es un programa compilado, ejecutable y/o un conjunto de bibliotecas, listos para ser invocados.

- *Verificar*

Se verifica que el software ejecutable y/o bibliotecas pasen las pruebas esenciales que constata la nueva funcionalidad. El resultado cuantificable y observable es el reporte de pruebas realizadas.

- *Ejecutar*

Se ejecuta el software en un entorno de “pruebas” que permita observar su nueva funcionalidad. El resultado cuantificable y observable es el reporte de pruebas realizadas.

- *Aplicar*

Se crea una aplicación inicial para confirmar que es posible usar el software ya adecuado para tomar la oportunidad o resolver el problema planteado en el objetivo particular de la organización. Se itera en este estadio para crear aplicaciones cada vez más demandantes que permitan explorar el subestadio *Aplicar* con todo detalle haciendo uso de la adecuación realizada. El resultado observable es el grado en que se logra el objetivo.

---

<sup>60</sup>En adición a que pase, por supuesto, todas las pruebas anteriores que confirman que no se introducen —en principio— nuevos errores en el código.

El subproceso-estadio **Contribución** consiste de las acciones siguientes:

- *Observar*

Se observa la red social de innovación vía los medios de comunicación e interacción que utilizan: listas de correo, sitios en internet. El resultado es la elección de un medio que se considere pertinente y una forma particular de abordar la red social de innovación.

- *Proponer*

Se propone la adecuación al software que se considere pertinente por el medio elegido. El resultado es la respuesta inicial de la red social de innovación.

- *Negociar*

Se negocia con la red social de innovación la responsabilidad sobre el mantenimiento de la adecuación propuesta al software. El resultado final es la decisión integrar la adecuación al código base o mantener de forma separada y pública dicha adecuación.

- *Proveer*

Se provee el software que compone la adecuación de forma acorde al resultado de la negociación del paso anterior. El resultado observable es que la adecuación este disponible al público en general, de acuerdo con los lineamientos de la licencia de uso del software.

En la figura 4.2 se pretende ilustrar el la implantación del MACOSC en la organización, como parte de los procesos de administración del conocimiento ilustrados por las circunferencias AC. En estos procesos se incorpora la práctica administrativa de implementar los subsistemas de acciones descritos.

En la figura 4.3 se pretende ilustrar el tránsito esperado de la organización, por estadios observables, durante la implantación del MACOSC en la organización, como parte de los procesos de administración del conocimiento.

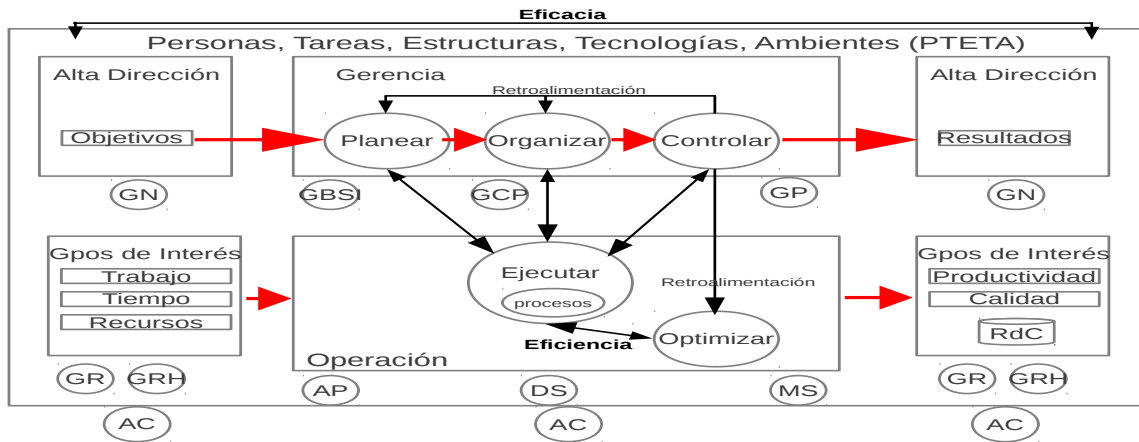


Figura 4.2: MACOSC en la Organización (Fuente: Adecuación a partir de Modelos de Procesos de la Teoría General de la Administración [Chiavenato, 2000](#), 945-946, y de los modelos de [Oktaba et al., 2009](#), 29; [Wang, 2008](#), 855-864 y [Carrillo, 2008](#), 50-53).

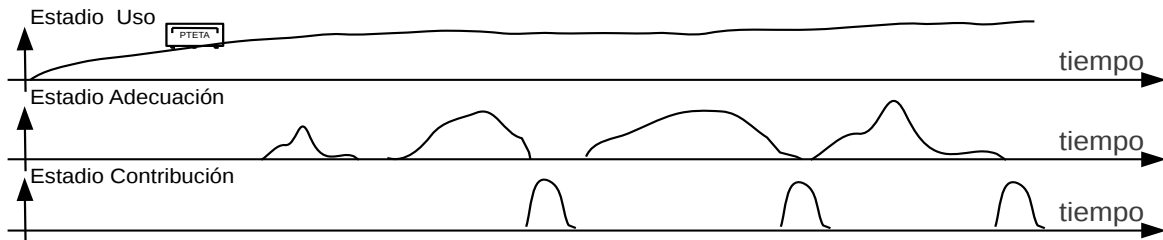


Figura 4.3: Tránsito esperado por los estadios Uso, Adecuación y Contribución al implantar el MACOSC en la Organización (Fuente: Elaboración propia).

Con esta nueva concepción<sup>61</sup> la organización como agente, el conjunto de relaciones que la caracterizan como sistema, se considera como una totalidad organizada en su acción eficaz y en su lógica de significaciones<sup>62</sup> en un momento dado. La práctica administrativa de administración del conocimiento propicia entonces la emergencia del conocimiento organizacional como sistema complejo cuando se articula y ejecuta el sistema de acciones y cuando este cumple las condiciones e incluye las causalidades descritas. El análisis de sus procesos a través del tiempo hace observable esta práctica administrativa en los niveles de efectividad reiterada y en la eficiencia alcanzados con el sistema de acciones así como en los efectos, deseados y no deseados, en el dominio<sup>63</sup>.

---

<sup>61</sup>Khun, en el capítulo de las *Revoluciones como cambios de la visión del mundo* (Khun, 2007), afirma que tras estos cambios “los científicos trabajan en un mundo distinto” (Khun, 2007, 246), y que la percepción del científico “ha de aprender una nueva Gestalt” (Khun, 2007, 213). Sin embargo, nos dice, no se debe olvidar que “sea lo que sea que ve el científico, después de una revolución sigue mirando el mismo mundo.” (Khun, 2007, 238). Por ello advierte que es posible mantener “sin remedio un diálogo de sordos” (Khun, 2007, 242), y que “incluso una vez aceptada la teoría, aún tenían que meter en cintura a la naturaleza, un proceso que acabó por llevar casi otra generación” (Khun, 2007, 246).

<sup>62</sup>Las implicaciones entre acciones.

<sup>63</sup>En la perspectiva del observador que interpreta.

# Capítulo 5

## Hipótesis

*Si* Las pequeñas organizaciones que desarrollan software, para el caso en que se propongan adecuar programas de código abierto, formulan sus acciones con el modelo MACOSC, *entonces* van a conseguir en grado elevado su objetivo.



# Capítulo 6

## Marco Metodológico

Este capítulo inicia con una perspectiva sobre las diversas alternativas de diseño de la investigación consideradas; en seguida establece los criterios utilizados para desarrollar el diseño de investigación propuesto; concluye con la descripción del diseño mixto: investigación-acción de sistemas complejos **IASC**, creado a partir del método investigación-acción y la teoría de sistemas complejos, para el estudio empírico de la pertinencia del modelo **MACOSC** propuesto para la administración del conocimiento en la pequeña organización que desarrolla software, en el caso en que se proponga adecuar programas de código abierto.

### 6.1. Perspectiva sobre diversas alternativas de diseño de la investigación

Kerlinger y Lee proponen que el diseño de investigación constituye el plan y la estructura de la investigación, y se concibe de determinada manera para obtener respuestas a las preguntas de investigación. Que el diseño, expresa tanto la estructura del problema de investigación, como el plan de investigación utilizado para obtener evidencia empírica sobre las relaciones del problema. Que los diseños tienen el propósito de permitir a los investigadores responder preguntas de la forma más válida, objetiva, precisa y económica posible. Afirman que un diseño planeado y ejecutado en forma adecuada ayuda en mucho a permitirse confiar tanto en las observaciones como en las inferencias. Concluyen que un diseño no “dice” precisamente qué hacer, sino que “sugiere” la dirección de cómo realizar las observaciones y el análisis (Kerlinger y Lee, 2002, 403-404). A partir de entender por varianza una medida de la variabilidad de los constructos (Kerlinger y Lee, 2002, 93), establecen como principio que el diseño de investigación debe controlar la varianza; que de acuerdo con este principio, al construir un diseño de investigación eficiente, el investigador intenta: 1) maximizar la varianza de la variable o variables de la hipótesis sustantiva de investigación, 2)

controlar la varianza de variables extrañas o “indeseables”, que puedan tener un efecto en los resultados experimentales, y 3) minimizar la varianza del error o aleatoria, incluyendo los llamados errores de medición (Kerlinger y Lee, 2002, 409).

Respecto de los tipos de diseño de investigación, Kerlinger y Lee hacen la distinción entre diseño experimental y no experimental. Plantean que un experimento es una investigación científica, donde un investigador manipula y controla una o más variables independientes y observa la(s) variable(s) dependiente(s) para determinar si hay variación concomitante a la manipulación de las variables independientes; que un diseño experimental es aquél en el que el investigador manipula por lo menos una variable independiente, y que en la investigación no experimental no es posible manipular las variables, asignar aleatoriamente a los participantes o tratamientos, debido a que la naturaleza de las variables es tal que imposibilita su manipulación; que ambos tipos de diseño tienen el mismo propósito básico: estudiar relaciones entre fenómenos; que su lógica científica también es la misma: obtener evidencia empírica, para realizar proposiciones condicionales de la forma *si p, entonces q*. (Kerlinger y Lee, 2002, 420). Respecto de la validez de los diseños, la dividen en dos: interna y externa. Plantean, que a mayor confianza del investigador respecto de la variable independiente manipulada, es más fuerte la validez interna, lo cual implica que tanto el experimentador puede establecer el efecto de la variable independiente sobre la variable dependiente. Afirman que hay ocho clases básicas de variables extrañas y que de no ser controladas pueden confundirse con la variable independiente y son por tanto amenazas de la validez interna. La validez externa, implica que tan fuerte es la afirmación que el experimentador puede hacer, respecto de la generalización de los resultados del estudio. Identifican cuatro amenazas a la validez externa (Kerlinger y Lee, 2002, 431-432). Kerlinger y Lee afirman también que cuando se conducen de forma correcta los estudios no experimentales son tan valiosos como los experimentales; que un ingrediente para un buen estudio no experimental es el desarrollo de hipótesis antes de iniciar el estudio, y que existen tres debilidades principales en la investigación no experimental<sup>1</sup> (Kerlinger y Lee, 2002, 516). Reconocen que los estudios no experimentales son más débiles en cuanto a la validez interna que los estudios experimentales (Kerlinger y Lee, 2002, 431); sin embargo, que a pesar de que los experimentos de laboratorio tienen una validez interna relativamente alta, carecen de validez externa (Kerlinger y Lee, 2002, 524). Asimismo, caracterizan a los experimentos de laboratorio como adecuados, principalmente para comprobar aspectos de teorías, mientras que los experimentos de campo son adecuados tanto para comprobar hipótesis derivadas de teorías, como para encontrar respuestas a problemas prácticos (Kerlinger y Lee, 2002, 527). También caracterizan a los estudios de campo como cualquier estudio científico (grande o pequeño) que busque relaciones de manera sistemática y que pruebe hipótesis, que sea no experimental y que se realice

---

<sup>1</sup>Las tres debilidades son: 1) las variables independientes no pueden ser manipuladas, 2) la carencia de aleatorización, y 3) el riesgo de la interpretación inadecuada.

en situaciones de la vida (por ejemplo, comunidades, escuelas, fábricas, organizaciones e instituciones), sin manipular variables independientes (Kerlinger y Lee, 2002, 528-529). Afirman, por otra parte que los estudios de campo son fuertes en su realismo, significancia, fortaleza de las variables, orientación teórica y calidad heurística; que la varianza de muchas variables en escenarios de campo reales es grande, especialmente comparada con la varianza de las variables de experimentos de laboratorio. Sin embargo, también establecen que en una situación de campo por lo común existe tanto ruido en el canal que aunque los efectos sean fuertes y la varianza sea grande para el experimentador no resulta fácil separar las variables. Reconocen como área dentro de los estudios de campo de la investigación cualitativa a cinco tradiciones<sup>2</sup> (Kerlinger y Lee, 2002, 530-531). Finalmente, afirman que la mayoría de las ciencias del comportamiento —especialmente la psicología— se han mostrado a favor del modelo cuantitativo (Kerlinger y Lee, 2002, 534-535).

Por otra parte, Hernández, Fernández y Baptista, proponen que el investigador debe concebir la manera práctica y concreta de responder a las preguntas de investigación y cubrir sus objetivos, que implica seleccionar o desarrollar un diseño de investigación, al cual definen como un plan o estrategia que se desarrolla para obtener la información que se requiere en una investigación (Hernández, Fernández y Baptista, 2003, 184-185). Plantean que un diseño de investigación es experimental o no experimental; el primero se asocia con un enfoque cuantitativo y el segundo puede tener un enfoque cuantitativo o cualitativo. También que el diseño experimental no es mejor que el no experimental y que ambos son necesarios y relevantes. Que el diseño de investigación depende del enfoque (cuantitativo, cualitativo o mixto) de los objetivos y preguntas planteadas, el alcance del estudio (exploratorio, descriptivo, correlacional o explicativo) y las hipótesis formuladas (si se establecieron) (Hernández, Fernández y Baptista, 2003, 187-188). Recomiendan también que una investigación se debe hacer a la medida del problema de investigación propuesto (Hernández, Fernández y Baptista, 2003, 132).

En el contexto de la hipótesis planteada, las pequeñas organizaciones que desarrollan software que se proponen adecuar programas de código abierto, consideramos relevante citar a Easterbrook, Singer, Storey y Damian quienes eligen y proponen diversos diseños empíricos a partir de consideraciones teóricas y prácticas (Easterbrook *et al.*, 2008, 286)<sup>3</sup>.

Sus consideraciones teóricas parten de la premisa que afirma que lo que es aceptable “como verdad empírica” depende de la postura filosófica, y proponen las siguientes<sup>4</sup>, como las más comunes:

- Positivismo

---

<sup>2</sup>La biografía, la fenomenología, la teoría básica, la etnografía y el estudio de caso.

<sup>3</sup>Véase la sección de “Creación de Conocimiento”, “Capítulo 11”, “Selección de Métodos Empíricos para la Investigación en Ingeniería de Software” (Easterbrook *et al.*, 2008).

<sup>4</sup>Este apretado esquema es una lista mínima que dista de pretender ofrecer aquí una descripción completa de cada postura filosófica.

Todo el conocimiento tiene que estar basado en la inferencia lógica y partir de un conjunto de hechos observables básicos. Se asocia principalmente con los diseños de experimentos controlados. Sin embargo, también se asocia con los estudios de caso y las encuestas. Es necesario creer en el reduccionismo para aceptar los experimentos de laboratorio como válidos en la ingeniería de software —estar convencido de que los fenómenos que son de interés, pueden estudiarse aislados de su contexto (Easterbrook *et al.*, 2008, 291).

- **Constructivismo**

Rechaza la idea de que el conocimiento científico puede ser separado de su contexto humano. Se concentra más en entender cómo diferentes personas dan sentido al mundo y cómo le asignan significado a las acciones, y menos en contrastar o verificar teorías. Se asocia principalmente con las etnografías, estudios de caso exploratorios y encuestas (Easterbrook *et al.*, 2008, 291).

- **Teoría Crítica**

Juzga al conocimiento científico por su habilidad de liberar a las personas de sistemas restrictivos de pensamiento. Prefiere estudios participativos que involucren a los grupos que requieren ayuda, inclusive para fijar los objetivos de la investigación. En ingeniería de software incluye investigaciones que activamente retan las percepciones existentes sobre sus prácticas. Se asocia principalmente con los estudios de caso y la investigación-acción (Easterbrook *et al.*, 2008, 291-292).

- **Pragmatismo**

Reconoce que todo el conocimiento es aproximado e incompleto, y que su valor depende del método con el que se obtiene. Juzga al conocimiento por su utilidad para resolver problemas prácticos. La verdad es lo que funciona actualmente. Se asocia con los métodos disponibles, prefiere principalmente los mixtos (Easterbrook *et al.*, 2008, 292).

Afirman que es imposible evitar una postura filosófica al llevar a cabo una investigación porque es indispensable para juzgar lo que constituye conocimiento válido.

Los diseños de investigación que proponen los podemos resumir como sigue:

- **Experimento Controlado**

Es un diseño de investigación que utiliza una hipótesis contrastable que tiene una o más variables independientes que se manipulan para medir su efecto en una o más variables dependientes. Este diseño, en su versión de experimento controlado, reduce la complejidad al permitir que sólo unas cuantas variables de interés varíen, mientras que el resto se mantienen fijas manteniendo todo ello bajo control; esto permite determinar precisamente como se

relacionan las variables y, específicamente, si existe una relación de causa-efecto entre ellas (Easterbrook *et al.*, 2008, 294).

- Estudio de Caso

Desde una perspectiva empírica es un diseño de investigación que utiliza un fenómeno contemporáneo, en su contexto real, cuando los límites entre el fenómeno y el contexto no están perfectamente definidos. Este diseño permite un entendimiento profundo de cómo y por qué ciertos fenómenos ocurren y puede revelar mecanismos por los que ocurren relaciones de causa-efecto (Easterbrook *et al.*, 2008, 296).

- Encuesta

Es un diseño de investigación que se utiliza para identificar las características de una población amplia de individuos. Se selecciona una muestra representativa de la población y por medio de cuestionarios, entrevistas estructuradas o registro de datos (*data loggin*) se captura información de la muestra. La respuesta a la pregunta de investigación así obtenida se generaliza a la población utilizando diversas técnicas (Easterbrook *et al.*, 2008, 298).

- Etnografía

Es un diseño de investigación que se utiliza para entender cómo una comunidad de personas, crea sentido de su interacción social a través de la observación de campo. Evita imponer teorías pre-existentes y en su lugar identifica las categorías creadas por la comunidad. El resultado es una descripción detallada de la cultura de la comunidad. Este diseño, en su versión de observación participante, permite realizar la descripción desde el punto de vista privilegiado de un miembro de la comunidad (Easterbrook *et al.*, 2008, 300).

- Investigación-Acción

Es un diseño de investigación que se utiliza para resolver un problema real y simultáneamente estudiar la experiencia de resolver el problema. La mayoría de los diseños empíricos pretende observar el mundo como existe; sin embargo, este diseño tiene el objetivo de intervenir la situación estudiada con el propósito de mejorarla (Easterbrook *et al.*, 2008, 301).

