

Caracterización de Sistemas Fiables basada en un modelo estándar de calidad*

(Characterization of Dependable Systems based on a standard quality model)

Isi S. Castillo

Universidad Nacional Experimental Sur del Lago, Lab. de Informática, Santa Bárbara, Venezuela, 5448
Universidad Central de Venezuela, Centro ISYS, Escuela de Computación, Caracas, Venezuela, 1041-A
castilloi@cantv.net

Rafael J. Caldera

Universidad de Oriente - Núcleo de Sucre, Prog. Lic. en Informática, Cumaná, Venezuela, 6101
Universidad Central de Venezuela, Centro ISYS, Escuela de Computación, Caracas, Venezuela, 1041-A
rjcaldera@sucre.udo.edu.ve

Francisca Losavio

Universidad Central de Venezuela, Centro ISYS, Escuela de Computación, Caracas, Venezuela, 1041-A
flosav@cantv.net

Alfredo Matteo

Universidad Central de Venezuela, Centro ISYS, Escuela de Computación, Caracas, Venezuela, 1041-A
almatteo@cantv.net

Resumen

La construcción de sistemas altamente confiables constituye aún un reto para la industria del software; las tecnologías y formalismos actuales no son suficientemente poderosos para asegurar la fiabilidad de los sistemas y para garantizar que éstos se adecuen a las complejidades técnicas propias del entorno donde se desempeñan. En consecuencia, es necesario manejar en los sistemas la ocurrencia de fallas, las faltas y los errores, así como los cambios en el entorno y en los requisitos de los usuarios, implicando un esfuerzo de desarrollo temprano y costoso. El objetivo principal de este trabajo es representar, a través de un modelo conceptual, los principales elementos y la terminología básica utilizada en el dominio de los sistemas fiables, así como las propiedades de calidad de este tipo de sistemas. Adicionalmente, en este trabajo se presenta una correspondencia de los requisitos necesarios para la fiabilidad de uso común en el dominio, con el modelo de calidad estándar ISO/IEC 9126-1. Este trabajo es una propuesta para sentar las bases para un entendimiento común y consenso sobre las definiciones, como punto de partida para una ontología en el dominio de los sistemas fiables.

Palabras Claves: sistemas fiables, fiabilidad, modelo conceptual, calidad de software, ISO/IEC 9126-1.

Abstract

The development of dependable systems is still a challenge for the software industry; existing technologies and formalisms are not sufficiently powerful to ensure systems dependability and to guarantee their compliance with the technical complexity of the operational environment. As a consequence, faults, failures and errors have to be handled into the systems, as well as changes in the environment and in the user requirements, implying an early and costly development effort. The main goal of this work is to define a conceptual model to represent the main elements and basic terminology used in the dependable systems domain, including the quality requirements for these systems. Additionally, this work presents a mapping of the requirements for dependability, commonly used in the domain, to the standard ISO/IEC 9126-1 quality model. This work aims to establish the basis of a common understanding and consensus on the definitions, as the starting point for an ontology in the dependable systems domain.

Keywords: dependable systems, dependability, conceptual model, software quality, ISO/IEC 9126-1.

* Esta investigación ha sido financiada por el Programa Alma Mater de la Oficina de Planificación del Sector Universitario (OPSU) y es parte de los resultados del proyecto MODABC No.PI 03-00-5821-2005, del Consejo de Desarrollo Científico y Humanístico (CDCH) de la Universidad Central de Venezuela.

1. INTRODUCCIÓN

A medida que el mundo de la computación avanza, surgen sistemas de software cada vez más complejos, tanto en estructura como en comportamiento, adaptados a los más variados usos y ambientes. Los sistemas financieros, médicos, de transporte, de servicios públicos, de seguridad y defensa, así como las demás infraestructuras básicas, dependen en la actualidad en gran medida de plataformas de información y comunicación; sin embargo, hoy en día estas plataformas y sus aplicaciones son cada vez más vulnerables a diversas situaciones no previstas o no tomadas en cuenta durante su concepción inicial.

El desarrollo de software durante los últimos años ha requerido cumplir con importantes necesidades, entre ellas garantizar la fiabilidad y seguridad de sus sistemas, mejorar la calidad de los procesos de desarrollo y los productos de software; garantizar la seguridad en las redes para evitar: ataques externos, denegación de servicios, pérdida de data confidencial, así como también garantizar sustancialmente la satisfacción de los usuarios.