- Mixto

Es un diseño de investigación que emerge de reconocer que todos los métodos tienen limitaciones y que las debilidades de un método pueden ser compensadas con las fortalezas de otro. Demanda familiaridad con los enfoques cualitativo y cuantitativo, decidir sobre la secuencia del enfoque y las respectivas formas de recolección de datos y enfrentar los acrecentados requerimientos de análisis de datos. Las estrategias más comunes son las siguientes:

- Estrategia Secuencial Explicatoria

Se caracteriza por la recolección y análisis de datos cuantitativos, seguidos por la recolección y análisis de datos cualitativos. Los resultados cualitativos sirven para explicar los cuantitativos; sin embargo, es de particular utilidad, cuando los resultados de la fase cuantitativa no son lo que se esperaba (Easterbrook *et al.*, 2008, 303).

- Estrategia Secuencial Exploratoria

Se caracteriza por la recolección y análisis de datos cualitativos, seguidos por la recolección de datos cuantitativos. Los resultados cuantitativos se utilizan para asistir en la interpretación de los cualitativos. Es también útil para contrastar elementos de una teoría emergente, resultante de un estudio cualitativo (Easterbrook *et al.*, 2008, 303).

- Estrategia de Triangulación Concurrente

Es quizás la más utilizada para los diseños mixtos. Utiliza concurrentemente diferentes métodos para intentar confirmar, validar en forma cruzada o corroborar resultados. Es motivada por el hecho de que frecuentemente “lo que las personas dicen” es diferente de “lo que las personas hacen”. Puede, sin embargo, ser difícil comparar los resultados de dos tipos de análisis o resolver contradicciones (Easterbrook *et al.*, 2008, 304).

También, en el contexto del marco teórico de la hipótesis que concibe al conocimiento organizacional como un sistema complejo, consideramos relevante citar a García, principalmente el “Capítulo II”, “Marco Conceptual y Metodológico para el Estudio de Sistemas Complejos”, en el que establece que el enfoque metodológico empírico se deriva de dos componentes, una posición epistemológica y una concepción de “la realidad”, que asocia con dos preguntas ¿Cómo conocemos? y ¿Qué conocemos?:

- Posición Epistemológica y ¿Cómo conocemos?

Por *conocimiento*, entiende “un fenómeno social intersubjetivo”. Por *conocer*, entiende “establecer relaciones en una materia prima que es provista por la experiencia, y es organizada por el sujeto cognoscente”. Define *observables* como datos de la experiencia *ya interpretados*<sup>5</sup>, y *hechos* como *relaciones* entre observables (García, 2008, 43). Afirma que conocemos construyendo, de forma intersubjetiva, teorías<sup>6</sup> en un proceso dialéctico en espiral, de dife-

---

<sup>5</sup>Es decir no hay datos “puros”. Todo registro de “la realidad” que realiza el investigador corresponde a sus propios esquemas interpretativos. No hay “lectura pura” de la experiencia; toda experiencia está “cargada de teoría” (García, 2008, 43).

<sup>6</sup>Teorías como instrumentos asimiladores de la experiencia, que incluyen teorías científicas formuladas con cierto rigor, y el conjunto de afirmaciones y suposiciones, explícitas o implícitas, sobre la base de las cuales un investigador establece su hipótesis o realiza sus inferencias (García, 2008, 44). Instrumentos asimiladores, podemos agregar, que se construyen y reconstruyen en cada nivel cognoscitivo del investigador como sujeto cognoscente.

renciaciones e integraciones sucesivas<sup>7</sup> (García, 2008, 83-84).

■ Concepción de “la realidad”, ¿Qué conocemos?

Establece como características del Universo que es estratificado y no lineal para explicar la concepción de “la realidad” y responder la pregunta.

Para afirmar que el Universo es estratificado, parte de dos *hechos* actuales que caracterizan al Universo. El primero afirma que las mismas leyes y las mismas formas de organización no rigen en todos los dominios y en todas las escalas de fenómenos; que está constituido por múltiples niveles de organización semi-autónomos, regidos por respectivas dinámicas específicas de cada uno de ellos, pero que interactúan entre sí. El segundo afirma que los diferentes niveles están “desacoplados” en el sentido de que las teorías desarrolladas en cada uno de los niveles tienen suficiente estabilidad, como para no ser invalidadas por descubrimientos o desarrollos en otros niveles (García, 2008, 74-75).

Para afirmar que el Universo no es lineal, parte del *hecho* de que fenómenos que integran totalidades (sistemas), se transforman en el tiempo respondiendo a una ley muy general: evolución no-lineal, con discontinuidades estructurales, que procede por sucesivas reorganizaciones (García, 2008, 75-76).

Respecto de la pregunta, ¿Qué conocemos?, consideramos que en la sección 4, “Implicaciones Metodológicas de la Estructuración de la Realidad”, parte de esta concepción de “la realidad” para responder que conocemos sistemas. Primero define un sistema de forma general como “un recorte de los datos empíricos en totalidades relativas, suficientemente autónomas, como para servir de marco a un trabajo científico”, y de forma más precisa como la *representación* de un conjunto de situaciones, fenómenos, procesos que pueden ser modelizados como una totalidad organizada, con una forma de funcionamiento característica (García, 2008, 79). A continuación, caracteriza como “sistemas descomponibles” a aquéllos cuyas partes pueden ser aisladas y modificadas independientemente unas de otras, y como “sistemas complejos” aquéllos en que los procesos que determinan su funcionamiento son interdefinibles<sup>8</sup> y múltiples, en tanto resultan de la confluencia de diversos factores que interactúan de manera tal que no pueden ser aislados (García, 2008, 79-80). Adicionalmente, define los principios característicos que tienen los sistemas complejos: a) una disposición

---

<sup>7</sup>Porque las teorías en un momento inicial contienen “recortes de la realidad”, totalidades, definidas con cierto grado (a veces muy alto) de imprecisión y, por un proceso de análisis, se van diferenciando elementos que las integran (también con cierta imprecisión). El estudio de estos elementos permiten caracterizarlos mejor, con lo que se reconstruye, se integra, “un recorte de la realidad” o una totalidad mejor determinada. De esta forma el proceso comienza y prosigue en sucesivas etapas en cada una de las cuales se realizan ajustes que pueden consistir en incorporar factores omitidos o eliminar factores que aparecen como innecesarios o secundarios (García, 2008, 83-84).

<sup>8</sup>Por interdefinible se entiende que no es independiente, que se determina mutuamente con otro(s) elemento(s) (García, 2008, 49).



de sus elementos por niveles de organización con dinámicas propias, pero interactuantes entre sí; b) una evolución que no procede por desarrollos continuos sino por reorganizaciones sucesivas.

De esta forma, las pautas específicas que propone para el ordenamiento metodológico de la investigación (García, 2008, 79-85) las resumimos así:

- Realizar un análisis cualitativo para:
  - Identificar niveles de organización de la “realidad estructurada”.
  - Identificar la interacción entre niveles de organización.  
Notando que en el sistema complejo, cada cambio estructural de él mismo o de alguno de los niveles de organización que lo componen, está fuertemente condicionado por las interacciones denominadas condiciones de entorno.
  - Calendarizar la evolución por las reorganizaciones sucesivas de cada nivel de organización. Reconociendo “etapas macro” que sirven para ordenar el “análisis” del “estudio de caso”:
    - Actividades involucradas en el desarrollo de cada sistema semi-autónomo.
    - Grandes procesos concomitantes con el desarrollo de cada sistema semi-autónomo porque ellos configuran las condiciones de entorno de cada uno.
- Repetir el proceso sucesivamente para agregar factores omitidos, eliminar factores innecesarios para así construir conocimiento y una realidad mejor determinada.

## 6.2. Criterios utilizados para el diseño de la investigación

Para establecer el primer criterio, consideramos que es necesario clarificar lo que entendemos por el concepto mismo de la relación entre variables que Kerlinger y Lee (2002, 420) describieron así: *si p, entonces q*. Como primer paso, recurrimos a Hume, quien respecto de la idea de relaciones entre fenómenos dice: “Resulta, pues, aquí que, de las tres relaciones que no dependen de las meras ideas, la única que puede ser llevada más allá de los sentidos e informarnos de existencias y objetos que no podemos ver o tocar es la causalidad.” (Sección II, De la probabilidad y de la idea de causa y efecto) (Hume, 1998, 58). Más adelante, afirma: “Los objetos no poseen una conexión entre sí que pueda descubrirse, y por ningún otro principio más que por la costumbre, que actúa sobre la imaginación, podemos hacer una inferencia del uno para llegar a la existencia del otro” (Sección VIII, De las causas de la creencia) (Hume, 1998, 76). Finaliza sus argumentos, con una



definición precisa de la relación causa y efecto: “Una causa es un objeto precedente a otro y tan unido a él que la idea del uno determina al espíritu a formarse la idea del otro y la impresión del uno a formarse una idea más vivaz del otro. Si esta definición es rechazada, por razón idéntica no veo otro remedio sino que las personas que se manifiesten tan delicadas pongan en su lugar una definición más precisa.” (Sección XV, De la idea de la conexión necesaria) (Hume, 1998, 116). Como segundo paso recurrimos a García, quien afirma: “La concepción de la causalidad como una ‘atribución’ de necesidades lógicas (teóricas) a la experiencia (observables y hechos), constituye la respuesta que surge de la epistemología piagetiana con respecto al célebre problema que planteó Hume”. En ese sentido, afirma que: “lo que definitivamente no es aceptable es la antigua y persistente idea baconiana de la ciencia, según la cual se llega a la relaciones causales por vía inductiva, a través de una generalización de ‘regularidades’ observadas en la experiencia”; concluye al afirmar que: “El punto de vista según el cual las relaciones causales se establecen en el nivel teórico (aunque puedan ser ‘sugeridas’ por la experiencia) tienen fundamental importancia para la práctica de la investigación científica. Significa, en primer instancia, reconocer que las relaciones entre observables (o entre hechos) no surgen de la simple ‘evidencia’ empírica” (Sección 1.3 Las relaciones causales) (García, 2008, 46).

Establecemos así, como criterio fundamental de diseño de investigación, respecto de la relación entre variables:

- Por los argumentos de Hume, que no es posible encontrar “relaciones observables” entre los mismos fenómenos.
- Por los argumentos de García, que lo que pretenderemos es construir explicaciones causales, y las consideraremos científicamente válidas, porque es posible hacer con ellas una atribución lógica<sup>9</sup> y coherente a la “realidad empírica”.

Respecto de la “realidad empírica” que nos ocupa, las organizaciones que administran conocimiento, Griseri desde una posición crítica considera que los trabajos de investigación que involucran directamente a las organizaciones, en contraste de investigaciones separadas e independientes de ellas, y que involucran múltiples perspectivas e intereses, enriquecen el proceso de investigación en las ciencias de la administración al permitir generar conocimiento en las diferentes áreas de interés involucradas. También que estos elementos adicionales de conocimiento se caracterizan por ser fruto de la acción y la reflexión conjunta entre administradores de la organización e investigadores. Finalmente, considera que estos elementos tienen el potencial, al ser expresados de

---

<sup>9</sup>En la lógica, como reglas de inferencia utilizamos preferentemente el *Modus tollendo tollens*: **si p, entonces q; no q, no p.** en lugar del *Modus ponendo ponens*: **si p, entonces q; p, q.** para establecer explicaciones causales entre variables.

manera apropiada, de reflejarse en el progreso de las organizaciones y en el progreso en los procesos de investigación en las Ciencias de la Administración (Griseri, 2002, 272-273). Por otra parte, Rivas desde la teoría de la complejidad, propone que la gestión “distingue ocho elementos que configuran el funcionamiento de toda organización: aspectos jurídicos, económicos y financieros, políticos y organizativos, culturales, educativos, sociales, tecnológicos, e internacionales” (Rivas *et al.*, 2009, 14-15), y propone utilizar diez conceptos para caracterizar a la organización como un sistema adaptativo complejo (Rivas *et al.*, 2009, 39-59).

Por los argumentos de Griseri y de Rivas, proponemos como criterios adicionales de diseño de la investigación los siguientes:

- En las Ciencias de la Administración existen teorías y regularidades, sin embargo no hay leyes. Por ello, nos dice Griseri, la investigación en Ciencias de la Administración contribuye a describir, explicar que pasa en y a las organizaciones, así como encontrar y evaluar métodos y herramientas, que ayuden a los administradores a lograr los fines organizacionales más efectivamente; parten de que lo que es apropiado hoy, puede ser inapropiado en cincuenta años más, así como pudo haber sido inapropiado para los retos de hace cincuenta años (Griseri, 2002, 42-43).
- Las teorías en las Ciencias de la Administración son condicionadas culturalmente. Las variables culturales, subyacentes a los conceptos de dichas teorías, son condiciones necesarias para la aplicación de dichos conceptos (Griseri, 2002, 79-93).
- El método de investigación a elegir debe considerar que la forma principal de conocimiento en las Ciencias de la Administración es el de las soluciones locales que los administradores, con intereses imbricados, desarrollan para resolver los problemas que confrontan (Griseri, 2002, 272).
- Pretender encontrar un método de administración del conocimiento organizacional único, con validez general todo el tiempo en las organizaciones, no es racional.

Respecto de las propuestas de Maturana y Varela, consideramos pertinente:

- Adoptar la identidad entre acción y conocimiento como criterio básico de contrastación.
- La identidad entre acción y conocimiento implica que todos nuestros actos, sin excepción, contribuyen a formar el mundo en que existimos y que validamos precisamente a través de ellos en un proceso que configura nuestro devenir, y tienen un carácter ético porque tienen lugar en el dominio social (Maturana y Varela, 2003, 164). Esta implicación la adoptamos como criterio general de investigación.

- Como observadores, podemos ver una unidad en dominios *diferentes*, según sean las distinciones que realicemos. Así, por un lado, podemos considerar a un sistema en el dominio del operar de sus componentes, en el dominio de sus estados internos y sus cambios estructurales. Desde este operar, para la dinámica interna del sistema, el ambiente no existe, es irrelevante. Por otro lado, *también* podemos considerar a una unidad en sus interacciones con el medio y describir su historia de interacciones en él. Para esta perspectiva en la que el observador puede establecer relaciones entre ciertas características del medio y la conducta de la unidad, la dinámica interna de ésta es irrelevante. Ninguno de estos dos posibles dominios de descripción es problemático en sí, y ambos son necesarios para satisfacer nuestro sentido de cabal entendimiento de una unidad (Maturana y Varela, 2003, 91). Esto lo adoptamos como criterio general de observación empírica.

Finalmente, al pretender identificar *observables y hechos —explicaciones causales—* consideramos dos criterios que restringen los trabajos. El primero parte de la propuesta de Sowa, respecto de la vaguedad, incertidumbre, aleatoriedad e ignorancia que prevalecen cuando se desea determinar exactamente lo que existe:

- La gradación continua y el rango abierto de excepciones, hacen imposible dar definiciones precisas y completas de cualquier concepto aprendido a través de la experiencia (Sowa, 2000, 360).

El segundo parte de la propuesta de García respecto de la investigación interdisciplinaria:

- Lo que integra a un equipo interdisciplinario, para el estudio de un sistema complejo, es un marco conceptual<sup>10</sup> y metodológico común (García, 2008, 35).

### **6.3. El diseño de la investigación-acción de sistemas complejos (IASC)**

La postura filosófica que adopto en esta investigación para realizar los trabajos de diseño es pragmática, como lo expresa Laudan (1993). Así, realizo un diseño mixto que integra el método de investigación-acción y la teoría de sistemas complejos.

Para justificar la decisión de utilizar en el diseño la investigación-acción, en primer lugar, considero como opinión experta la de Oktaba, quien reporta utilizar ese método en sus trabajos de

---

<sup>10</sup>Son entonces un aspecto crítico las definiciones de conceptos, la ontología. Para tratar este aspecto, Rolando nos refiere (García, 2008, 74) a Willard Van Orman Quine: Parte IV de *Methods of Logic* (1950). Sugiero ver también la 4a edición de 1984 (Quine, 1984, 256-303).

investigación sobre procesos de ingeniería de software para pequeñas y medianas empresas. El sustento de esta consideración es que sus trabajos son reconocidos en el nivel internacional. Asimismo, Oktaba ha logrado que los procesos de software que ella y sus colaboradores han propuesto sean aceptados como estándares internacionales (Morales, Ventura, Oktaba y Torres, 2012), (Oktaba *et al.*, 2009). En segundo lugar considero que, desde la perspectiva de la informática, realizar investigación sobre sistemas de información con el método de investigación-acción se defiende apropiadamente en diversos artículos (Avison *et al.* 1999a; Avison *et al.* 1999b, y Baskerville 1999). También, es relevante utilizar el método investigación-acción pues, la intervención puede ser beneficiosa para la organización que participa en la investigación, lo que conlleva el potencial de fortalecer los vínculos entre las pequeñas organizaciones y los centros de investigación universitarios.

En este contexto, la justificación de la decisión de utilizar la teoría de sistemas complejos, desde la perspectiva de la epistemología genética, tiene dos aspectos. El primero es que considero como opiniones expertas la de Carrillo, quien reporta utilizar esta teoría de la complejidad en sus trabajos de investigación de gestión del conocimiento en entornos académicos (Carrillo, 2008), así como la de Amozurrutia, quien propone aplicarla en la construcción de sistemas adaptativos para el análisis social (Amozurrutia, 2011). El segundo aspecto es mi propósito de entrar en detalle y estudiar esta teoría al pretender aplicarla como columna vertebral teórica del diseño investigación-acción, para explorar su potencial como solución<sup>11</sup> a varias recomendaciones reportadas en los artículos de Madeiros dos Santos y Horta respecto de la tendencia creciente del uso del diseño investigación-acción en ingeniería de software (Madeiros dos Santos y Horta, 2009), y en el de (Kock *et al.*, 1997) respecto del rigor científico del mismo método investigación-acción.

Así, en esta investigación parto de las anteriores consideraciones para realizar el diseño de investigación, que denomino diseño mixto investigación-acción de sistemas complejos (IASC), con el objetivo de observar la pertinencia del MACOSC para la administración de conocimiento en una organización concreta que pretende adecuar programas de código abierto específicos. Este diseño, IASC, consiste en un proceso iterativo incremental que contempla la múltiple realización de subprocesos de investigación-acción que se repiten, con maestría creciente, en cada iteración. Un subproceso de investigación-acción está compuesto por cuatro actividades fundamentales: diagnóstico de la oportunidad, planeación de la acción, intervención, aprendizaje reflexivo (Avison *et al.* 1999a; Kock *et al.* 1997).

La propuesta teórica de la investigación de sistemas complejos, sintetizada por García (2008, 181-190), Amozurrutia (2011, 227 y 319-324) y por Rivas *et al.* (2009, 39-97), se usa como co-

---

<sup>11</sup> La falta de control sobre las variables (del ambiente, etc.) o rigor científico, con la consecuente: imposibilidad de generalizar los resultados, falta de imparcialidad, y riesgo de involucramiento del investigador y su efecto en los resultados.

lumna vertebral de las cuatro actividades que componen cada subproceso de la forma siguiente:

### 1. Diagnóstico de la oportunidad

- a) Actividad fundamental en donde se entrecruzan la perspectiva cualitativa y la cuantitativa. Exige aplicar ambas perspectivas; primero, para identificar variables cualitativas que permitan iterativamente construir la *explicación causal* que dé cuenta de la funcionalidad del sistema complejo —el conocimiento organizacional como acción eficaz, eficiente y reiterada—, y segundo, para identificar aspectos mensurables en el **complejo empírico** —*la realidad*<sup>12</sup>— que puedan ser atribuidos<sup>13</sup> a las variables cuantitativas en la teoría que se construye.

Así, en esta investigación propongo realizar un análisis cualitativo y cuantitativo, para identificar la oportunidad<sup>14</sup> o problema que confronta la organización, y para caracterizarla en los entornos que la albergan<sup>15</sup> desde una perspectiva de sistemas complejos.

- b) Identificar el objeto de estudio<sup>16</sup>, los objetos empíricos de estudio<sup>17</sup>, que estarán en correspondencia con la cadena de inferencias en la teoría. Crear la versión<sup>18</sup> del sistema

---

<sup>12</sup>El complejo empírico es la realidad. Esta realidad se abstrae y sistematiza en descripciones, en un conjunto observable de datos que *cazan*, que se atribuyen a esa realidad (véase [García, 2008](#), 186). La distinción que hace García entre el *complejo empírico* y el *sistema* construido —a partir de abstracciones— por el investigador, me permite recurrir a Kant quien distingue entre *fenómeno* y la *cosa en sí*: Un *fenómeno* es una cosa en cuanto objeto de la experiencia; tal y como nos es dada, se rige por nuestra forma de representación, admite determinaciones. La *cosa en sí* es una cosa que no podemos conocer; es lo incondicionado, lo indeterminado, lo que no puede conocerse sin contradicción —porque si el conocimiento en Kant es determinación, ¿cómo conocer lo indeterminado?—, a pesar de ser real por sí misma. Afirma Kant que si la crítica —a la razón pura— no se ha equivocado al tomar un objeto en estos dos sentidos [...] entonces “el principio de causalidad se aplica únicamente a las cosas en el primer sentido, es decir, en cuanto objetos de la experiencia, sin que le estén sometidas, en cambio, esas mismas cosas en el segundo sentido”. Para mayor detalle véase ([Kant, 1995](#), 22-26). Véase también ([Maass et al., 2012](#), 291).

<sup>13</sup>Atribución en el sentido que expresa García: La *explicación causal* es una *atribución* a la realidad de transformaciones que estarían en correspondencia con la cadena de inferencias (deductivas o no) dentro de la teoría explicativa. Esta atribución no implica ningún tipo de hipótesis, porque no otorga *realidad objetiva* a tales transformaciones ([García, 2008](#), 189).

<sup>14</sup>En nuestro caso, la pertinencia del MACOSC para la administración del conocimiento en la pequeña organización que desarrolla software y pretende adecuar programas de código abierto.

<sup>15</sup>En nuestro caso, la pequeña organización que desarrolla software.

<sup>16</sup>En nuestro caso la relación, representada en la teoría, entre los **objetivos**, las **acciones** y los **resultados**, para transitar por los tres subprocesos-estadios para la administración del conocimiento indicados en el MACOSC como **unidad de análisis**.

<sup>17</sup>García los denomina en su conjunto como una abstracción del **complejo empírico** ([García, 2008](#), 185, 186) que, en nuestro caso, corresponde al cúmulo real de objetivos, acciones y resultados de una organización. Dicho cúmulo se abstrae, para sistematizarlo, como el conjunto compuesto por los **objetivos** expresados por la organización, las **acciones** indicadas para los subprocesos-estadios y los **resultados** reportados a la organización. Todas las acciones, que componen los subprocesos-estadios, se asocian con objetos empíricos: Los programas de código abierto, los requerimientos detallados, la listas de objetivos y la de resultados por mencionar algunos de los objetos empíricos de nuestra **unidad de observación**.

<sup>18</sup>Le llamo versión porque debemos generar una nueva “versión” del sistema complejo —el conocimiento organizacional siempre cambiante— para cada iteración en el ciclo básico de las cuatro actividades que componen cada subproceso: diagnóstico de la oportunidad, planeación de la acción, intervención, aprendizaje reflexivo.

complejo<sup>19</sup> que representa el **complejo empírico** de estudio. Proceder con la distinción y análisis de los niveles del sistema complejo:

- 1) Identificar niveles de organización de la “realidad estructurada”: del conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización.
- 2) Identificar la interacción entre niveles de organización del sistema complejo.
- 3) Calendarizar la evolución por las reorganizaciones sucesivas de cada nivel de organización del sistema complejo.

## 2. Planeación de la acción

- a) Identificar los cursos alternativos de acción para aprovechar la oportunidad, mejorar la situación o resolver el problema.
- b) Seleccionar el curso de acción que se considere más apropiado.
- c) Establecer los instrumentos de observación y análisis del complejo empírico.

## 3. Intervención

- a) Realizar el curso de acción elegido.
- b) Observar los efectos sobre el complejo empírico.

## 4. Aprendizaje reflexivo

- a) Evaluar los resultados del curso de acción realizado, desde una perspectiva científica, sobre las observaciones realizadas en el complejo empírico:  
Agregar variables omitidas y eliminar variables innecesarias en la teoría, para así construir conocimiento<sup>20</sup>, respecto del objeto de estudio y una realidad mejor determinada.
- b) Desde una perspectiva organizacional, evaluar los resultados del curso de acción realizado:

---

<sup>19</sup>La intención es explicar, en cada iteración, de forma más precisa y atinada los procesos al interior del sistema complejo, tanto en las interrelaciones de sus partes como en su totalidad organizada. Para propiciar con maestría creciente la emergencia del conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización.

<sup>20</sup>Conocimiento en el sentido de acción eficaz, eficiente y reiterada de los agentes epistémicos, cognoscitivos que conforman el grupo de investigación.

Identificar la consolidación de invariantes, que se consideren con valor organizacional<sup>21</sup> y para los cuales sea posible establecer una *explicación causal* que da cuenta de la intervención<sup>22</sup>.

- c) Si al evaluar los resultados, se considera que puede ser conveniente realizar otro ciclo, tanto desde la perspectiva académica como organizacional, entonces hay que reiniciar un nuevo paso del proceso iterativo incremental que contemple la realización de los subprocesos: diagnóstico del problema, planeación de la acción, intervención, aprendizaje reflexivo. Por otro lado, en caso que se considere pertinente dar por terminado el proceso, hay que generar los reportes finales apropiados para la comunidad académica y para la organización.

A continuación, se muestra un esquema gráfico con mayor detalle del ciclo y fases que componen el diseño mixto **investigación-acción de sistemas complejos (IASC)**, y que se utiliza en esta investigación para intervenir en la organización sujeto de estudio. Véase la figura 6.1.

Se advierte que el diseño del método es un primer paso y que éste no se propone como único o que responde puntualmente a todos los requerimientos de científicidad actualmente reconocidos en las Ciencias de la Administración. Sin embargo, pretende ser el inicio de un círculo virtuoso de estudio-transformación que asume que no es posible considerar efectos constantes en las acciones de los individuos y de la pequeña organización sujeto de estudio *i.e.* que la misma causa no necesariamente producirá el mismo efecto y que no es posible controlar de manera constante las variables de una situación<sup>23</sup>.

En la figura 6.1 se muestran las fases del ciclo del diseño mixto creado (**IASC**) descritas en la sección 6.3: El Diseño de la investigación. Al realizar un ciclo del **IASC** se implementa un estadio del MACOSC y así se contribuye a la emergencia del conocimiento organizacional. Mientras que con el estado del sistema —rectángulo de esquinas redondeadas dibujado con línea punteada— se pretende representar la implementación de alguno de los subprocesos-estadios por los que transita la organización (**Uso, Adecuación o Contribución**), con el sistema —rectángulo externo dibujado

---

<sup>21</sup>Valor organizacional asociado a la(s) variable(s) dependiente(s). En nuestro caso esta variable corresponde a los **resultados** que se representan en la tríada: **objetivos, acciones, resultados**. Con diversos índices e instrumentos pretendemos caracterizarlas y observarlas, para ser capaces de consolidar, el avance en el logro de los objetivos organizacionales para usar, adecuar y aportar/distribuir los programas de código abierto.

<sup>22</sup>Vía la(s) variable(s) independiente(s). En nuestro caso el sistema de acciones formulado en el MACOSC. Con este modelo se pretende caracterizar, analizar y propiciar la diversidad de estadios por los que atraviesa la organización al realizar, en la práctica, la administración del conocimiento organizacional. Estas acciones deben ser conmensurables tener similar adecuación empírica y, sin embargo, diferente nivel de respuesta funcional empírica, al resistir de mejor o peor manera, el enfrentamiento con contrastaciones exigentes. Contrastaciones en las que incluimos, el grado en que propician la eficacia de los integrantes de la organización, para usar, adecuar y aportar/distribuir los programas.

<sup>23</sup>**Poteete, Janssen y Ostrom** proponen: “El supuesto tal vez más importante de este trabajo es que la comparación es valiosa, aunque rara vez existe un patrón único o lineal. Esperamos la heterogeneidad causal: hay más de una ruta para llegar al mismo resultado” (**Poteete, Janssen y Ostrom, 2012, 71**).



con línea punteada— se pretende representar la emergencia del sistema complejo (el conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización).

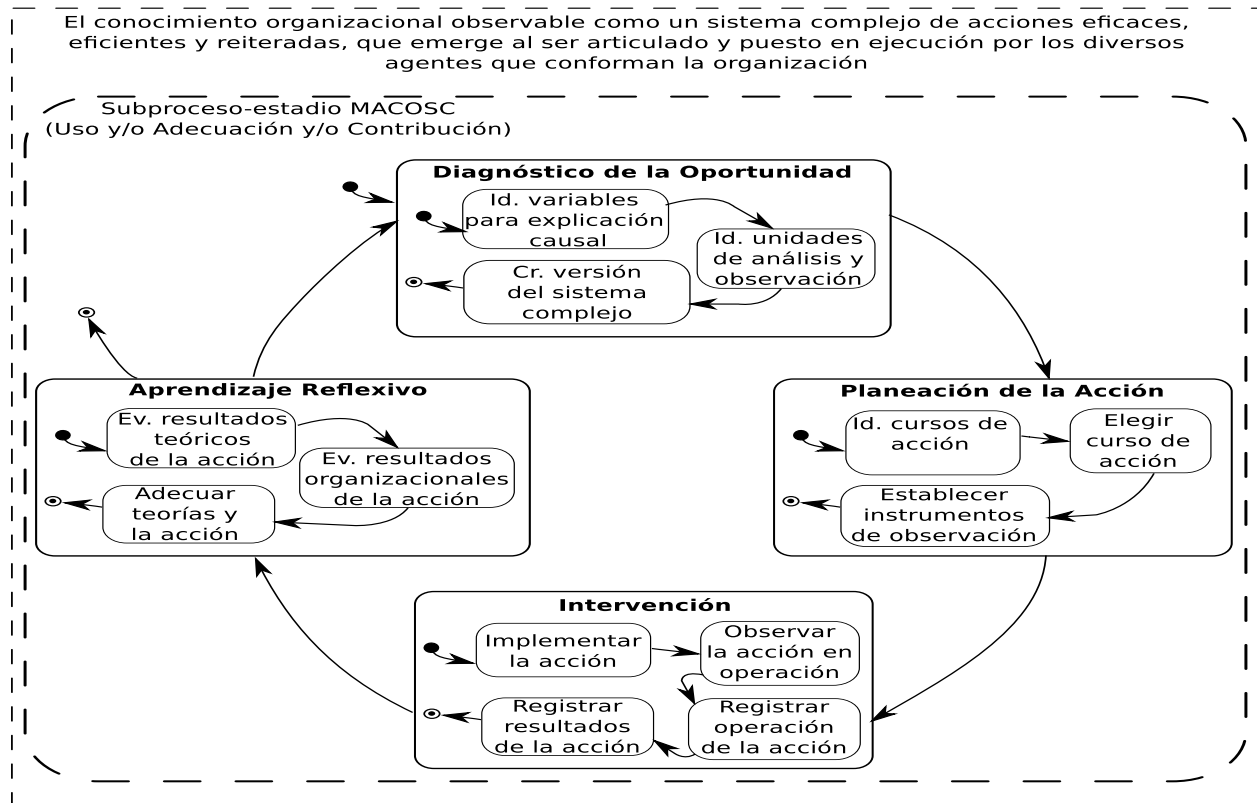


Figura 6.1: Detalle del diseño mixto IASC para intervenir la organización (Fuente: Elaboración propia.)

Una representación gráfica de la ubicación<sup>24</sup> del método que he diseñado para esta investigación, que denomino diseño mixto **investigación-acción de sistemas complejos (IASC)**, se muestra en la figura 6.2. En el detalle que se muestra del método, la espiral corresponde al proceso iterativo incremental; asimismo, un rombo marca el inicio de un subproceso compuesto por las cuatro actividades que se realizan en la investigación<sup>25</sup> al implementar el MACOSC<sup>26</sup> para observar el conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización<sup>27</sup>.

<sup>24</sup>En el espacio de diseños de investigación considerados.

<sup>25</sup>Diagnóstico de la oportunidad, planeación de la acción, intervención y aprendizaje reflexivo.

<sup>26</sup>Es decir, se cruza el MACOSC con el IASC: MACOSC-IASC.

<sup>27</sup>Dicho sistema complejo emerge cuando el sistema de acciones en ejecución cumple las condiciones e incluye las causalidades que se describen en el capítulo 4. *Marco Teórico Conceptual* en la sección 4.3. *Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución*, en la página 44.



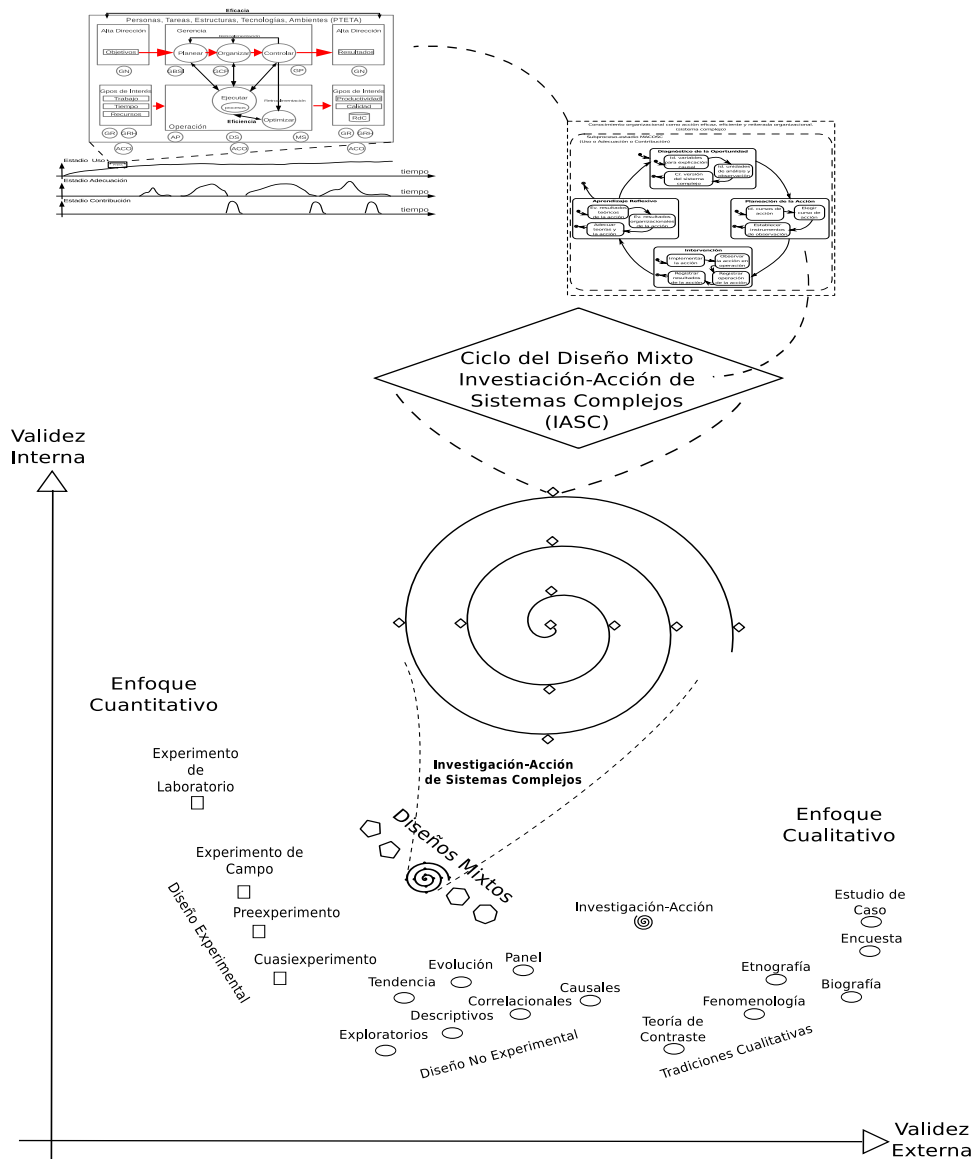


Figura 6.2: Ubicación del método Investigación-Acción de Sistemas Complejos (Fuente: Elaboración propia a partir de (Kerlinger y Lee 2002, 403-561; Hernández, Fernández y Baptista 2003, 182-296; García 2008, 79-80; Easterbrook *et al.* 2008, 285-311).

Finalmente, las acciones que se siguen para contrastar y *explicar causalmente* la hipótesis, desde una perspectiva general del diseño mixto **IASC**, se muestran en la tabla de la figura 6.3. Estas acciones se visualizan para cada subproceso-estadio y para cada ciclo de investigación-acción del sistema complejo que emerge como conocimiento organizacional durante la implantación del modelo MACOSC en la organización sujeto de estudio<sup>28</sup>.

		IASC																	Aprendizaje Reflexivo				
		Investigación-acción de sistemas complejos (Diseño mixto)																					
		Planeación de la acción <i>Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)</i>								Intervención <i>Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)</i>													
Diagnóstico de la Oportunidad		O	R	M	C	V	E	A	Pp	N	Pv	O	R	M	C	V	E	A	Pp	N	Pv		
MACOSC	Uso	Op 1																				Ar 1	
		Op 2																					Ar 2
		Op 3																					Ar 3
		... Op n																					... Ar n
	Adecuación	Op 1																					Ar 1
		Op 2																					Ar 2
		Op 3																					Ar 3
		... Op n																					... Ar n
	Contribución	Op 1																					Ar 1
		... Op n																					... Ar n

Figura 6.3: MACOSC-IASC para las pequeñas organizaciones que desarrollan software, aplicable al caso que deseen adecuar programas de código abierto (Fuente: Elaboración propia).

<sup>28</sup>Es pertinente notar que con el cruce MACOSC-IASC también se observa la interacción entre el método (IASC) y la teoría representada por el modelo (MACOSC).

# Capítulo 7

## Estudio Empírico

En esta sección se describe la aplicación del diseño de investigación para el estudio empírico de la pertinencia del modelo de administración del conocimiento (MACOSC) en una pequeña organización (COC) que desarrolla software, para el caso en que se propone adecuar programas de código abierto (PCA). Se inicia con una breve descripción contextual y se describen las diversas iteraciones del estudio<sup>1</sup>.

### 7.1. Perspectiva general de la pequeña organización sujeto de estudio

En 1995 COC inició la comercialización de servicios y tecnologías propietarias CORBA<sup>2</sup> para implementar sistemas distribuidos en México.