El enfoque en la construcción de sistemas de software complejos está cambiando desde el esquema tradicional orientado a los productos, hacia el esquema de los sistemas orientados a servicios; surgen nuevas aplicaciones sometidas a constantes cambios, cuyos límites son muy dinámicos. La construcción de sistemas altamente confiables en estos contextos se ha convertido en un gran reto para la industria del software. En la actualidad se hacen necesarias mejores tecnologías y formalismos para especificar y manejar la fiabilidad. Estas deben ser más acordes con las complejidades técnicas propias de este tipo de sistemas, los cuales son propensos a la ocurrencia de fallas, faltas y errores, cambios en el entorno y en los requisitos de los usuarios.

Son múltiples los aspectos a considerar cuando se desarrollan los sistemas de alta disponibilidad que rodean nuestro entorno; en caso de producirse una falla en un sistema, o incluso en un único componente, las consecuencias pueden ser desastrosas si no se han establecidos mecanismos para manejar adecuadamente dicha falla. Esta situación ha exigido de la industria del software el desarrollo de sistemas que incorporen y cumplan con servicios que van más allá del alcance de sus funcionalidades principales; es decir, demandan el cumplimiento de requisitos importantes de fiabilidad, a los cuales el software debe responder correctamente, de acuerdo con su objetivo funcional establecido, como por ejemplo las transacciones en un contexto de “e-banking”.

Durante el desarrollo de los sistemas de software, resulta de gran importancia tener un conocimiento claro y preciso de los principales elementos conceptuales asociados al dominio del sistema, con el fin de manejar un lenguaje común. En muchas ocasiones los miembros de los equipos de investigación y desarrollo experimentan serias deficiencias en el manejo de los términos relacionados con el producto de software que deben obtener; esto puede dificultar la adecuada implementación de los requisitos exigidos por el producto. Esta debilidad ha despertado el interés de las comunidades de investigadores por buscar un consenso en los términos y/o acepciones de los elementos que caracterizan cada tipo de sistema. En particular, dicha preocupación ha llevado a la construcción de ontologías en dominios específicos. Dentro del contexto de este artículo, un ejemplo claro de este interés se ha evidenciado en los intentos llevados a cabo en [2], [3], [9], [10], [12] para definir una terminología estándar dentro del dominio de los sistemas fiables.

El objetivo principal de este trabajo es representar a través de un modelo conceptual los elementos y la terminología básica utilizada en el dominio de los sistemas fiables, así como las propiedades de calidad específicas de este tipo de sistemas; adicionalmente, se presenta un refinamiento del modelo anterior utilizando el estándar de calidad ISO/IEC 9126-1. Este trabajo es una propuesta para sentar las bases para un entendimiento común y consenso sobre las definiciones utilizadas en la comunidad como parte inicial de una ontología en el área de los sistemas fiables o computación fiable.

Este artículo está estructurado en 6 secciones. Luego de la introducción, en la sección 2 se presentan y definen los principales términos y conceptos asociados al dominio de los sistemas fiables. La sección 3 introduce y define brevemente el modelo de calidad del estándar ISO/IEC 9126-1. La sección 4 presenta y describe el modelo conceptual propuesto para los sistemas fiables según la terminología utilizada actualmente en la comunidad internacional. En la sección 5 se presenta el refinamiento del modelo propuesto utilizando el modelo de calidad ISO/IEC 9126-1, para especificar la fiabilidad en estos sistemas. Finalmente en la sección 6 se presentan las conclusiones sobre el trabajo realizado.

2. SISTEMAS FIABLES

En líneas generales, un sistema de software [3], [10] es una entidad que interactúa con otras entidades o sistemas (hardware, software o humanos), cumpliendo con una o más funciones (funcionalidad del sistema) descritas por su especificación funcional (requisitos funcionales); tal funcionalidad es implementada por un comportamiento especificado mediante un conjunto de estados. Desde el punto de vista estructural, un sistema está compuesto de un conjunto de componentes que interactúan entre sí; un sistema cumple su función entregando un servicio, este servicio entregado por el sistema es el comportamiento percibido por sus usuarios.