En noviembre de 1998, logró iniciar un proyecto piloto en la empresa de telecomunicaciones de mayor tamaño del país. De agosto de 2000 a agosto de 2001, concursó y logró la asignación de un contrato de venta de productos, transferencia tecnológica y consultoría para la implementación de un sistema distribuido de misión crítica en múltiples áreas de la empresa. El proyecto concluyó en enero de 2010.

En junio de 2005 introdujo la tecnología en la segunda empresa de telecomunicaciones de mayor tamaño del país. En octubre de 2007 inició un proyecto de misión crítica en esta empresa, que fue exitosamente implementado en la subsidiaria de México y posteriormente en las subsidiarias de diversos países de Centro América. El proyecto concluyó en agosto de 2008.

Los sistemas distribuidos implantados en ambas empresas se encuentran actualmente en producción.

---

<sup>1</sup>En la que la organización sujeto de estudio, COC, accede al primero y segundo de los tres subprocesos-estadios del modelo MACOSC.

<sup>2</sup>En la sección siguiente se da una perspectiva general de CORBA.

En octubre de 2011 COC, debido al cambio recurrente en las condiciones comerciales de clientes<sup>3</sup> y proveedores<sup>4</sup>, aceptó iniciar un proyecto piloto interno para evaluar la posibilidad de adoptar programas de código abierto que le permitieran implementar sistemas distribuidos de misión crítica. El primer paso sería utilizar el estándar de la industria CORBA.

## 7.2. Perspectiva general de CORBA

CORBA es una especificación de una arquitectura de sistemas distribuidos y se considera un estándar de la industria de tecnologías de información, que ha sido utilizada a escala global. En inglés la palabra CORBA es la abreviación de *Common Object Request Broker Architecture*. Este estándar se define y aprueba por los miembros de la organización Object Management Group (OMG):

- La OMG es una de las organizaciones de tecnologías de información que participa y mantiene un cuerpo de estándares con aceptación a escala global.
- La OMG coordina los grupos de trabajo que crean los estándares. Sin embargo, dichos grupos no son grupos de ingeniería de desarrollo de productos.

La versión 1.0 de CORBA se liberó en octubre de 1991. En marzo de 2012 la OMG aprobó la versión 3.3 de CORBA (OMG, 2012a). En abril de 2012, la OMG aprobó la adopción de dos especificaciones: a) La invocación asíncrona de métodos para el modelo de componentes CORBA (Asynchronous Method Invocation For CORBA Component Model: AMI4CCM) (OMG, 2012b), y b) El mapeo de lenguajes IDL a C++ versión 2011 (IDL To C++11 Language Mapping: IDL2C++11) (OMG, 2012c). Organizaciones independientes son las que proveen la implementación de los estándares. Para el caso del estándar CORBA, promueven la implementación como un producto denominado Object Request Broker (ORB). Cada una de estas organizaciones registran y le asignan un nombre al ORB que promueven, por ejemplo: Orbix, VisiBroker, JacORB, etcétera.

La tecnología CORBA es utilizada en diversos sectores de la industria; por ejemplo, las organizaciones que requieren sistemas de gran tamaño y resistencia la han utilizado para manejar volúmenes enormes de invocaciones para acceder a datos y servicios con la garantía de un alto nivel de continuidad y de disponibilidad. La implementación de CORBA provee a estas organizaciones soluciones escalables y orientadas a servicios que permiten, a partir de una arquitectura abierta y reutilizable, integrar sistemas propietarios o sistemas en operación (“legados”) con los nuevos sistemas, aplicaciones y bases de datos que surgen por la natural evolución de las tecnologías de información. Organizaciones de diversos sectores de la industria que utilizan CORBA

---

<sup>3</sup>Para reducir el costo de licencias y servicios.

<sup>4</sup>Para reducir comisiones y pretender tomar el control directo de los clientes.

han publicado “casos de éxito” que dan mayor detalle de cada solución específica. Cada uno de los casos típicamente describen sistemas que se consideran de misión crítica para la organización (OMG, 2006).

En la actualidad CORBA se considera una tecnología madura, que está en la fase de reemplazo<sup>5</sup> o de complementación con *SOA* y *web services*. Sin embargo, diversas implementaciones de servidores de aplicaciones que permiten la integración de aplicaciones vía *web services*, *DDS*, etc., así como implementaciones del estándar para sistemas incrustados que requieren tiempos de respuesta en tiempo real y de alto rendimiento<sup>6</sup>, utilizan en la actualidad para su implementación esta tecnología.

### 7.3. Perspectiva general de la investigación

El investigador laboró en COC y creó una infraestructura (*framework*) de integración de aplicaciones empresariales (Orozco, 2001), que utilizó durante la realización de los principales proyectos descritos para implementar los sistemas distribuidos empresariales de misión crítica, en las empresas de telecomunicaciones de mayor tamaño en México. Mantiene contacto con la actual dirección de la organización.

El investigador consciente de la oportunidad y reto que significa el software de código abierto se propuso investigar la necesidad de administrar el conocimiento organizacional en las pequeñas organizaciones que desarrollan software, cuando se proponen adecuar ese tipo de software. Creó el modelo MACOSC a partir de la definición de conocimiento organizacional<sup>7</sup> como acción eficaz, eficiente y reiterada (AEER) de la organización, que emerge al articular la AEER de los diversos agentes involucrados en la organización. Diseñó un método de investigación-acción para sistemas complejos (IASC) y propuso, en julio de 2011, a COC su participación como sujeto de estudio en la realización de la investigación. COC aceptó participar en el estudio en octubre de 2011.

Para la investigación se planteó un doble objetivo en la fase de intervención, del diseño mixto investigación-acción de sistemas complejos (IASC):

- Contrastar y explicar causalmente la hipótesis:

**Si** La pequeña organización que desarrolla software (COC), para el caso en que se propone adecuar los programas de código abierto (PCA) JacORB v 2.3.1, formula sus acciones con el modelo MACOSC, **entonces** va a conseguir en grado elevado su objetivo.

---

<sup>5</sup>Veáse por ejemplo el artículo de Henning (2008).

<sup>6</sup>Sistemas descritos en inglés como: *Real-time, embedded, and high-performance systems*.

<sup>7</sup>El conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización. Véase la sección 4.3. *Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución*, en la página 44.

- Que la organización sujeto de estudio (COC), para el caso de los programas de código abierto (PCA) JacORB versión 2.3.1, formule sus acciones con el modelo de administración del conocimiento (MACOSC) para lograr el objetivo de realizar la lista de adecuaciones que se consideran necesarias, para crear y utilizar servidores CORBA de alto rendimiento (SCAR) que sean funcionales en un entorno de producción organizacional.

### 7.3.1. Contrastación y Explicación Causal de la Hipótesis

Para proceder a contrastar y *explicar causalmente* la hipótesis, se utiliza el diseño mixto investigación-acción de sistemas complejos (IASC). Se participa de manera práctica y activa en la organización (COC) que ha decidido adoptar programas de código abierto, y se propone adecuarlos a las condiciones particulares de sus problemas y oportunidades específicos. Así, con el objetivo de crear y utilizar servidores CORBA de alto rendimiento (SCAR), se implanta el modelo de administración del conocimiento MACOSC en la organización. Para realizar la investigación, se siguen las cuatro actividades de cada subproceso, que conforman el proceso iterativo e incremental del IASC descritas en la sección 6.3: El Diseño de la Investigación.

Se establece como condición necesaria, para contrastar y explicar causalmente la hipótesis, la siguiente:

- Que por administración del conocimiento organizacional entendamos que en la organización se articulan y ejecutan los pasos propuestos en el MACOSC para acceder a los subprocesos-estadios:
  - **Uso** de los programas de código abierto (PCA) JacORB v 2.3.1,
  - **Uso** de los programas de código abierto (PCA) git v 1.7.4.1,
  - **Adecuación** del control de versiones de los PCA JacORB v 2.3.1 de CVS a git v 1.7.4.1,
  - **Contribución** a los PCA JacORB v 2.3.1 a partir de los resultados obtenidos en las fases anteriores de **Uso** (JacORB, git) y en la fase de **Adecuación**.

Guiados por el objetivo de **crear** servidores CORBA de alto rendimiento (SCAR) que sean funcionales en un entorno de producción organizacional.

De la misma forma, para contrastar y *explicar causalmente* la hipótesis, se considera como principio el siguiente:

- Que el conocimiento organizacional es un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización que desarrolla software (COC); y que el sistema complejo emerge cuando el sistema de acciones en ejecución cumple las condiciones e incluye las causalidades descritas.

Para la aplicación del diseño *investigación-acción de sistemas complejos* (IASC), éste se cruza con el modelo de administración del conocimiento MACOSC. Se identifica como **unidad de análisis** a este cruce MACOSC-IASC; y como **unidad de observación** al complejo empírico de estudio conformado por las triadas: Objetivos, acciones, resultados.

### 7.3.2. Plan general de acción para la contrastación y explicación causal

Las acciones que se siguen para contrastar y *explicar causalmente* la hipótesis en el estudio empírico se han sustentado y descrito en la sección 6.3: El Diseño de la Investigación. En la figura 6.1 se muestran las fases del ciclo del diseño mixto creado (IASC). Desde una perspectiva general, las acciones que se siguen para contrastar y *explicar causalmente* la hipótesis en el estudio empírico se visualizan en el cruce del MACOSC con el diseño mixto IASC, en la tabla de la figura 6.3. Estas acciones se visualizan para cada subproceso-estadio y para cada ciclo de investigación-acción del conocimiento organizacional como sistema complejo, en el cruce MACOSC-IASC, para el caso específico que nos ocupa en la tabla de la figura 7.1 donde se muestra el cruce y la lista correspondiente de acciones propuestas. Estas acciones se siguen para contrastar y *explicar causalmente* la hipótesis en el estudio empírico. Esta *explicación causal* es una *atribución*<sup>8</sup> a la realidad de transformaciones observadas, en el caso de que estén en correspondencia con la inferencia establecida en la hipótesis<sup>9</sup>.

### 7.3.3. Diagnóstico general de la oportunidad

#### Entreverar la perspectiva cualitativa y cuantitativa

La perspectiva cualitativa tiene varias facetas:

- Mantener como guía de todo el proceso el objetivo de negocio de la pequeña organización que desarrolla software: Crear SCAR funcionales en un entorno de producción organizacional.

<sup>8</sup>Esta atribución no implica ningún tipo de hipóstasis, porque no otorga “realidad objetiva” a tales transformaciones (García, 2008, 189).

<sup>9</sup>Si La pequeña organización que desarrolla software (COC), para el caso en que se propone adecuar los programas de código abierto (PCA) JacORB v 2.3.1, formula sus acciones con el modelo MACOSC, *entonces* va a conseguir en grado elevado su objetivo.

		Diseño mixto investigación-acción de sistemas complejos (IASC) para intervenir la organización																							
Diagnóstico de la Oportunidad	Planeación de la acción Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)	Intervención Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)	Aprendizaje Reflexivo																						
			O	R	M	C	V	E	A	Pp	N	Pv	O	R	M	C	V	E	A	Pp	N	Pv			
			Modelo de administración del conocimiento organizacional (MACOSC) Subprocesos-estadios (S-E)	Uso	ACE																				
JacORB.																									
Mico																									
Orbacus																									
git																									
Adecuación	JacORB: de CVS a git																								
	JacORB: de Ant a Gradle																								
	JacORB: Monitoreo de JVM, ORB, POA, POA Mgr. y Servants.																								
	Modularizar vía "ClassLoaders".																								
Contribución	JacORB: Control de versiones distribuido.																								
	JacORB: Modularización.																								

Figura 7.1: Acciones propuestas en el cruce del MACOSC con el IASC (Fuente: Elaboración propia).

- Adecuar el control de versiones original —CVS— al un control de versiones distribuido —git— se considera esencial.
- Poder monitorear y verificar la funcionalidad de las aplicaciones en un entorno de producción organizacional. Se considera esencial y tiene como precondition los requerimientos previos.
- Que no dependan de una versión específica de Java los programas de código abierto. Se considera que esto podría ser un elemento para contribuir de forma innovadora a los programas.

La perspectiva cuantitativa entreverada a la cualitativa tiene sendas facetas:

- Crear, con programas de código abierto, un servidor CORBA de alto rendimiento (SCAR) que mantenga tiempos de respuesta constantes<sup>10</sup>.

<sup>10</sup>Con una carga de al menos treinta clientes, que operan en paralelo, que realizan, por lo menos, cien invocaciones cada uno; que el tiempo entre cada invocación no es mayor que el tiempo de proceso de cada invocación por el servidor.



- Se considera que se han adecuado a sus prácticas de administración de control de versiones de software (git) la administración de software actual (CVS), si se permite identificar los cambios que han ocurrido en la evolución del código de JacORB v 2.3.1, por lo menos durante un año previo a la liberación de dicha versión.
- Adecuar interfaces para la integración con sistemas de monitoreo organizacional (empresarial) los principales objetos: ORB, POA, POA Manager y Servants. Aplicable para cada JVM en ejecución.
- Modularizar vía ClassLoaders los programas de código abierto.

### **La versión inicial del sistema complejo**

La acción eficaz, eficiente y reiterada en la dimensión individual de cada agente<sup>11</sup> se articula de forma colectiva interdefinida, con interdependencia de funciones para que emerja el sistema complejo, considerado como el sistema de acciones eficaces, eficientes y reiteradas en ejecución<sup>12</sup>. Así la organización, el conjunto de relaciones que la caracterizan como sistema, se observa como una totalidad organizada en su acción eficaz y en su lógica de significaciones<sup>13</sup> en un momento dado. El análisis de sus procesos a través del tiempo hace observable esta práctica administrativa en los niveles de efectividad reiterada y, posteriormente, de eficiencia en el sistema organizacional<sup>14</sup> para transitar por los subprocesos-estadios del MACOSC cruzado con el IASC, como se muestra en la figura 7.1.

### **Identificar niveles de organización de la “realidad estructurada”**

Son tres niveles:

- Uso
- Adecuación
- Contribución

### **Identificar la interacción entre los niveles de organización de la “realidad estructurada”**

Se consideran dos interacciones principales entre niveles:

<sup>11</sup>Por ejemplo: aprobar la participación de COC como sujeto de estudio; programar SCAR con PCA.

<sup>12</sup>Por ejemplo: utilizar exitosamente SCAR, creados con PCA, en producción; resolver requerimientos de problemas de los SCAR en producción (veáse [García, 2008](#), 144).

<sup>13</sup>Las implicaciones entre acciones.

<sup>14</sup>Como se pretende ilustrar en la figura 4.2.

- Uso - Adecuación y
- Adecuación - Contribución

### **Calendarizar la evolución, por reorganizaciones sucesivas de cada nivel de organización del sistema**

Los periodos de tiempo en que transcurre el tránsito por los subprocesos-estadios:

- Uso de JacORB original: octubre de 2011 a diciembre de 2011
- Uso de git original: enero de 2012 a junio de 2012
- Adecuación del control de versiones de JacORB, de CVS a git: julio de 2012 a diciembre de 2012

### **7.3.4. Planeación general de las acciones**

El plan de acción, guiado por la teoría e hipótesis inicial, consiste de identificar cursos alternativos de acción para aprovechar la oportunidad, seleccionar el curso de acción para aprovechar la oportunidad, y establecer los instrumentos de observación del complejo empírico.

#### **Identificar cursos alternativos de acción para aprovechar la oportunidad**

Se registran las opciones de implementación de CORBA con PCA. En la figura 7.1 se muestran las opciones propuestas.

#### **Seleccionar el curso de acción para aprovechar la oportunidad**

- Racionalización de la elección de JacORB
- Racionalización de la elección de git

El plan de acción guiado por la teoría se muestra en la tabla de la figura 7.1 en el cruce MACOSC-IASC.

#### **Establecer los instrumentos de observación del complejo empírico**

- Sistema de observación tecnificada que proporciona observaciones neutrales que permiten comparar la teoría en su evolución en el tiempo (González C. E., 2011).

De la **unidad de observación** identificamos:

- Acciones de los agentes involucrados
  - Objetivos de la organización
  - Sistema de acciones (MACOSC) en ejecución
  - Resultados de ejecutar el sistema de acciones (MACOSC)
  - Programas de código abierto (PCA)
- Se utilizan escalas de medición de intervalo o proporcionales, sobre las observaciones realizadas, que permiten el acuerdo fáctico (González C. E., 2011).

## **7.4. Primera iteración de la investigación-acción de un sistema complejo: El conocimiento organizacional como un sistema complejo de acciones en ejecución para crear SCAR con JacORB en el subproceso-estadio Uso**

A continuación se describe la primer iteración del método que resulta de cruzar el MACOSC con el IASC, método que se utiliza para intervenir la organización.

### **7.4.1. Diagnóstico de la oportunidad**

#### **Entreverar la perspectiva cualitativa y cuantitativa para la primera iteración**

La perspectiva cualitativa en la primera iteración tiene la faceta de:

- Crear SCAR funcionales con la versión original 2.3.1 de JacORB.

La perspectiva cuantitativa entreverada a la cualitativa en la primera iteración tiene la faceta siguiente:

- Crear, con programas de código abierto, un servidor CORBA de alto rendimiento (SCAR) que mantenga tiempos de respuesta constantes<sup>15</sup>.

---

<sup>15</sup>En al menos una hora de operación, con una carga de al menos treinta clientes, que operan en paralelo, que realizan, por lo menos, cien invocaciones cada uno; que el tiempo entre cada invocación no es mayor que el tiempo de proceso de cada invocación por el servidor.

## **La versión del sistema complejo para la primera iteración**

La acción eficaz, eficiente y reiterada en la dimensión individual de cada agente para programar servidores CORBA de alto rendimiento (SCAR) con los PCA JacORB v 2.3.1 se articula como un sistema de acciones que se ejecuta de forma colectiva interdefinida, con interdependencia de funciones para que emerja como sistema complejo de acciones en ejecución, *i.e.*, como conocimiento organizacional<sup>16</sup>.

## **Identificar niveles de organización de la “realidad estructurada”**

En este caso es un nivel, el que he denominado: Uso.

## **Identificar la interacción entre los niveles de organización de la “realidad estructurada”**

Se considera que en esta fase no hay aún interacciones entre niveles.

## **Calendarizar la evolución, por reorganizaciones sucesivas de cada nivel de organización del sistema**

Los periodos de tiempo en que transcurre el tránsito por los subprocesos-estadios:

- Uso de JacORB original: octubre de 2011 a diciembre de 2011.

### **7.4.2. Planeación de la acción**

Plan de acción guiado por la teoría e hipótesis inicial, como se muestra en la tabla de la figura 7.2 en el cruce MACOSC-IASC.

## **Identificar cursos alternativos de acción para aprovechar la oportunidad**

Se registran las opciones de implementación de CORBA con programas de código abierto (PCA). El plan de acción guiado por la teoría se muestra en la tabla de la figura 7.2 en el cruce MACOSC-IASC.

## **Seleccionar el curso de acción para aprovechar la oportunidad**

- Racionalización de la elección de JacORB.

---

<sup>16</sup>La sección 4.3. *Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución* en la página 44, describe la nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización. Dicho sistema complejo emerge cuando el sistema de acciones en ejecución cumple las condiciones e incluye las causalidades que ahí se describen.



miento. A estas acciones de ser eficaces se les puede nombrar como conocimiento organizacional para el **Uso** de los programas de código abierto y es observable al corroborar que:

- Con los PCA originales, se satisface el requerimiento de crear SCAR.
- Se crean los reportes, con el registro correspondiente, para la organización y para la comunidad académica.
- Se modifican las teorías, se define el cambio siguiente.
- Se prepara lo necesario para realizar el ciclo siguiente o concluir con los trabajos de investigación.

En la figura 7.3 se muestra el reporte que registra que se accede al estadio **Uso**.

		Diseño mixto investigación-acción de sistemas complejos (IASC) para intervenir la organización																									
Diagnóstico de la Oportunidad		Planeación de la acción <i>Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)</i>							Intervención <i>Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)</i>							Aprendizaje Reflexivo											
		O	R	M	C	V	E	A	Pp	N	Pv	O	R	M	C		V	E	A	Pp	N	Pv					
Modelo de administración del conocimiento organizacional (MACOSC) Subprocesos-estadios (S-E)	Uso 1ª Iteración	ACE	█																								Diferencias del S-E Uso respecto a utilizar los PCA como "caja negra": Al controlar la construcción del ejecutable de los PCA, gradualmente incrementa la independencia. Al concluir con éxito la actividad Aplicar, se vislumbra una alternativa emancipadora (respecto a clientes y proveedores de código propietario). Al concluir con éxito la actividad Aplicar, se confirma que se accede al S-E Uso. Se confirma inferencia de la explicación causal.
		JacORB.	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	
		Mico	█																								
		Orbacus	█																								
	Uso	git																									
		JacORB: de CVS a git																									
	Adecuación	JacORB: de Ant a Gradle																									
		JacORB: Monitoreo de JVM, ORB, POA, POA Mgr, y Servants.																									
	Contribución	Modularizar vía "ClassLoaders".																									
		JacORB: Control de versiones distribuido.																									
	JacORB: Modularización.																										

Figura 7.3: Resultados de la primera iteración en el cruce del MACOSC con el IASC (Fuente: Elaboración propia).

## 7.4.4. Aprendizaje Reflexivo

Los resultados y lecciones aprendidas en la primera iteración del estudio, son los siguientes:

### ■ Resultados Empíricos

La organización que intervine para implementar el modelo **MACOSC** para la administración del conocimiento organizacional accede al estadio **Uso** y concluye con éxito la acciones de ese estadio<sup>17</sup>, que van de *Observar* hasta *Aplicar*. Se logra el objetivo organizacional de construir servidores de alto rendimiento (SCAR) con programas de código abierto JacORB v 2.3.1. En la figura 7.4 se muestra el comportamiento del SCAR.

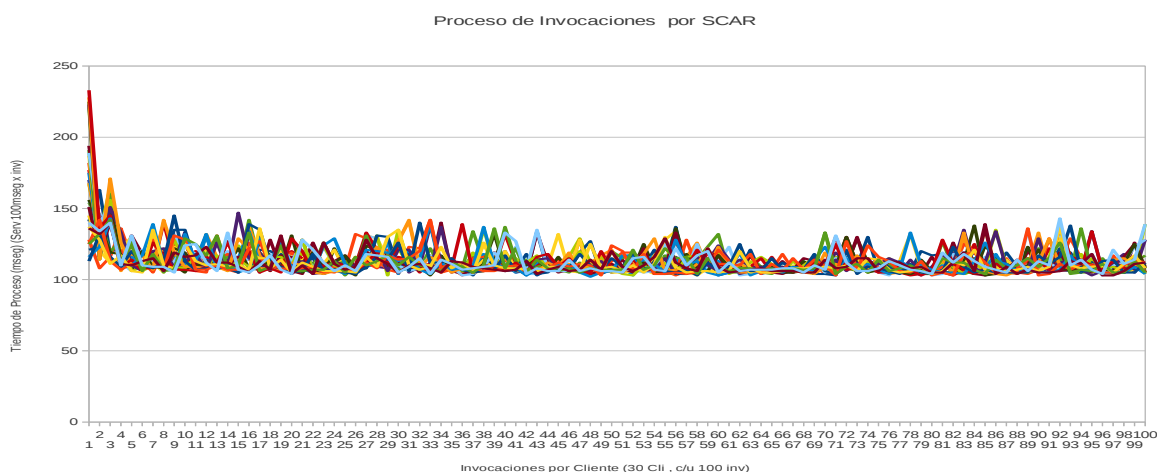


Figura 7.4: Comportamiento de Servidores CORBA de Alto Rendimiento (SCAR) (Fuente: Elaboración propia).

- Diferencias del subproceso-estadio **Uso** respecto de utilizar los PCA como “caja negra”: Al controlar la construcción del ejecutable de los PCA gradualmente incrementa la independencia.
- Al concluir con éxito la actividad *Aplicar*, se vislumbra una alternativa emancipadora respecto del código propietario que afectará la relación con clientes y proveedores.

### ■ Resultados Teóricos

<sup>17</sup>Ver 4.4 Modelo para la administración del conocimiento organizacional.

La consecuencia de corroborar la hipótesis para el estadio-subproceso **Uso** es que las acciones propuestas en el **MACOSC** son efectivas. Esto tiene como consecuencia que el **MACOSC** es pertinente para la administración del conocimiento organizacional en la pequeña organización que desarrolla software en el caso que se propone hacer **Uso** de los programas de código abierto.

- Al concluir con éxito la actividad *Aplicar*, se confirma que se accede al Subproceso-Estadio **Uso**. Asimismo, se confirma la inferencia de la explicación causal.

## **7.5. Segunda iteración de la investigación-acción de un sistema complejo: El conocimiento organizacional como un sistema complejo de acciones en ejecución para realizar con git el control de versiones distribuido, en el subproceso-estadio Uso**

### **7.5.1. Diagnóstico de la oportunidad**

La organización se propone utilizar un mecanismo diferente para el control de versiones al utilizado en JacORB. Así, se propone la Adecuación del control de versiones de JacORB, de CVS a git. Por esta razón es necesario, como primer paso, acceder al estadio **Uso** de git.

#### **Entreverar la perspectiva cualitativa y cuantitativa para la segunda iteración**

La perspectiva cualitativa en la segunda iteración tiene la faceta de:

- Crear sistemas de conjuntos de archivos, pueden o no ser programas de código abierto, controlados como código fuente de manera funcional con la versión original 1.7.4.1 de git.

La perspectiva cuantitativa entreverada a la cualitativa en la primera iteración tiene la faceta siguiente:

- Crear sistemas a partir de conjuntos de archivos, pueden o no ser programas de código abierto, controlados como código fuente de manera funcional con la versión original 1.7.4.1 de git durante al menos seis meses, en los que se controla efectivamente la evolución de dichos sistemas de archivos.



## **La versión del sistema complejo para la segunda iteración**

La acción eficaz, eficiente y reiterada en la dimensión individual de cada agente para controlar —de forma distribuida— la versión de un sistema de archivos elegidos por conveniencia se articula como parte de un sistema de acciones que se ejecuta de forma colectiva interdefinida, con interdependencia de funciones para que emerja como sistema complejo de acciones en ejecución, *i.e.*, como conocimiento organizacional<sup>18</sup>.

## **Identificar niveles de organización de la “realidad estructurada”**

En este caso es un nivel, el que he denominado: Uso.

## **Identificar la interacción entre los niveles de organización de la “realidad estructurada”**

Se considera que en esta fase no hay aún interacciones entre niveles.

## **Calendarizar la evolución, por reorganizaciones sucesivas de cada nivel de organización del sistema**

Los periodos de tiempo en que transcurre el tránsito por los subprocesos-estadios:

- Uso de git original: enero de 2012 a junio de 2012.

### **7.5.2. Planeación de la acción**

Plan de acción guiado por la teoría e hipótesis inicial.

Se registra la implementación del control de versiones distribuido con git. El plan de acción guiado por la teoría se muestra en la tabla de la figura 7.5 en el cruce MACOSC-IASC.

## **Establecer los instrumentos de observación del complejo empírico**

Véase la figura 7.5.

### **7.5.3. Intervención**

Para contrastar y *explicar causalmente* la pertinencia del MACOSC a la administración del conocimiento organizacional, se identifica como objetivo particular, de la segunda iteración acceder

---

<sup>18</sup>La sección 4.3. *Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución* en la página 44, describe esta nueva concepción de conocimiento organizacional.

		Diseño mixto investigación-acción de sistemas complejos (IASC) para intervenir la organización																					
		Planeación de la acción Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)								Intervención Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)								Aprendizaje Reflexivo					
Diagnóstico de la Oportunidad		O	R	M	C	V	E	A	Pp	N	Pv	O	R	M	C	V	E	A	Pp	N	Pv		
Modelo de administración del conocimiento organizacional (MACOSC) Subprocesos-estadios (S-E)	Uso 1ª Iteración	ACE																					
		JacORB.																				Diferencias del S-E Uso respecto a utilizar los PCA como "caja negra": Al controlar la construcción del ejecutable de los PCA, gradualmente incrementa la independencia. Al concluir con éxito la actividad Aplicar, se vislumbra una alternativa emancipadora (respecto a clientes y proveedores de código propietario). Al concluir con éxito la actividad Aplicar, se confirma que se accede al S-E Uso. Se confirma inferencia de la explicación causal.	
		Mico																					
		Orbacus																					
	Uso 2ª Iteración	git																					
		JacORB: de CVS a git																					
	Adecuación	JacORB: de Ant a Gradle																					
		JacORB: Monitoreo de JVM, ORB, POA, POA Mgr. y Servants.																					
		Modularizar via "ClassLoaders".																					
	Contribución	JacORB: Control de versiones distribuido.																				Prospectiva: Por definir.	
		JacORB: Modularización.																				Prospectiva: Por definir.	

Figura 7.5: Plan de la segunda iteración en el cruce del MACOSC con el IASC (Fuente: Elaboración propia).

al estadio **Uso** de los PCA git. Tal y como viene el código fuente se compila y se transita por las fases que permitan el control de versiones distribuido de uno o más sistemas de archivos.

De esta forma, se aplica el siguiente esquema de trabajo:

- En la organización, el equipo de programadores utiliza git, a partir de los fuentes compilados, para **usar** el control de versiones<sup>19</sup> de uno o más sistemas de archivos con el fin de controlar su evolución. A estas acciones de ser eficaces se les puede nombrar como conocimiento organizacional para el **Uso** de los programas de git y es observable al corroborar que:
  - Con git, se satisface el requerimiento de controlar la evolución de uno o más sistemas de archivos.

<sup>19</sup>Acorde al objetivo general de crear Servidores CORBA de Alto Rendimiento a partir de JacORB v 2.3.1. Controlados los fuentes de JacORB con git.

- Se crean los reportes, con el registro correspondiente, para la organización y para la comunidad académica.
- Se modifican las teorías, se define el cambio siguiente.
- Se prepara lo necesario para realizar el ciclo siguiente o concluir con los trabajos de investigación.

En la figura 7.6 se muestra el reporte que registra que se accede al estadio **Uso**.

#### 7.5.4. Aprendizaje Reflexivo

Los resultados y lecciones aprendidas en la segunda iteración del estudio, son los siguientes:

- Resultados Empíricos

La organización que intervine para implementar el modelo **MACOSC** para la administración del conocimiento organizacional accede al estadio de **Uso** y concluye con éxito la acciones de ese estadio<sup>20</sup>, que van de *Observar* hasta *Aplicar*. Se logra el objetivo organizacional de controlar un sistema de archivos con programas de código abierto git v 1.7.4.1.

- Al concluir con éxito la actividad *Aplicar*, se vislumbra una alternativa de participación en la red de innovación que desarrolla git.
- Diferencias del subproceso-estadio **Uso** respecto de utilizar los PCA como “caja negra”: Al controlar la construcción del ejecutable de los PCA, gradualmente incrementa la independencia.

- Resultados Teóricos

La consecuencia de corroborar la hipótesis para el estadio-subproceso **Uso** es que las acciones propuestas en el **MACOSC** son efectivas. Esto tiene como consecuencia que el **MACOSC** es pertinente para la administración del conocimiento organizacional en la pequeña organización que desarrolla software en el caso que se proponga hacer **Uso** de los programas de código abierto git.

- Al concluir con éxito la actividad *Aplicar*, se confirma que se accede al Subproceso-Estadio **Uso**. Asimismo, se confirma la inferencia de la explicación causal.

El resultado de seguir el plan de acción, para la segunda iteración, guiado por la teoría se muestra en la tabla de la figura 7.6 en el cruce **MACOSC-IASC**.

---

<sup>20</sup>Ver 4.4 Modelo para la administración del conocimiento organizacional.

		Diseño mixto investigación-acción de sistemas complejos (IASC) para intervenir la organización																				
Diagnóstico de la Oportunidad	Planeación de la acción Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)	Intervención Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)														Aprendizaje Reflexivo						
		O	R	M	C	V	E	A	Pp	N	Pv	O	R	M	C		V	E	A	Pp	N	Pv
		Modelo de administración del conocimiento organizacional (MACOSC) Subprocesos-estadios (S-E)	Uso 1ª Iteración	ACE																		
JacORB.																					Diferencias del S-E Uso respecto a utilizar los PCA como "caja negra": Al controlar la construcción del ejecutable de los PCA, gradualmente incrementa la independencia. Al concluir con éxito la actividad Aplicar, se vislumbra una alternativa emancipadora (respecto a clientes y proveedores de código propietario). Al concluir con éxito la actividad Aplicar, se confirma que se accede al S-E Uso. Se confirma inferencia de la explicación causal.	
Mico																						
Orbacus																						
Uso 2ª Iteración	git																				Diferencias del S-E Uso respecto a utilizar los PCA como "caja negra": Al controlar la construcción del ejecutable de los PCA, gradualmente incrementa la independencia. Al concluir con éxito la actividad Aplicar, se vislumbra una alternativa de participación en la red de innovación que desarrolla git. Al concluir con éxito la actividad Aplicar, se confirma que se accede al S-E Uso. Se confirma inferencia de la explicación causal.	
Adecuación	JacORB: de CVS a git																					
	JacORB: de Ant a Gradle																					
	JacORB: Monitoreo de JVM, ORB, POA, POA Mgr, y Servants. Modularizar via "ClassLoaders")																					
Contribución	JacORB: Control de versiones distribuido.																					Prospectiva: Por definir.
	JacORB: Modularización.																					Prospectiva: Por definir.

Figura 7.6: Resultados de la segunda iteración en el cruce del MACOSC con el IASC (Fuente: Elaboración propia).

## 7.6. Tercera iteración de la investigación-acción de un sistema complejo: El conocimiento organizacional como un sistema complejo de acciones en ejecución para realizar el control de versiones de JacORB con git, en el subproceso-estadio Adecuación

### 7.6.1. Diagnóstico de la oportunidad

La organización se propone utilizar un mecanismo diferente para el control de versiones al utilizado en JacORB. Así, se propone la **Adecuación** del control de versiones de JacORB, al pasar de CVS a git. En esta tercera iteración se pretende acceder al subproceso-estadio **Adecuación**.

## **Entreverar la perspectiva cualitativa y cuantitativa para la tercera iteración**

La perspectiva cualitativa en la tercera iteración tiene la faceta de:

- Controlar las versiones del sistema de archivos en código fuente que conforman JacORB es fundamental.

La perspectiva cuantitativa entreverada a la cualitativa en la primera iteración tiene la faceta siguiente:

- Crear con el sistema de archivos de JacORB v 2.3.1 y todas las versiones previas, generadas durante un período que abarca al menos un año de liberaciones de código abierto, un sistema de control de código fuente de JacORB funcional con la versión original 1.7.4.1 de git.

## **La versión del sistema complejo para la tercera iteración**

La acción eficaz, eficiente y reiterada en la dimensión individual de cada agente para controlar —también de forma distribuida— las versiones del sistema de archivos que conforman JacORB hasta la versión 2.3.1, se articulan como un sistema de acciones que se ejecuta de forma colectiva interdefinida, con interdependencia de funciones para que emerja como sistema complejo de acciones en ejecución, como conocimiento organizacional<sup>21</sup>.

## **Identificar niveles de organización de la “realidad estructurada”**

En este caso son dos niveles, los que he denominado: Uso y Adecuación.

## **Identificar la interacción entre los niveles de organización de la “realidad estructurada”**

Se identifica en esta fase la interacción entre los niveles Uso y Adecuación: El Uso de git permite realizar por etapas el cambio del control de versiones del conjunto de archivos que conforman JacORB, para arribar al estadio de Adecuación una vez que se logre crear, con el sistema de archivos de JacORB v 2.3.1 y todas las versiones previas generadas durante al menos un año de código abierto, un sistema de control de código fuente de JacORB v 2.3.1 de manera funcional con la versión original 1.7.4.1 de git.

---

<sup>21</sup>La sección 4.3. *Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución* en la página 44, describe esta nueva concepción de conocimiento organizacional.

## **Calendarizar la evolución, por reorganizaciones sucesivas de cada nivel de organización del sistema**

Los periodos de tiempo en que transcurre el tránsito por los subprocesos-estadios:

- Adecuación del control de versiones de JacORB, de CVS a git: julio de 2012 a diciembre de 2012.

### **7.6.2. Diagnóstico de la oportunidad de utilizar instrumentos adicionales de investigación-acción de sistemas complejos**

La oportunidad de utilizar instrumentos adicionales para observar con mayor detalle la interacción entre el método (IASC) y la teoría representada por el modelo (MACOSC) para propiciar la construcción de conocimiento organizacional<sup>22</sup> se describe a continuación.

Al llevar a cabo la tercera iteración del método MACOSC-IASC, surgió la oportunidad de explorar la pertinencia de incorporar a los procesos llevados a cabo un método ágil y un sistema adaptativo. La organización COC aceptó evaluar una nueva propuesta que podría beneficiar sus actuales prácticas de desarrollo de software. Ésta consistió en utilizar un repositorio para representar los subprocesos-estadios de Uso, Adecuación y Contribución del MACOSC con el fin de establecer mecanismos que permitan: organizarlos, consultarlos y mejorarlos en el tiempo. Esta propuesta se construyó en dos fases.

La primera consiste en utilizar una metodología orientada hacia la creación y ejecución de métodos de ingeniería de software ágiles basados en la respuesta enviada por la UNAM, en agosto de 2012 (Oktaba, Morales y Dávila, 2012)<sup>23</sup>, a la solicitud de propuestas publicada por el Object Management Group (OMG) el 26 de junio de 2011 (OMG-FACESEM, 2011)<sup>24</sup>.

La segunda fase consiste en crear e implementar un sistema adaptativo (SiA) para el análisis de las acciones de los agentes involucrados con un espacio de percepción, integración y respuesta (EPIR). (Véase [Amozurrutia, 2011](#), 396-401).

#### **Primera fase: KUALI-BEH**

El repositorio a crearse es la infraestructura de métodos y prácticas para el MACOSC-IASC; de ser funcional se extendería para abarcar el resto de las prácticas de ingeniería de software de COC.

---

<sup>22</sup>Como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización. El sistema complejo emerge cuando el sistema de acciones en ejecución cumple las condiciones e incluye las causalidades descritas.

<sup>23</sup>*KUALI-BEH Software Project Common Concepts revised submission-version 1.1.*

<sup>24</sup>*Foundation for the Agile Creation and Enactment of Software Engineering Methods (FACESEM) RFP (OMG Document ad/2011-06-26).*

Al repositorio se le denomina Infraestructura de Métodos y Prácticas (COC-IMP). La intención es que los integrantes de la organización participen en la creación de COC-IMP y contribuyan con su experiencia para incorporar diversas representaciones del conocimiento a dicho repositorio y registren en él la ejecución de proyectos. De tener éxito en la creación y uso del COC-IMP, se prevé que será útil para organizar estas representaciones del conocimiento y propiciar la reflexión, mejora continua y entrenamiento de nuevo personal<sup>25</sup>.