Son diversas las definiciones que se han establecido para describir en particular a los sistemas fiables o sistemas de computación fiable; en líneas generales los sistemas fiables o “dependable systems”, han sido definidos como todos aquellos sistemas capaces de entregar correctamente un servicio que ha sido previamente confiado o acordado; estos sistemas se fundamentan en el concepto de fiabilidad o “dependability” como propiedad principal [1], [3], [9]. Según [3], [9] la fiabilidad se define como la habilidad de un sistema para evitar frecuentes y severas faltas de un servicio, es un concepto integrado, que incluye los atributos o aspectos de calidad: Disponibilidad (“Availability”), Confiabilidad (“Reliability”), Resguardo (“Safety”), Integridad (“Integrity”), Confidencialidad (“Confidentiality”) y Mantenibilidad (“Maintainability”). La asociación entre disponibilidad, confidencialidad e integridad, conlleva a la definición de la propiedad denominada Seguridad (“Security”), como puede visualizarse en la figura 1 [3], [10]. Otra definición más sencilla para la fiabilidad presentada en [12] es “cualidad o característica de ser fiable”, donde el adjetivo “fiable” es atribuido a aquel sistema cuyas faltas son consideradas suficientemente escasas o insignificantes.



Figura 1: Árbol de Fiabilidad [3].

Del árbol de fiabilidad, se define la disponibilidad como la predisposición para obtener y/o proveer un servicio correcto, la confiabilidad como la continuidad de un servicio correcto, el resguardo como la ausencia de consecuencias catastróficas sobre el o los usuarios y el ambiente, la confidencialidad como la habilidad para mantener en reserva la información, la integridad como la ausencia de alteraciones inadecuadas del sistema y la mantenibilidad como la habilidad para experimentar modificaciones y reparaciones; finalmente, la seguridad se define como la ausencia de acceso no autorizado para el manejo del estado del sistema [3],[10].

En [1], [3], [10] se indica además que la fiabilidad en un sistema es afectada por una serie de amenazas denominadas fallas (“faults”), errores (“errors”) y faltas (“failures”). La falta en un sistema es un evento que ocurre cuando el servicio entregado se desvía del servicio esperado o servicio correcto; un error es aquella parte del estado del sistema que pueda causar una falta de servicio y una falla es definida como la causa hipotética de un error. Un error también se define como la parte total del estado del sistema que conlleva a faltas subsecuentes de servicio. Las diversas formas en la que un sistema puede dejar de proporcionar un servicio se denominan modos de falta (“failures modes”) [10].

3. EL MODELO DE CALIDAD ESTANDAR ISO/IEC 9126-1

Los productos de software se construyen o desarrollan para ser utilizados cumpliendo con necesidades o funciones específicas, que son requeridas por el usuario. Su calidad está determinada por la medida con la cual se cumplen estas funciones. Por tanto, el principal objetivo de un producto de software es satisfacer una necesidad (o varias) de un usuario y por consiguiente, ofrecerle algún beneficio por su utilización. Un control sobre la calidad del sistema (que se cumplan los objetivos establecidos durante el desarrollo y en el producto final) permite garantizar el cumplimiento de la funcionalidad global del sistema. La especificación de la calidad de un sistema de software debe ser detallada y

precisa; una manera de formalizarla es mediante un modelo de calidad, el cual permite definir los requisitos de calidad del producto que influyen a la hora de medir su calidad [4]. Según ISO/IEC 9126-1 [6], un modelo de calidad es definido como un conjunto de características de calidad y sus correspondientes relaciones, las cuales proporcionan un marco de trabajo para especificar los requisitos de calidad y evaluar la calidad de un producto de software; estas características (de alto nivel) son refinadas en subcaracterísticas, formando una jerarquía de múltiples niveles, las cuales a su vez son descritas por un conjunto de atributos. Siguiendo la terminología de ISO/IEC 9126-1, se entiende por característica de calidad de un producto de software a un conjunto de propiedades mediante las cuales se evalúa y describe su calidad. Un atributo se define como una propiedad de calidad medible, a la que puede asignarse una métrica, entendiéndose por métrica un método o procedimiento que examina un componente y produce un dato simple o valor [6], [11].

En particular, el estándar ISO/IEC 9126-1 [6], [7] describe un modelo para la calidad del producto, basado en tres vistas diferentes: calidad interna, calidad externa y calidad en uso (fig. 2 y 3). La calidad interna trata las propiedades del producto de software que normalmente están disponibles durante el desarrollo, está relacionada a las propiedades estáticas del software. La calidad externa trata las propiedades relacionadas con la ejecución del software en un ambiente de prueba y finalmente la calidad en uso está relacionada a la aplicación del software en su ambiente operacional definitivo, durante la ejecución de tareas específicas llevadas a cabo por sus usuarios finales. Así, la calidad interna tiene impacto sobre la calidad externa y esta última sobre la calidad en uso [7].