Para la implementación y ejecución del método MACOSC-IASC con instrumentos adicionales, en la primer fase se utiliza KUALI-BEH v1.1 (Oktaba, Morales y Dávila, 2012).

Las plantillas para la ejecución y registro de resultados de implantar el método MACOSC-IASC en la organización de estudio<sup>26</sup> se describen a continuación:

Tabla 7.1: Plantilla de Proyecto de Software (PPS)

<b>Organización</b>	
Nombre de la organización.	
<b>Proyecto de software</b>	
Nombre del proyecto de software.	
<b>Grupo de interés</b>	
Lista de los interesados.	
<b>Fecha de inicio</b>	<b>Fecha de finalización</b>
Fecha de inicio del proyecto.	Fecha de finalización del proyecto.
<b>Entrada necesaria para la Acción</b>	
Entradas necesarias para la acción que deben ser observables y verificables. Pueden ser múltiples: originadas por el grupo de interés o por las pre-condiciones para el producto de software. Son condiciones para el logro de objetivos.	
<b>Objetivos</b>	<b>Resultados</b>
Objetivos o requerimientos observables y verificables.	Condiciones, productos —de software u otros— que satisfacen requerimientos y objetivos.
<b>Método</b>	
Método seleccionado.	

*Continúa en la página siguiente: PPS*

<sup>25</sup>De hecho para implementar KUALI-BEH v1.1, me parece apropiado aplicar el MACOSC-IASC de manera recursiva, es decir, se definen tres métodos: Uso, Adecuación y Contribución; con sendas prácticas: Diagnóstico de la Oportunidad, Planeación de la Acción, Intervención y Aprendizaje Reflexivo.

<sup>26</sup>Se utiliza como marco de referencia KUALI-BEH v1.1 (Oktaba, Morales y Dávila, 2012) y se planea adecuar una vez que se concluya la integración con Essence (Jacobson *et al.*, 2012).

Tabla 7.1 – PPS: Continúa de la página anterior

<b>Grupo de trabajo</b>
Profesional A, (...), profesional Z.

Tabla 7.2: Plantilla de Método (PM)

<b>Organización</b>	
Nombre de la organización.	
<b>Método</b>	
Nombre del método.	
<b>Propósito</b>	
Propósito del método.	
<b>Entrada necesaria para la Acción</b>	
Entradas necesarias para la acción que deben ser observables y verificables. Pueden ser múltiples: originadas por el grupo de interés o por las pre-condiciones para el producto de software. Son condiciones para el logro de objetivos.	
<b>Objetivos</b>	<b>Resultados</b>
Objetivos o requerimientos observables y verificables.	Condiciones, productos —de software u otros— que satisfacen requerimientos y objetivos.
<b>Prácticas</b>	
Lista de prácticas que conforman el método. Opcional: Requerimientos de la(s) práctica(s) que reciben la entrada al método, entregable(s) de la(s) práctica(s) que generan la salida del método, etcétera.	

Tabla 7.3: Plantilla de Práctica (PP)

<b>Organización</b>
Nombre de la Organización.
<b>Práctica</b>
Nombre de la práctica.

*Continúa en la página siguiente: PP*



Tabla 7.3 – PP: Continúa de la página anterior

<b>Entrada necesaria para la Acción</b>	
Entradas necesarias para la acción que deben ser observables y verificables. Pueden ser múltiples: originadas por el grupo de interés o por las pre-condiciones para el producto de software. Son condiciones para el logro de objetivos.	
<b>Objetivos</b>	<b>Resultados</b>
Objetivos o requerimientos observables y verificables.	Condiciones, productos —de software u otros— que satisfacen requerimientos y objetivos.
<b>Guía</b>	
<b>Actividades</b>	
Lista de actividades.	
<b>Tareas (opcional)</b>	
Opcional: Hacer esto, (...) , hacer aquello.	
<b>Herramientas (opcional)</b>	
Opcional: Lista de las herramientas propuestas.	
<b>Conocimientos y Habilidades</b>	
Habilidades, competencias, logros.	
<b>Criterios de verificación</b>	
Criterio A, criterio B, etcétera.	
<b>Métricas</b>	
Métrica A, métrica B, etcétera.	

Tabla 7.4: Plantilla del Tablero de Estado de Ejecución de Método (PTEEM)

Organización						
Nombre de la organización.						
Id. Proyecto	Id. Método	TEEM	Fecha de inicio	Fecha actual	Fecha de fin	Días restantes
Entradas						

Continúa en la página siguiente: PTEEM

Tabla 7.4 – PTEEM: Continúa de la página anterior

Entradas necesarias para la acción que deben ser observables y verificables. Pueden ser múltiples: originadas por el grupo de interés o por las pre-condiciones para el producto de software. Son condiciones para el logro de objetivos.								
Objetivos			Resultados					
Objetivos o requerimientos observables y verificables.			Condiciones, productos —de software u otros— que satisfacen requerimientos y objetivos.					
Adaptada		Lista para Iniciar	En Progreso			Perspectiva del Progreso		
Id.	Instancia 20 %.	Puede iniciar 40 %.	En ejecución 60 %.	En verificación 80 %.	En espera (N/A %).	Cancelada (N/A %).	Terminada 100 %.	Progreso (en %).
1								
2								
3								
4								
							Total	Progreso Total
Productos de trabajo o Condiciones y su status respectivo								
Lista de productos de trabajo o condiciones con su status asociado.								

Tabla 7.5: Plantilla del Tablero de Estado de Ejecución de Práctica (PTEEP)

<b>Organización</b>							
Nombre de la organización.							
Id. Proyecto	Id. Método	Id. Práctica	TEEP	Fecha de inicio	Fecha actual	Fecha de fin	Días restantes
Entradas							
Entradas necesarias para la acción que deben ser observables y verificables. Pueden ser múltiples: originadas por el grupo de interés o por las pre-condiciones para el producto de software. Son condiciones para el logro de objetivos.							

Continúa en la página siguiente: PTEEP

Tabla 7.5 – PTEEP: Continúa de la página anterior

Objetivos		Resultados					
Objetivos o requerimientos observables y verificables.		Condiciones, productos —de software u otros— que satisfacen requerimientos y objetivos.					
Grupo de Trabajo							
Profesional A, (...), profesional Z.							
Métricas Estimadas			Métricas Actuales				
■ Lista de métricas estimadas.			■ Lista de métricas actuales.				
Progreso de Actividades							
Actividades	Progreso	Responsable		Comentarios			
Actividad “X”	Valor numérico	Profesional	Respon- sable	Comentarios y notas relevantes			
Actividad “X”	Valor numérico	Profesional	Respon- sable	Comentarios y notas relevantes			
Estados de la Instancia de la Práctica							
Instancia 20 %.	Puede iniciar 40 %.	En eje- cución 60 %.	En verifica- ción 80 %.	En espera (N/A %).	Cancelada (N/A %).	Terminada 100 %.	Progreso (en %).

## Vista Estática

Se presenta la vista estática, compuesta de la infraestructura de métodos y prácticas, a continuación.

Tabla 7.6: Proyecto de Software (PS) - SCAR

<b>Organización</b>	
COC	
<b>Proyecto de software</b>	
Creación de servidores CORBA de alto rendimiento utilizando los programas de código abierto JacORB controlados con git.	
<b>Grupo de interés</b>	
Director (COC), Tecnólogo-Administrador de proyectos (COC), Programador (COC), Investigador.	
<b>Fecha de inicio</b>	<b>Fecha de finalización</b>
julio 2012	diciembre 2012
<b>Entrada necesaria para la Acción</b>	

*Continúa en la página siguiente: PS*

Tabla 7.6 – PS: Continúa de la página anterior

<ul style="list-style-type: none"> <li>■ Necesidades del grupo de interés organizacional: En octubre de 2011 COC debido al cambio recurrente en las condiciones comerciales de clientes<sup>27</sup> y proveedores<sup>28</sup> aceptó iniciar un proyecto piloto interno para evaluar la posibilidad de adoptar programas de código abierto que le permitan implementar sistemas distribuidos de misión crítica.             <ul style="list-style-type: none"> <li>● Se han realizado dos iteraciones en las que la organización COC ha accedido a los estadios <b>Uso</b> de los PCA JacORB y git.</li> <li>● En la tercera iteración se requiere adecuar el control de versiones de los programas de código abierto JacORB al pasar de CVS a git.</li> </ul> </li> <li>■ Necesidades del Investigador: Contrastar y explicar causalmente la hipótesis siguiente             <ul style="list-style-type: none"> <li>● <b>Si</b> La pequeña organización que desarrolla software (COC), para el caso en que se propone adecuar los programas de código abierto (PCA) JacORB v 2.3.1, formula sus acciones con el modelo MACOSC, <b>entonces</b> va a conseguir en grado elevado su objetivo.</li> </ul> </li> <li>■ Condiciones del proyecto: El primer paso es utilizar el estándar de la industria CORBA.</li> <li>■ Producto de software: Programas de código abierto JacORB v 2.3.1 controlados con git v 1.7.4.1.</li> </ul>	
<b>Objetivos</b>	<b>Resultados</b>

*Continúa en la página siguiente: PS*

<sup>27</sup>Reducción de precios.

<sup>28</sup>Tomar control de la cuenta en forma directa.

Tabla 7.6 – PS: Continúa de la página anterior

<p>Se planteó el doble objetivo:</p> <ul style="list-style-type: none"> <li>■ Primero que la organización sujeto de estudio (COC), para el caso de los programas de código abierto (PCA) JacORB, en una tercera iteración formule sus acciones con el modelo de administración del conocimiento “cruzado” con el diseño de investigación acción de sistemas complejos (MACOSC-IASC v 1.0). Este cruce se implementa con la propuesta KUALIBEH v 1.1 lo que resulta en un total de tres métodos<sup>29</sup>: Uso, Adecuación, Contribución. A cada método se le asocian 4 prácticas<sup>30</sup> Diagnóstico de la Oportunidad, Planeación de la Acción, Intervención, Aprendizaje Reflexivo; con lo que se tiene un total de 12 prácticas.</li> <li>■ Segundo, crear y utilizar servidores CORBA de alto rendimiento (SCAR) con los programas de código abierto JacORB v 2.3.1 con la <b>adecuación</b> de la forma de controlar versiones. La adecuación consiste en pasar del control de versiones CVS a git v 1.7.4.1.</li> </ul>	<p>En construcción (. . .).</p>
<p><b>Método</b></p>	

*Continúa en la página siguiente: PS*

<sup>29</sup>Correspondientes a los tres subprocesos-estadios **Uso, Adecuación y Contribución** del MACOSC.

<sup>30</sup>Correspondientes a las cuatro fases **Diagnóstico de la Oportunidad, Planeación de la Acción, Intervención, Aprendizaje Reflexivo** del IASC aplicadas a cada subproceso-estadio.

Tabla 7.6 – PS: Continúa de la página anterior

MACOSC-IASC Modelo de administración del conocimiento MACOSC con el subproceso-estadio-método: <b>Adecuación</b> cruzado con el diseño de investigación-acción de sistemas complejos IASC para intervenir la organización: Diagnóstico de la Oportunidad, Planeación de la Acción, Intervención y Aprendizaje Reflexivo. Consideradas éstas como tareas de cada subproceso-estadio-método.
<b>Grupo de trabajo</b>
Toma de decisiones: Director (COC). Administración: Tecnólogo-Administrador de proyectos (COC). Realización: Programador (COC), Investigador.

Tabla 7.7: Método (M) - Adecuación

<b>Organización</b>	
COC	
<b>Método</b>	
Adecuación	
<b>Propósito</b>	
El propósito del método <b>Adecuación</b> es que los PCA JacORB controlados en sus versiones con git se utilicen para crear SCAR a partir de la v 2.3.1.	
<b>Entrada necesaria para la Acción</b>	
Código fuente de JacORB sin control de versiones.	
<b>Objetivos</b>	<b>Resultados</b>
El objetivo del método Adecuación es que el código fuente de JacORB hasta la v 2.3.1 se controle con git.	En construcción (...).
<b>Prácticas</b>	
Del método <b>Adecuación</b> : <ul style="list-style-type: none"> <li>■ Diagnóstico de la oportunidad.</li> <li>■ Plan de acción.</li> <li>■ Intervención.</li> <li>■ Aprendizaje reflexivo.</li> </ul>	

Tabla 7.8: Práctica (P) - Diagnóstico de la oportunidad

<b>Organización</b>	
COC	
<b>Práctica</b>	
Diagnóstico de la oportunidad del método Adecuación.	
<b>Entrada necesaria para la Acción</b>	
<b>Objetivos</b>	<b>Resultados</b>
<b>Guía</b>	
<b>Actividades</b>	
<ul style="list-style-type: none"> <li>▪ Porque git.</li> <li>▪ FODA de git.</li> </ul>	
<b>Tareas (opcional)</b>	
<b>Herramientas (opcional)</b>	
<ul style="list-style-type: none"> <li>▪ git</li> <li>▪ gitk</li> <li>▪ giggle</li> </ul>	
<b>Conocimientos y Habilidades</b>	
<ul style="list-style-type: none"> <li>▪ Conocimientos y habilidades para utilizar o aplicar Herramientas para realizar tareas de control de versiones con git, gitk, giggle.</li> </ul>	
<b>Criterios de verificación</b>	

*Continúa en la página siguiente: P*



Tabla 7.8 – P: Continúa de la página anterior

<ul style="list-style-type: none"> <li>■ Criterios para verificar que se realizan las <b>acciones efectivas, eficientes y reiteradas</b> (Conocimientos y habilidades) para utilizar o aplicar Herramientas para realizar tareas de control de versiones con git, gitk, giggle.</li> </ul>
<b>Métricas</b>
<ul style="list-style-type: none"> <li>■ Métricas para verificar que se realizan las <b>acciones efectivas, eficientes y reiteradas</b> (Conocimientos y habilidades) para utilizar o aplicar Herramientas para realizar tareas de control de versiones con git, gitk, giggle.</li> </ul>

Tabla 7.9: Práctica (P) - Plan de acción

<b>Organización</b>	
COC	
<b>Práctica</b>	
Práctica de Plan de acción del método Adecuación.	
<b>Entrada necesaria para la Acción</b>	
<b>Objetivos</b>	<b>Resultados</b>
<b>Guía</b>	
<b>Actividades</b>	
<b>Tareas (opcional)</b>	
<b>Herramientas (opcional)</b>	
<b>Conocimientos y Habilidades</b>	
<b>Criterios de verificación</b>	

Continúa en la página siguiente: P

Tabla 7.9 – P: Continúa de la página anterior

<b>Métricas</b>

Tabla 7.10: Práctica (P) - Intervención

<b>Organización</b>	
COC	
<b>Práctica</b>	
Práctica de Intervención del método Adecuación.	
<b>Entrada necesaria para la Acción</b>	
<b>Objetivos</b>	<b>Resultados</b>
<b>Guía</b>	
<b>Actividades</b>	
<b>Tareas (opcional)</b>	
<b>Herramientas (opcional)</b>	
<b>Conocimientos y Habilidades</b>	
<b>Criterios de verificación</b>	
<b>Métricas</b>	

Tabla 7.11: Práctica (P) - Aprendizaje reflexivo

<b>Organización</b>
---------------------

*Continúa en la página siguiente: P*

Tabla 7.11 – P: Continúa de la página anterior

COC	
<b>Práctica</b>	
Práctica de Aprendizaje reflexivo del método Adecuación.	
<b>Entrada necesaria para la Acción</b>	
<b>Objetivos</b>	<b>Resultados</b>
<b>Guía</b>	
<b>Actividades</b>	
<b>Tareas (opcional)</b>	
<b>Herramientas (opcional)</b>	
<b>Conocimientos y Habilidades</b>	
<b>Criterios de verificación</b>	
<b>Métricas</b>	

## Segunda fase: SIAs y EPIRs

La segunda fase consiste en explorar la posibilidad de crear un sistema adaptativo (SIA) para el registro y análisis de las acciones de los agentes involucrados con un espacio de percepción, integración y respuesta (EPIR). (Véase [Amozurrutia, 2011](#), 396-401). La implementación del sistema permitiría explorar aspectos adicionales para el acuerdo fáctico sobre aspectos cualitativos y hacer “reflexiones distintas a las que ofrece el análisis estadístico clásico” ([Amozurrutia, 2011](#), 371-377); se procedería en cinco etapas como sigue<sup>31</sup>.

- Delimitación del complejo empírico y definición de la unidad de observación

Para ir del complejo empírico a la unidad de observación se recomienda realizar preguntas interdisciplinarias al problema práctico (véase [Amozurrutia, 2011](#), 323, 324-329).

<sup>31</sup>Seguir estas cinco etapas es, de hecho, la recomendación del autor (véase [Amozurrutia, 2011](#), 321).

- Definición de técnicas y marco teórico

Se utilizaría el marco teórico expuesto en el capítulo 4. *Marco Teórico Conceptual* en la sección 4.3. *Una nueva concepción de conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución*, en la página 44.

Las técnicas que se utilizarían se exponen en la sección 6.3. *El diseño de la investigación-acción de sistemas complejos (IASC)*, en la página 64.

El marco teórico y las técnicas se complementarían con lo expuesto en (Amozurrutia, 2011, 323, 324, 329-330); y con modelos y experimentos en el laboratorio y en el campo —modelos de agente de acción colectiva— expuestos en (Poteete, Janssen y Ostrom, 2012, 267-378).

- Construcción del esquema categórico

Para responder a preguntas adicionales —a la inicial que realicé al problema sobre la pertinencia del MACOSC y que se responde con la unidad de análisis representada por el IASC-MACOSC— generadas por más investigadores, de disciplinas diversas, e involucrados para afinar el alcance de la investigación, Amozurrutia recomienda la creación de un esquema categórico. La correspondiente unidad de análisis sería entonces más sutil y permitiría valorar con mayor detalle el problema que enfrentan las pequeñas organizaciones que desarrollan software cuando se proponen adecuar software de código abierto. El autor, sin embargo, advierte que es la actividad más difícil porque requiere conciliar las preguntas de investigación con la naturaleza de los observables definidos y recopilados *i.e.* requiere conciliar el complejo empírico con el complejo cognoscitivo (véase Amozurrutia, 2011, 324, 330-337).

Para observar con mayor detalle la interacción de los agentes involucrados con el software de código abierto, como primer paso mi propuesta consiste de:

- A partir de las diversas estructuras del software y relaciones principales entre ellas, en cada uno de los subprocesos-estadios: Crear representaciones del conocimiento (RC) que utilizan lógicas, ontologías, modelos computables, conmensurabilidad. En un primer acercamiento, la categorías son generadas utilizando un **meta-modelo**. La intención es generar RC conmensurables a partir del software y de acuerdo con cada una de las cuatro capas de los meta-modelos: Knowledge Discovery Meta-model (KDM) y del Software Metrics Metamodel (SMM). Así, a partir de un conjunto elegido por la organización de programas de código abierto del sistema JacORB versión 2.3.1. se utilizarían los meta-modelos estandarizados en la industria por la OMG e internacionalmente por la ISO-IEC. (Véase OMG 2011, 1-19; e ISO-IEC 19506:2012E 2012, 1-14).

- Los agentes eligen las estructuras del software con la que pretenden interactuar en el tránsito por los estadios *Uso, Adecuación y Contribución*.
- Se considera, entonces, como niveles de estructuración y variables dependientes a los subprocesos-estadios *Uso, Adecuación y Contribución*. Como variables independientes tendríamos las actividades que se realizan en cada subproceso (*Observar, Representar, etcétera*) y para cada una de éstas se establecen los niveles de observación apropiados (generados por bitácoras, apuntes, etcétera).

- Integración de elementos y relaciones en el sistema adaptativo

Implica la representación en software del vínculo entre las variables —dependientes e independientes, establecidas en la etapa anterior— y categorías dentro del modelo, para analizar los observables registrados. El reto de llevar a cabo esta etapa depende en buena medida de la decisión de utilizar la infraestructura creada por el autor con hojas electrónicas (véase [Amozurrutia, 2011](#), 324, 338-339); consistente con el modelo MACOSC propuesto, la decisión de arribar al estadio *Uso* de las hojas electrónicas (EPIR) sería la más recomendable.

- Cierre del proceso heurístico

Corresponde al cierre del proyecto y el autor nos recomienda realizarlo si se encuentra un comportamiento asintótico en la definición de valoraciones. El caso normal es que en la marcha del proyecto se arribe a la fase de las conclusiones (véase [Amozurrutia, 2011](#), 324, 339-342).

## **Conclusión del diagnóstico de la oportunidad de utilizar instrumentos adicionales de investigación-acción de sistemas complejos**

Utilizar instrumentos adicionales para observar con mayor detalle la interacción entre el método (IASC) y la teoría representada por el modelo (MACOSC), con las propuestas exploradas de KUALI-BEH v1.1, SIAs-EPIRs y KDM-SMM es prometedor. Sin embargo, gracias a la exploración realizada, son evidentes dos hallazgos. Primero, que se requiere un mayor número de iteraciones en la investigación para profundizar y generar herramientas que faciliten su aplicación al caso que nos concierne. Segundo, que para realizar estas iteraciones sería conveniente conformar un equipo multidisciplinario<sup>32</sup> que se proponga construir un sistema que represente con mayor

---

<sup>32</sup>Considero que el lenguaje compartido —una clara necesidad de los equipos multidisciplinarios que estudian sistemas complejos— puede verse como un lenguaje de interfaz, el punto de contacto dialógico, que permita que el trabajo de cada disciplina individual se articule en un trabajo común, coherente y funcional. Finalmente, para la construcción del problema de investigación con dicho equipo multidisciplinario sería razonable recurrir adicionalmente a las propuestas planteadas por ([Poteete, Janssen y Ostrom, 2012](#)) y ([Maass et al., 2012](#)).



#### 7.6.4. Intervención

Para contrastar y *explicar causalmente* la pertinencia del **MACOSC** a la administración del conocimiento organizacional, se identifica como objetivo particular, de la tercera iteración, acceder al estadio **Adecuación** del control de versiones de los PCA JacORB con git. Tal y como viene el código fuente de git se compila y se transita por las fases que permitan el control de versiones distribuido del sistema de archivos en código fuente de JacORB. Se considera que al menos se debe controlar una año previo e incluyente de la liberación de JacORB v 2.3.1.

De esta forma, se aplica el siguiente esquema de trabajo:

- En la organización, el equipo de programadores utiliza git<sup>33</sup> para **adecuar** el control de versiones de los fuentes de JacORB<sup>34</sup> con el propósito de controlar su evolución.

A estas acciones, de ser eficaces, se les puede nombrar como conocimiento organizacional para la **Adecuación** del control de versiones de JacORB con los programas git. Esto es observable al corroborar que:

- Con git v 1.7.4.1 se controla la evolución del código fuente de JacORB y todas las versiones previas generadas durante al menos un año de código abierto, lo que incluye la liberación de JacORB v 2.3.1.
- Se crean los reportes, con el registro correspondiente, para la organización y para la comunidad académica.
- Se modifican las teorías, se define el cambio siguiente.
- Se prepara lo necesario para realizar el ciclo siguiente o concluir con los trabajos de investigación.

En la figura 7.8 se muestra el reporte que registra que se accede al estadio **Adecuación**.

#### 7.6.5. Aprendizaje Reflexivo

Los resultados y lecciones aprendidas en la tercera iteración del estudio, son los siguientes:

- Resultados Empíricos

La organización que intervine para implementar el modelo **MACOSC** para la administración del conocimiento organizacional, accede al estadio de **Adecuación** y concluye con éxito la acciones de ese estadio<sup>35</sup> que van de *Observar* hasta *Aplicar*. Resaltando que en estas

---

<sup>33</sup>A partir de los fuentes —de git v 1.7.4.1— compilados.

<sup>34</sup>Acorde al objetivo general de crear Servidores CORBA de Alto Rendimiento a partir de JacORB v 2.3.1., y el objetivo particular de controlar los fuentes de JacORB con git de forma tal que permita controlar su evolución.

<sup>35</sup>Veáse 4.4 Modelo para la administración del conocimiento organizacional.

acciones se incluye *Modificar*. Se logra el objetivo organizacional de controlar el sistema de archivos de código fuente de JacORB, lo que abarca por lo menos un período de un año hasta la liberación de JacORB v 2.3.1 con programas de código abierto git v 1.7.4.1.

- Al adecuar la construcción del ejecutable de los PCA a los procesos de la organización, gradualmente incrementará la interdependencia con la red de innovación que desarrolla JacORB.

Lo anterior se suma a lo ya logrado en el estadio anterior:

- Diferencias del subproceso-estadio **Uso** respecto a utilizar los PCA como “caja negra”: Al controlar la construcción del ejecutable de los PCA, gradualmente incrementa la independencia.
- Al concluir con éxito la actividad *Aplicar*, se vislumbra una alternativa emancipadora respecto al código propietario, que afectará la relación con clientes y proveedores.

#### ■ Resultados Teóricos

La consecuencia de corroborar la hipótesis para el estadio-subproceso **Adecuación** es que las acciones propuestas en el **MACOSC** son efectivas. Esto tiene como consecuencia que el **MACOSC** es pertinente para la administración del conocimiento organizacional en la pequeña organización que desarrolla software en el caso que se proponga hacer la **Adecuación** de controlar las versiones de los programas de código abierto JacORB con git.

- Al concluir con éxito la actividad *Aplicar*, se confirma que se accede al Subproceso-Estadio **Adecuación**. Asimismo, se confirma la inferencia de la explicación causal.

El resultado de seguir el plan de acción, para la tercera iteración, guiado por la teoría se muestra en la tabla de la figura 7.8 en el cruce MACOSC-IASC.



		Diseño mixto investigación-acción de sistemas complejos (IASC) para intervenir la organización																							
		Diagnóstico de la Oportunidad	Planeación de la acción <i>Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)</i>								Intervención <i>Observar, Representar, Modificar, Compilar, Verificar, Ejecutar, Aplicar, Proponer (Pp), Negociar, Proveer (Pv)</i>								Aprendizaje Reflexivo						
			O	R	M	C	V	E	A	Pp	N	Pv	O	R	M	C	V	E		A	Pp	N	Pv		
Modelo de administración del conocimiento organizacional (MACOSC) Subprocesos-estadios (S-E)	Uso 1ª Iteración	ACE																							
		JacORB:																							Diferencias del S-E Uso respecto a utilizar los PCA como "caja negra": Al controlar la construcción del ejecutable de los PCA, gradualmente incrementa la independencia. Al concluir con éxito la actividad Aplicar, se vislumbra una alternativa emancipadora (respecto a clientes y proveedores de código propietario). Al concluir con éxito la actividad Aplicar, se confirma que se accede al S-E Uso. Se confirma inferencia de la explicación causal.
		Mico																							
		Orbacus																							
	Uso 2ª Iteración	git																							Diferencias del S-E Uso respecto a utilizar los PCA como "caja negra": Al controlar la construcción del ejecutable de los PCA, gradualmente incrementa la independencia. Al concluir con éxito la actividad Aplicar, se vislumbra una alternativa de participación en la red de innovación que desarrolla git. Al concluir con éxito la actividad Aplicar, se confirma que se accede al S-E Uso. Se confirma inferencia de la explicación causal.
		JacORB: de CVS a git																							Prospectiva: Al adecuar la construcción del ejecutable de los PCA, a los procesos de la organización gradualmente incrementará la interdependencia (con la red de innovación). Al concluir con éxito la actividad Aplicar, se confirmará que se accede al S-E Adecuación. Diferencias del S-E Adecuación respecto a usar los PCA como "caja negra".
	Adecuación	JacORB: de Ant a Gradle																							
		JacORB: Monitoreo de JVM, ORB, POA, POA Mgr. y Servants.																							
		Modularizar via "ClassLoaders".																							
	Contribución	JacORB: Control de versiones distribuido.																							Prospectiva: Por definir.
		JacORB: Modularización.																							Prospectiva: Por definir.

Figura 7.8: Resultados de la tercera iteración en el cruce del MACOSC con el IASC (Fuente: Elaboración propia).

# Conclusiones

A partir de argumentar la pertinencia de la propuesta de una nueva concepción del conocimiento organizacional y su consecuente modelo de administración del conocimiento (MACOSC), que contempla la necesidad de estas organizaciones, y de argumentar la pertinencia de la propuesta de un diseño de investigación-acción de sistemas complejos (IASC), que es utilizado para realizar el estudio del complejo empírico observado con la nueva concepción, podemos concluir que:

- Concebir al conocimiento organizacional como un sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización, sistema complejo que emerge cuando el sistema de acciones en ejecución cumple las condiciones e incluye las causalidades descritas, permite que en la práctica administrativa dicho conocimiento organizacional sea observable en la dimensión colectiva en el nivel reiterado de eficacia y eficiencia del sistema de acciones.
- La creación y diseño del método investigación-acción de sistemas complejos (IASC), permite abordar la investigación de la pertinencia del modelo de administración del conocimiento (MACOSC), que se implanta en la organización sujeto de estudio.

A partir del incipiente cúmulo de evidencia que el estudio empírico ofrece, con los resultados de intervenir con el MACOSC-IASC una de estas organizaciones en México, y a partir de las explicaciones del funcionamiento del sistema complejo, podemos concluir que:

- Al implantar el MACOSC en la organización sujeto de estudio, ésta accede a los subprocesos-estadios de **Uso** y **Adecuación**. Se corrobora la hipótesis en este caso.

La consecuencia de corroborar la hipótesis, para los subprocesos-estadios de **Uso** y **Adecuación**, es que las acciones propuestas en el modelo MACOSC propiciaron la consecución en grado elevado del objetivo organizacional de adecuar los programas de código abierto. Lo anterior tiene como consecuencia que el modelo fue pertinente para la administración del conocimiento en la organización sujeto de estudio.

# Bibliografía

- Aime, F., Johnson, S., Ridge, J., Hill, A. *The Routine may be Stable but the Advantage is Not: Competitive Implications of Key Employee Mobility*. Strategic Management Journal; Wiley J. Published Online: November 12, 2009 (DOI: 10.1002/smj.809); Print Issue Date: January 2010. USA. (2009 y 2010).
- Ainamo, A. (2008). *Global Projects as New Organizational Form*. en Wankel, C. (ed.): *21st Century Management A Reference Handbook, Volume 1*: 482-489. SAGE Publications Inc., California, USA.
- Ali, N., Beecham, S., Mistrik, I. (2010). *Architectural Knowledge Management in Global Software Development: A Review*. ICGSE '10, Proceedings of the 2010 5th IEEE International Conference on Global Software Engineering, pp. 347-352, IEEE Computer Society Washington, DC, USA.
- Amozurrutia, J. A. *Complejidad y Ciencias Sociales: un modelo adaptativo para la investigación interdisciplinaria*. CEIICH, UNAM, México. (2011).
- Avison, D., Francis, L., Myers, M., Nielsen, P. A. *Action Research*. Communications of the ACM, Vol. 42 No. 1, Páginas 94-97 (1999).
- Avison, D., Baskerville, R., Myers, M., Wood-Harper, T. *IS Action Research: Can We Serve Two Masters?*. ICIS 99, Proceedings of the 20th international conference on Information Systems, AIS Atlanta, GA, USA, Páginas 582-585 (1999).
- Baskerville, R. *Investigating Information Systems with Action Research*. Communications of AIS, Vol. 2, article 19 (1999).
- Beck, K. *et al.* (2001). *Manifiesto por el Desarrollo Ágil de Software*. Disponible en: <http://agilemanifesto.org/iso/es/>. (Se accedió al sitio 02-enero-2013).

- Beyerlein, M., Kennedy, F. (2008). *Managing Intangible Capital* en Wankel, C. (ed.): *21st Century Management A Reference Handbook, Volume 2*: 401-410. SAGE Publications Inc., California, USA.
- Bjørnson, F. O., Dingsøyr T. *Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used*. Information and Software Technology, Volume 50, Issue 11, Páginas 1055-1068. (October 2008).
- Black Duck Software, Inc. (2013). *Debian GNU/Linux Project Summary*. Disponible en: <http://www.ohloh.net/p/debian>. (Se accedió al sitio 02-enero-2013).
- Brown, A., Wilson, G. (Editors) (2011). *The Architecture of Open Source Applications. Volume I: Elegance, Evolution, and a Few Fearless Hacks*. Disponible en: <http://www.aosabook.org/en/index.html>. (Se accedió al sitio 02-enero-2013).
- Brown, A., Wilson, G. (Editors) (2012). *The Architecture of Open Source Applications. Volume II: Structure, Scale, and a Few More Fearless Hacks*. Disponible en: <http://www.aosabook.org/en/index.html>. (Se accedió al sitio 02-enero-2013).
- Carrillo, L. P. *Sociedad del Conocimiento, Academia, administración, complejidad y tecnología*. UNAM - Facultad de Ciencias Políticas y Sociales; SITESA, México. (2008).
- Carrillo, L. P. *Knowledge Management Process and Technology Capacity in a Social Sciences Network Research*. The Journal of Knowledge Economy & Knowledge Management (JKEM), Volume: V, pp. 153-164. (2010).
- Carrillo, L. P. *Solving Social Research Problems in Latin America and the Caribbean: Creating Technological and Research Capabilities with Action Research*. International Journal of Social Science and Humanity, Vol. 1, No. 3, September. (2011).
- Carrillo, L. P. *Evaluación del desempeño de las organizaciones del sector social desde la perspectiva de la administración*. Proyecto de estancia posdoctoral, CEIICH, UNAM, México. (2012).
- Chiavenato, I. *Introducción a la Teoría General de la Administración*. Quinta Edición, McGraw-Hill Interamericana, México. (2000).
- Crawford, B., Castro, C., Monfroy, E. *Knowledge Management in the Development of Optimization Algorithms*. En: M.J. Smith, G. Salvendy (Eds.): *Human Interface, Part I, HCII 2007, LNCS 4557*, pp. 604–612. Springer-Verlag Berlin Heidelberg (2007).

- Davenport, T., Prusak L. *Excerpt of Working Knowledge: How organizations Manage What They Know*. Ubiquity, Volume 2000 Issue August, Article No. 6, ACM, N.Y., USA. <http://ubiquity.acm.org/article.cfm?id=348775>. (August 1 - August 31, 2000).
- de Gortari, R. *La nueva administración del conocimiento. El papel central de las áreas tecnológicas en las grandes empresas mexicanas*. Capítulo XIII, p. 365-383. en Dutrénit, G. Jasso, J., Villavicencio, D. (coordinadores): *Globalización, acumulación de capacidades e innovación: Los desafíos para las empresas, localidades y países*. FCE, México. (2007).
- Delgado, G. *Guerra por lo invisible. Negocio, implicaciones y riesgos de la nanotecnología*. UNAM, Centro de Investigaciones Interdisciplinarias en Ciencias y Humanidades, México. (2008).
- Descartes, R. *Las Meditaciones metafísicas*. Madrid, Gredos (traducción E. López y M. Graña) (1997, 1641, 1642).
- Desouza K. *Barriers to Effective Use of Knowledge Management Systems in Software Engineering*. Communications of the ACM, Vol. 46, Issue 1, ACM, N.Y., USA. (2003).
- Détienne, F., Burkhardt J-M., Barcellini, F. *Open source software communities: current issues*. Psychology of Programming Interest Group, Newsletter 2006-09 <http://www.ppig.org/newsletters/2006-09/2-overview-oss.pdf>. (September 2006).
- Devin, L., Austin, R. (2008). *Artistic Methods and Business Disorganization*. en Wankel, C. (ed.): *21st Century Management A Reference Handbook, Volume 1*: 490-499. SAGE Publications Inc., California, USA.
- Drucker, P. (1974). *Management*. Harper and Row Publishers, New York, USA.
- Easterbrook *et al.* (2008). *Selecting Empirical Methods for Software Engineering Research*. en Shull *et al.* (eds.): *Guide to Advanced Empirical Software Engineering*: 285-311. Springer-Verlag, London, UK
- Econsultancy.com, Ltd. *Android takes 52.5% of smartphone market*. Econsultancy, Ltd., (Nov 2011) <http://econsultancy.com/uk/blog/8279-android-doubles-market-share-to-take-52-5-of-smartphone-market>. (Se accedió al sitio el 24-Ene-2012).
- Federman, M. (2006). *The Penguinist Discourse: A Critical Application of Open Source Software Project Management to Organization Development*. Organization Development Journal, Vol. 24, No. 2

- Flichy, P. (2008). *Internet, un outil de la démocratie ?*. La Vie Des Ideas. Disponible en: <http://www.laviedesidees.fr/Internet-un-outil-de-la-democratie.html?lang=fr>. (Se accedió al sitio 02-enero-2013).
- Free Software Foundation Inc. *La Definición de Software Libre*. Free Software Foundation Inc. <http://www.gnu.org/philosophy/free-sw.es.html>. (Se accedió al sitio el 30-Abr-2010).
- Free Software Foundation Inc. *GNU General Public License*. Free Software Foundation Inc. <http://www.gnu.org/licenses/gpl.txt>. (Se accedió al sitio el 30-Abr-2010).
- García, R. *El conocimiento en construcción: de las formulaciones de Jean Piaget a la teoría de sistemas complejos*. Barcelona, Gedisa (2000).
- García, R. *Sistemas complejos: Conceptos, método y fundamentación epistemológica de la investigación interdisciplinaria*. Barcelona, Gedisa (2008).
- Geeknet, Inc. *Debian Source Code Analytics: Lines of Code By Language*. Geeknet, Inc. <https://www.ohloh.net/p/debian/analyses/latest>. (Se accedió al sitio el 30-Abr-2010).
- Gettier, E. *Is Justified True Belief Knowledge?*. *Analysis*, 23: 121–123. <http://www.ditext.com/gettier/gettier.html>. (1963).
- Goldman, R., Gabriel, R. P. (2005). *Innovation Happens Elsewhere: Open Source as Business Strategy*. Morgan Kaufmann Publishers, Elsevier, CA, USA. Disponible en: <http://www.dreamsongs.com/IHE/>. (Se accedió al sitio 02-enero-2013).
- González, C. E. *Commensurar la ciencia: observaciones tecnificadas y comparabilidad*. Tesis de Maestría, Posgrado en Filosofía de la Ciencia, UNAM, México. (2011).
- González, J., Amozurrutia, J., Maass, M. *Cibercultura e iniciación en la investigación*. Consejo Nacional para la Cultura y las Artes : Instituto Mexiquense de Cultura : UNAM, Centro de Investigaciones Interdisciplinarias en Ciencias y Humanidades, México. (2007).
- Griseri, P. *Management Knowledge - A critical view*. Palgrave, Great Britain (2002).
- Henning, M. *The Rise and Fall of CORBA*. *Communications of the ACM*, p53-57, Vol. 51, Num. 8, NY, USA. (2008)
- Hernández, R., Fernández, C., Baptista, P. *Metodología de la Investigación*. 3a. Ed., McGraw-Hill, México. (2003).