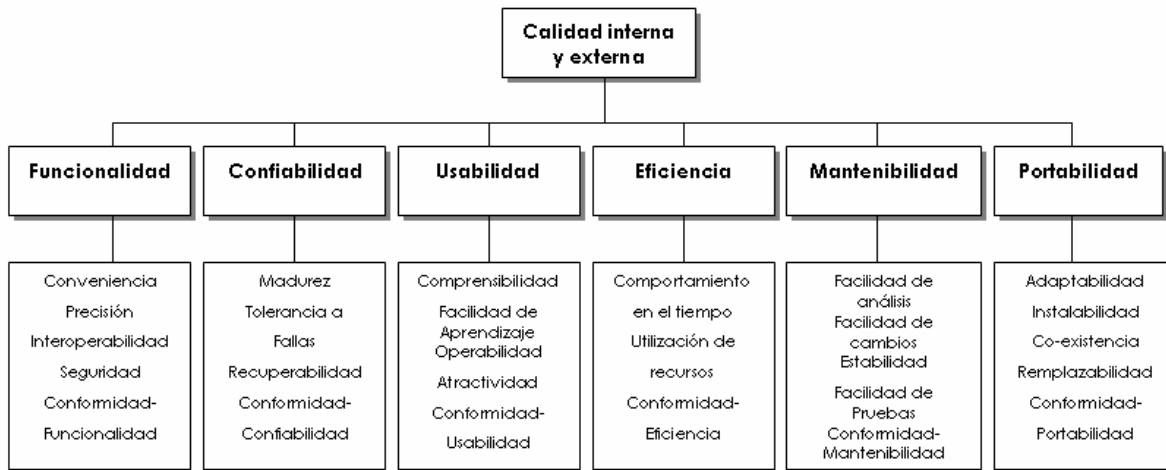


Figura 2: Modelo para la calidad interna y externa según ISO 9126-1 [6].

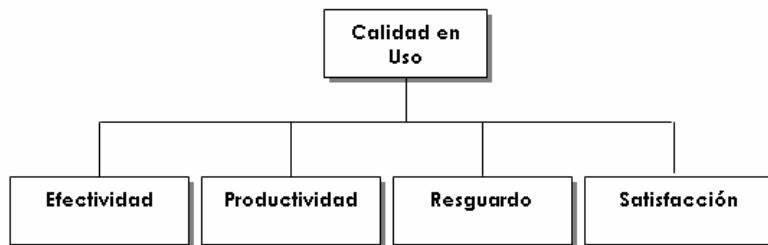


Figura 3: Modelo para la calidad en uso según ISO 9126-1 [6].

En el contexto particular de los sistemas fiables, la calidad es un factor determinante, el cual hay que garantizar necesariamente y viene dada de manera implícita por su propiedad principal la fiabilidad, la cual implica disponibilidad, confiabilidad, resguardo, integridad, confidencialidad y mantenibilidad según se indicó en la sección 2. A continuación se propone un modelo conceptual para este contexto y luego se establece una correspondencia entre estas propiedades y el marco de trabajo proporcionado por el estándar ISO/IEC 9126-1.

4. MODELO CONCEPTUAL PARA LOS SISTEMAS FIABLES

Esta sección presenta el modelo conceptual propuesto para el dominio de los sistemas fiables, al cual denominaremos DeSCMO (Dependable Systems Conceptual Model), expresado en UML [8]; en el mismo se refleja la terminología y los conceptos básicos comunes utilizados en consenso por la comunidad internacional. Este modelo pretende facilitar el entendimiento de los principales elementos asociados al dominio de estos sistemas. El hecho de expresarlo en UML facilita su reutilización en otros contextos, como por ejemplo bajo el enfoque MDA (“Model Driven Architecture”).

Hasta ahora un sistema ha sido implícitamente considerado como un todo, una entidad humana o física, de hardware y software que interactúa con otras entidades; cuyo comportamiento es percibido externamente por su entorno; es una entidad que cumple con una o más funciones para así entregar uno o más servicios a sus correspondientes usuarios [3], [10] (ver figura 4). El servicio entregado por un sistema se define como el comportamiento percibido por sus usuarios, es una secuencia de estados externos del sistema, un servicio puede ser correcto o incorrecto; un servicio es correcto cuando implementa adecuadamente la función del sistema [2], se dice que un servicio es incorrecto cuando el comportamiento del sistema no corresponde con la función esperada por los usuarios. Una función representa una parte de la funcionalidad de un sistema y es descrita mediante los requisitos del sistema de software (funcionales y no funcionales), los cuales son derivados de las necesidades o requisitos de la organización en la cual operará el sistema. Los requisitos funcionales especifican la funcionalidad del sistema de software; los requisitos no funcionales especifican las condiciones o restricciones que un sistema debe satisfacer, controlan la especificación, implementación y ejecución de los requisitos funcionales [5].