- Herrera, L. V., Ramírez, P. y May, G. (2012). *Organizaciones artísticas en México: Modelo e indicadores de capital intelectual*. Contaduría y Administración, 57-3, pp. 259-275, UNAM, México. Disponible en: <http://contaduriayadministracionunam.mx/articulo-2-266-32.html>. (Se accedió al sitio 02-enero-2013).
- Herrera, A. y Torres, J. A. (2007). *Falacias*. Editorial Torres Asociados, 2a. ed., D.F., México.
- Hume, D. *Tratado de la Naturaleza Humana: Ensayo para Introducir el Método del Razonamiento Humano en los Asuntos Morales*. Tecnos, Madrid. (1998, 1739).
- Hung, E. *The Nature of Science: Problems and Perspectives*. Wadsworth Publishing Company, CA, USA. (1997).
- IDC, Inc. *Worldwide Server Market Rebounds Sharply in Fourth Quarter as Demand for Blades and x86 Systems Leads the Way, According to IDC*. IDC, Inc. Documento número: prUS22224510, (Feb 2010), Research Press Release <http://www.idc.com/>. (Se accedió al sitio el 05-May-2010).
- IDC, Inc. *Worldwide Quarterly Mobile Phone Tracker*. IDC, Inc. , (14 febrero 2013), Research Press Release <http://www.idc.com/getdoc.jsp?containerId=prUS23946013#.UUJ75VHP5Ls>. (Se accedió al sitio el 14-febrero-2013).
- ISO/IEC Standard 29110-5-1-2:2011. (2011). *Software engineering - - Lifecycle profiles for Very Small Entities (VSEs) - - Part 5-1-2: Management and engineering guide: Generic profile group: Basic profile*. Disponible en: [http://standards.iso.org/ittf/PubliclyAvailableStandards/c051153\\_ISO\\_IEC\\_29110-5-1-2\\_2011.zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/c051153_ISO_IEC_29110-5-1-2_2011.zip). (Se accedió al sitio 02-enero-2013).
- ISO/IEC Standard 19506:2012(E). (2012). *Information technology - Architecture-Driven Modernization (ADM): Knowledge Discovery Meta-Model (KDM)*. Disponible en: [http://www.iso.org/iso/home/store/catalogue\\_tc/catalogue\\_detail.htm?csnumber=32625](http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=32625). (Se accedió al sitio 27-Jul-2012).
- Jennex, M. E. (2008). *Knowledge Management*. en Wankel, C. (ed.): *21st Century Management A Reference Handbook, Volume 2*: 336-347. SAGE Publications Inc., California, USA.
- Kant, I. *Crítica a la Razón Pura*. Alfaguara, Madrid, España. (1995).
- Kerlinger, F., Lee, H. *Investigación del Comportamiento: Métodos de Investigación en Ciencias Sociales*. 4a. Ed., (trad. Pineda, L., Mora, I.), (rev. Balbás, C., Vadillo, G.), McGraw-Hill, México. (2002).

- Khun, T. *La Estructura de las Revoluciones Científicas*. FCE, (trad. Solís, C.), Tercera edición (2006), primera reimpresión, México. (2007).
- Kock N.F. Jr., McQueen R. J., Scott, J. L. *Can action research be made more rigorous in a positivist sense? The contribution of an iterative approach*. *Journal of Systems and Information Technology*, Vol. 1 Iss: 1, pp. 1 - 23 (1997).
- Laudan, L. *La ciencia y el relativismo*. Alianza Editorial, Madrid (1993).
- Le Blanc, B. Ermine, J-L (2007). *A Shannon's Theory of Knowledge. Ser. on Innovation and Knowledge Management Series*. en Suliman Al-Hawamdeh (ed.), *Creating collaborative advantage through knowledge and innovation*: 51-67. World Scientific Pub. Disponible en: <http://books.google.com.mx/books?id=qWQCfXpec68C>.
- Lehaney, B., Clarke, S., Coakes, E., Jack, G. *Beyond Knowledge Management*. Idea Group Publishing, UK (2004).
- Madeiras dos Santos, S., Horta, G. *Action Research Use in Software Engineering: an Initial Survey*. ESEM 2009 Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement; IEEE Computer Society Washington, DC, USA. (2009).
- Maass, M., Amozurrutia, J., Almaguer, P., González, L., Meza, M. *Sociocibernética, cibercultur@ y sociedad*. CEIICH, UNAM, México. (2012).
- Maturana, H., Varela, F. *El Árbol del Conocimiento - Las bases biológicas del entendimiento humano*. Lumen - Editorial Universitaria, Argentina (2003).
- Marx, C. *Tesis sobre Feuerbach*. En C. Marx y F. Engels Obras Escogidas, pp. 24-26, Editorial Progreso, Moscú, URSS. (1975, 1845).
- MERITUM Project. (2001). *Measuring Intangibles to Understand and Improve Innovation Management: MERITUM - Intellectual capital guidelines for firms*. Universidad Autónoma de Madrid, Instituto Universitario de Administración de Empresas (IADE), España. Disponible en: [http://ec.europa.eu/research/social-sciences/projects/073\\_en.html](http://ec.europa.eu/research/social-sciences/projects/073_en.html). (Se accedió al sitio 02-enero-2013).
- Morado, R. *Compendio de Lógica*. Editorial Torres Asociados, D.F., México. (2009).
- Morales, M., Ventura, T., Oktaba, H., Torres, R. (2012) *From MoProSoft Level 2 to ISO/IEC 29110 Basic Profile: Bridging the Gap*. XV Ibero-American Conference on Software Engineering, CIbSE2012, Buenos Aires, Argentina. Disponible



en: [http://cibse.inf.puc-rio.br/CIBSEpapers/pdf\\_counter.lp?cibse=CIBSE12&file\\_name=paper\\_7.pdf](http://cibse.inf.puc-rio.br/CIBSEpapers/pdf_counter.lp?cibse=CIBSE12&file_name=paper_7.pdf). (Se accedió al sitio 02-enero-2013).

Morin, E. *Introducción al pensamiento complejo*. Primera reimpression, Editorial Gedisa Mexicana, D.F., México. (2004).

Morin, E. *Pensar la complejidad: Crisis y metamorfosis*. Escritos seleccionados, ed. Ana Sánchez, Universitat de València, España. (2010).

Nerur, S., Balijepally, V. *Theoretical reflections on agile development methodologies*. Communications of the ACM, Volume 50, Issue 3, N.Y., USA. (2007).

Nielsen T. M. (2008). *The Evolving Nature of Work Teams. Changing to Meet the Requirements of the Future*. en Wankel, C. (ed.): *21st Century Management A Reference Handbook, Volume 2*: 490-499. SAGE Publications Inc., California, USA.

Nonaka, I., Takeuchi, H. *The Knowledge-Creating Company - How Japanese Companies Create the Dynamics of Innovation*. Oxford University Press (1995).

Oktaba, H. (2006). *MoProSoft: A Software Process Model for Small Enterprises*. En: Proceedings of the First International Research Workshop for Process Improvement in Small Settings, 2005. (Ed. Garcia S., Graettinger C., Kost K.), SPECIAL REPORT CMU/SEI-2006-SR-001, pp. 93-101, Software Engineering Process Management, CarnegieMellon-SEI, Pittsburg, USA.

Oktaba, H. *et al.* (2007). *Software Process Improvement: The COMPETISOFT Project*. IEEE Computer, vol. 40, 10, pp. 21-28, USA.

Oktaba, H., Piattini, M., Pino, F.J., Orozco, M.J., Alquicira, C. *COMPETISOFT Mejora de Procesos Software para Pequeñas y Medianas Empresas y Proyectos*. Alfaomega Grupo Editor, México. (Enero 2009).

Olivé, L. *La ciencia y la tecnología en la sociedad del conocimiento: Ética, política y epistemología*. FCE, México (2008).

Olivé, L. *¿A quién pertenece el conocimiento? Poder y contrapoderes en el camino hacia las sociedades del conocimiento. En el libro: Sociedad del Conocimiento (Propuestas para una agenda conceptual), Suárez Rodolfo (Coordinador), págs. 89-108*. UNAM, Coordinación de Humanidades (Octubre 2009).

Object Management Group Inc. *CORBA Success Stories*. Object Management Group Inc. <http://www.corba.org/success.htm> (Se accedió al sitio 27-Jul-2012)

- Object Management Group Inc. *Architecture-Driven Modernization (ADM): Knowledge Discovery Meta-Model (KDM), v1.3*. Object Management Group Inc. <http://www.omg.org/spec/KDM/1.3/PDF/> (Se accedió al sitio 27-Jul-2012)
- Object Management Group Inc. *UPDATED CORBA 3.3 RTF report* Report of the CORBA/ZIOP Revision Task Force to the OMG Platform Technical Committee, 12 March 2012, <http://www.omg.org/cgi-bin/doc?ptc/12-03-15.pdf>, MA, USA. <http://www.omg.org/cgi-bin/doc?ptc/2012-03-15> (Se accedió al sitio 27-Jul-2012)
- Object Management Group Inc. *Asynchronous Method Invocation For CORBA Component Model (AMI4CCM)*. Version 1.0 Beta 1, Release Date April 2012, <http://www.omg.org/spec/AMI4CCM/1.0/Beta1>, MA, USA. <http://www.omg.org/spec/AMI4CCM/> (Se accedió al sitio 27-Jul-2012)
- Object Management Group Inc. *IDL To C++11 Language Mapping*. Version 1.0 Beta 1, Release Date April 2012, <http://www.omg.org/spec/C++11/1.0/Beta1>, MA, USA. <http://www.omg.org/spec/Cpp11/> (Se accedió al sitio 27-Jul-2012)
- Open Source Initiative Inc. (2013). *Open Source Licenses*. Open Source Initiative Inc. Disponible en: <http://www.opensource.org/licenses>. (Se accedió al sitio 02-enero-2013).
- Orozco, O. *Simple Enterprise Application Integration Framework*. Conferencia impartida en: IONA World 2001, Miami, FL, USA. (2001)
- Orozco, O. *Representación y Gestión del Conocimiento de los Programas de Código Abierto*. En Memorias del Primer Congreso de Alumnos de Posgrado de la UNAM, Área de Ciencias Sociales, Aporte al Conocimiento, poster 558, Coordinación de Estudios de Posgrado, UNAM. [http://www.posgrado.unam.mx/publicaciones/1congreso/03\\_2\\_CS.pdf](http://www.posgrado.unam.mx/publicaciones/1congreso/03_2_CS.pdf). (Octubre 2011).
- Orozco, O. y Carrillo, L. P. *La administración del conocimiento en las organizaciones que desarrollan sistemas de información: Análisis de la concepción de conocimiento*. En Memorias del XVII Congreso Internacional de Contaduría, Administración e Informática, Facultad de Contaduría y Administración, UNAM. <http://congreso.investiga.fca.unam.mx/docs/anteriores/xvii/docs/M03.pdf>. (Octubre 2012).
- Piaget, J., García, R. et al. *Hacia una lógica de significaciones*. Barcelona, Gedisa (1997).
- Platón *Teeteto*. Madrid, Gredos (Vol. 5 traducción, introducción y notas de Álvaro Vallejo Campos) (1988, c.a. 369 a.C.).

- Plato *Meno*. Plato in Twelve Volumes, Vol. 3 translated by W.R.M. Lamb. Cambridge, MA, Harvard University Press; London, William Heinemann Ltd. <http://www.perseus.tufts.edu/cgi-bin/ptext?lookup=Plat.+Charm.+153a&vers=original>. (1967, c.a. 386-382 a.C.).
- Polanyi, M. *Society, Economics and Philosophy - Selected Papers*. Transaction Publishers, New Brunswick, USA. (1997, 1961-1964).
- Polanyi, M. and Sen, A. (2009). *The tacit dimension* with a new foreword by Amartya Sen. The University of Chicago Press, USA.
- Poteete, A. R., Janssen, M. A. y Ostrom E. (2012). *Acción colectiva, bienes comunes y múltiples métodos en la práctica*. trad. Niles L. B., Merino, L. UNAM, CEIICH, CRIM, FCPS, FE, IIEc, IIS, PUMA; IASC, CIDE, Colsan, CONABIO, CCMS, FCE, UAM, México.
- Quine, W. V. (1984). *Methods of Logic*. 4th ed., Harvard University Press, MA, USA.
- Quintanilla, M. A. (2005). *Tecnología: Un enfoque filosófico y otros ensayos de filosofía de la tecnología*. Fondo de Cultura Económica, D.F., México.
- Reding, Viviane. *ICT research and telecoms: Europe opportunity to lead global competition*. European Policy Centre, Bruselas, Unión Europea. (Octubre 2008).
- Rivas, L., et al. *Efectos de la teoría de la complejidad en la gestión ambiental en México. Primera Edición*. Instituto Politécnico Nacional, Centro Mario Molina, México. (2009).
- Shannon, C. E. (1948). *A Mathematical Theory of Communication*. The Bell System Technical Journal, 27, 379-423, 623-656. Disponible, con correcciones, en: <http://cm.bell-labs.com/cm/ms/what/shannonday/shannon1948.pdf>. (Se accedió al sitio 01-febrero-2013).
- Solleiro et al. (coordinador) *Gestión del conocimiento en centros de investigación y desarrollo de México, Brasil y Chile*. Programa de Investigación sobre Economía del Conocimiento en América Latina y el Caribe IDRC-Flacso, FLACSO México, Centro Internacional de Investigaciones para el Desarrollo, México. (2009).
- Sotolongo, P. L. *La complejidad desde el sur*. Conferencia Magistral, XVII Congreso Internacional de Contaduría, Administración e Informática, UNAM, México. (2012).
- Sowa, J. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Brooks-Cole, CA, USA. (2000).

Suárez, S., Carreto, Ch. y Ruiz, E. F. (2012). *Modelo de administración del conocimiento aplicado a enseñanza-aprendizaje por medio de dispositivos móviles*. Revista Innovación Educativa, vol. 12 núm. 58, pp. 151-165, IPN, México.

TOP500.Org *Top 500 Supercomputer Sites*. TOP500.Org <http://www.top500.org/stats/list/34/osfam>. (Se accedió al sitio el 03-May-2010).

Tsetung, M. *Sobre la práctica: Sobre la relación entre el conocimiento y la práctica, entre el saber y el hacer*. En *Cinco Tesis Filosóficas de Mao Tsetung*, pp. 1-41, Ediciones en Lenguas Extranjeras, Pekin, China. (1975, 1937).

Villoro, L. *Creer, saber, conocer*. Decimoctava edición, Siglo veintiuno editores, México. (2008).

Wang, Y. *Software Engineering Foundations - A Software Science Perspective*. Auerbach Publications. (2008).

# Índice alfabético

- Acoplamiento estructural, 35
  - de tercer orden, 35
- Administración del conocimiento, 46
  - en las pequeñas organizaciones que desarrollan software, cuando pretenden adecuar software de código abierto, 46
  - modelo para la, 46
- Complejo empírico, 66
- Conocimiento organizacional
  - como sistema complejo de acciones eficaces, eficientes y reiteradas en ejecución por los diversos agentes involucrados en la organización, 44
  - como sistema complejo que emerge . . . , 45
  - observado como sistema complejo, 78
  - observado como sistema complejo, en la primera iteración, 81
  - observado como sistema complejo, en la segunda iteración, 86
  - observado como sistema complejo, en la tercera iteración, 90
- CORBA
  - en la industria, 73
  - definición, 73
- Diseño de la investigación-acción de sistemas complejos, 64
- Diseño mixto investigación-acción de sistemas complejos (IASC), 65
  - gráfica detallada del IASC, 69
- IASC como proceso iterativo incremental, 66
- Estudio empírico, 72
  - complejo empírico, 76
  - explicación causal de la hipótesis, 75
  - perspectiva general, 74
  - plan de acción, 71
  - plan de acción general de las iteraciones, 76
  - primera iteración, 80
  - segunda iteración, 85
  - tercera iteración, 89
  - unidad de análisis, 76
  - unidad de observación, 76
- Explicación causal de la hipótesis, 75
  - plan de acción, 71
  - plan de acción general de las iteraciones, 76
- Hipótesis, 53
- IASC: investigación-acción de sistemas complejos, 64
- Investigación-acción
  - problemas reportados, 65
- Investigación-acción de sistemas complejos, 64
- Investigación-acción de sistemas complejos (IASC)
  - Ubicación del método IASC, 70
- MACOSC en la práctica administrativa, 47
  - Adecuación**, 48
  - Contribución**, 50
  - Uso**, 47

Marco metodológico  
    complejo empírico, 66  
    objeto de estudio, 66  
    unidad de análisis, 66

Modelo para la administración del conocimiento, 46  
    MACOSC, 46  
    MACOSC en la práctica, 47

Object Management Group, 73

Objetivos de la investigación, 8, 74

OMG, 73

Pequeña organización que desarrolla software,  
    72  
    sujeto de estudio, 72

Pregunta de investigación, 8

Problema de investigación, 8

Problema práctico, 4

Redes sociales de innovación, 5

Sistema Complejo  
    como conocimiento organizacional, 44, 78  
    como conocimiento organizacional, en la primera iteración, 81  
    como conocimiento organizacional, en la segunda iteración, 86  
    como conocimiento organizacional, en la tercera iteración, 90

Sistema complejo  
    que emerge . . . , 52

Software de código abierto, 5



UNIVERSIDAD NACIONAL  
AVENIDA DE  
MEXICO

Programa de Posgrado en Ciencias de la  
Administración  
Oficio: PPCA/GA/2013

**Asunto:** Envío oficio de nombramiento de jurado de Doctorado.

**Coordinación**

**Dr. Isidro Ávila Martínez**  
**Director General de Administración Escolar**  
**de esta Universidad**  
**Presente**

At'n.: Dr. Manuel Enrique Vázquez Valdés  
Coordinador de la Unidad de Administración del Posgrado

Me permito hacer de su conocimiento, que el alumno **Octavio Orozco y Orozco**, presentará tesis dentro del Plan de **Doctorado en Ciencias de la Administración** toda vez que ha concluido el Plan de Estudios respectivo, por lo que el Subcomité de asuntos académicos y administrativos de Doctorado, tuvo a bien designar el siguiente jurado:

Dra. Graciela Bribiesca Correa	Presidente
Dr. José Antonio Amozurrutia de María y Campos	Vocal
Dra. Hanna Jadwiga Oktaba	Secretario
Dra. Lucía Patricia Carrillo Vélazquez	Suplente
Dr. Fernando Gamboa Rodríguez	Suplente

Por su atención le doy las gracias y aprovecho la oportunidad para enviarle un cordial saludo.

Atentamente  
"Por mi raza hablará el espíritu"  
Ciudad Universitaria, D.F., 27 de mayo de 2013.

**El Coordinador del Programa**

**Dr. Jorge Ríos Szalay**