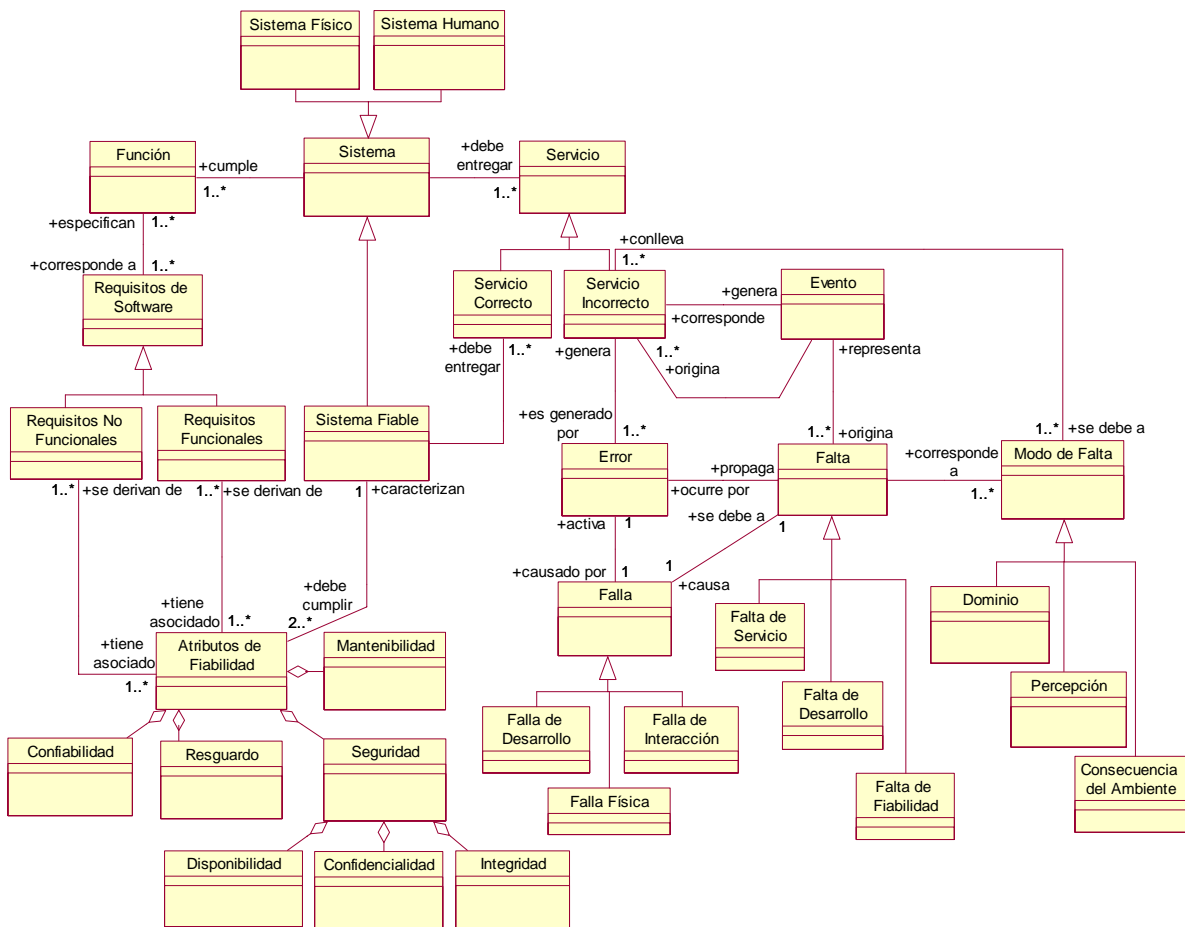


Figura 4: Modelo Conceptual UML para los Sistemas Fiables.

Un sistema fiable es un tipo particular de sistema que debe cumplir con una o más funciones y con determinados aspectos de calidad denominados atributos de fiabilidad [3][10]. Los atributos de fiabilidad representan el conjunto de propiedades esperadas de un sistema fiable, permiten especificar la fiabilidad y por ende, de manera implícita, determinan la calidad¹ global de los sistemas fiables. Los atributos de fiabilidad son derivados de los requisitos del software (funcionales y no funcionales) y se denominan confiabilidad, resguardo, mantenibilidad, y seguridad (representada por la funcionalidad, la confidencialidad y la integridad). Un sistema debe cumplir obligatoriamente con uno ó más requisitos de fiabilidad al mismo tiempo, pero no necesariamente con todos [3].

Un sistema fiable debe caracterizarse por entregar uno o más servicios correctos, un sistema fiable debe evitar presentar servicios incorrectos los cuales se originan por la influencia de ciertas amenazas indicadas en la sección 2: las fallas, los errores y las faltas [1],[3],[10]. Un servicio incorrecto es generado por uno o más errores, un error que afecta a un servicio es una indicación que ocurre o ha ocurrido una falta, un error es una desviación del servicio correcto y es causado por una falla. Una falta es una indicación que uno o más estados externos del sistema se desvían del estado de servicio correcto. Las faltas pueden ser de servicio, de desarrollo y de fiabilidad, las faltas de servicio se definen como un evento que ocurre cuando el servicio entregado por el sistema se desvía del servicio correcto. Las faltas de desarrollo corresponden a aquellas faltas introducidas al sistema durante su desarrollo, son ocasionadas por el equipo humano de desarrolladores y por las herramientas de desarrollo; estas faltas causan que el proceso de desarrollo sea terminado antes de que el sistema sea aceptado para su uso y puesto en servicio [3]. Las faltas de fiabilidad ocurren cuando un sistema experimenta faltas de servicio severas y frecuentes. Las diferentes formas en la que un sistema puede faltar se conocen como modos de falta, estos modos pueden ser de dominio, percepción y por consecuencias del ambiente [10]. El modo de falta de dominio se produce cuando el valor del servicio entregado y el tiempo que éste toma ya no se ajusta a la función del sistema; el modo de falta de percepción se distingue cuando un sistema tiene usuarios diversos y éstos pueden o no tener la misma percepción de una falta dada. El modo de falta por consecuencias del ambiente, hace referencia a los diferentes niveles de faltas generados por la incidencia de diferentes factores provenientes del ambiente o entorno del sistema. Por otra parte, como se indicó en la sección 2 la falla es toda causa (evento, acción o circunstancia) que puede provocar un error, una falla puede ser de desarrollo, física y de interacción. Se dice que una falta en un sistema puede causar una falla en otro sistema. Las fallas de desarrollo se originan durante el desarrollo del sistema, las fallas físicas se refieren a todas aquellas causas que afectan el hardware y las fallas de interacción se refieren a todas las causadas por fallas o interacciones externas.

5. LA FIABILIDAD Y EL ESTÁNDAR ISO/IEC 9126-1

Esta sección presenta ahora un refinamiento del modelo DeSCMO propuesto en la sección anterior, utilizando el modelo de calidad estándar ISO/IEC 9126-1 [6]. En este nuevo modelo se presentan los atributos o propiedades de fiabilidad utilizados por la comunidad internacional de investigadores desde la perspectiva propuesta en [6], es decir utilizando las vistas internas, externas y en uso de calidad. A pesar de que el marco de trabajo propuesto por ISO/IEC 9126-1 no contempla en su definición todas las propiedades de fiabilidad representadas en el modelo anterior y utilizadas por la comunidad de investigadores, el estándar permite la definición de nuevas subcaracterísticas dependiendo del contexto donde se aplique el modelo de calidad y facilitando por tanto su adaptación.

En el nuevo modelo propuesto (ver figura 5) y de acuerdo al estándar, se hace una correspondencia entre los términos “atributos” y “propiedades”, de manera que un sistema fiable debe cumplir con propiedades de fiabilidad las cuales son especificadas como características y subcaracterísticas del modelo de calidad estándar ISO/IEC 9126-1, adaptado para estos sistemas. Como puede verse en la figura 5, se ha establecido una correspondencia entre una propiedad de fiabilidad con una o más de las características y/o subcaracterísticas de calidad propuestas por el estándar ISO/IEC 9126-1. Este modelo de calidad para los sistemas fiables tiene dos perspectivas, la primera corresponde al modelo de calidad en uso el cual ha sido representado solamente por la característica de calidad Resguardo; la segunda corresponde al modelo de calidad interna y externa descrito por las características Confiabilidad, Funcionalidad y Mantenibilidad.

¹ Se entiende por “calidad” el conjunto de rasgos y características que dan a un producto o servicio su habilidad para satisfacer necesidades definidas o implícitas [6].

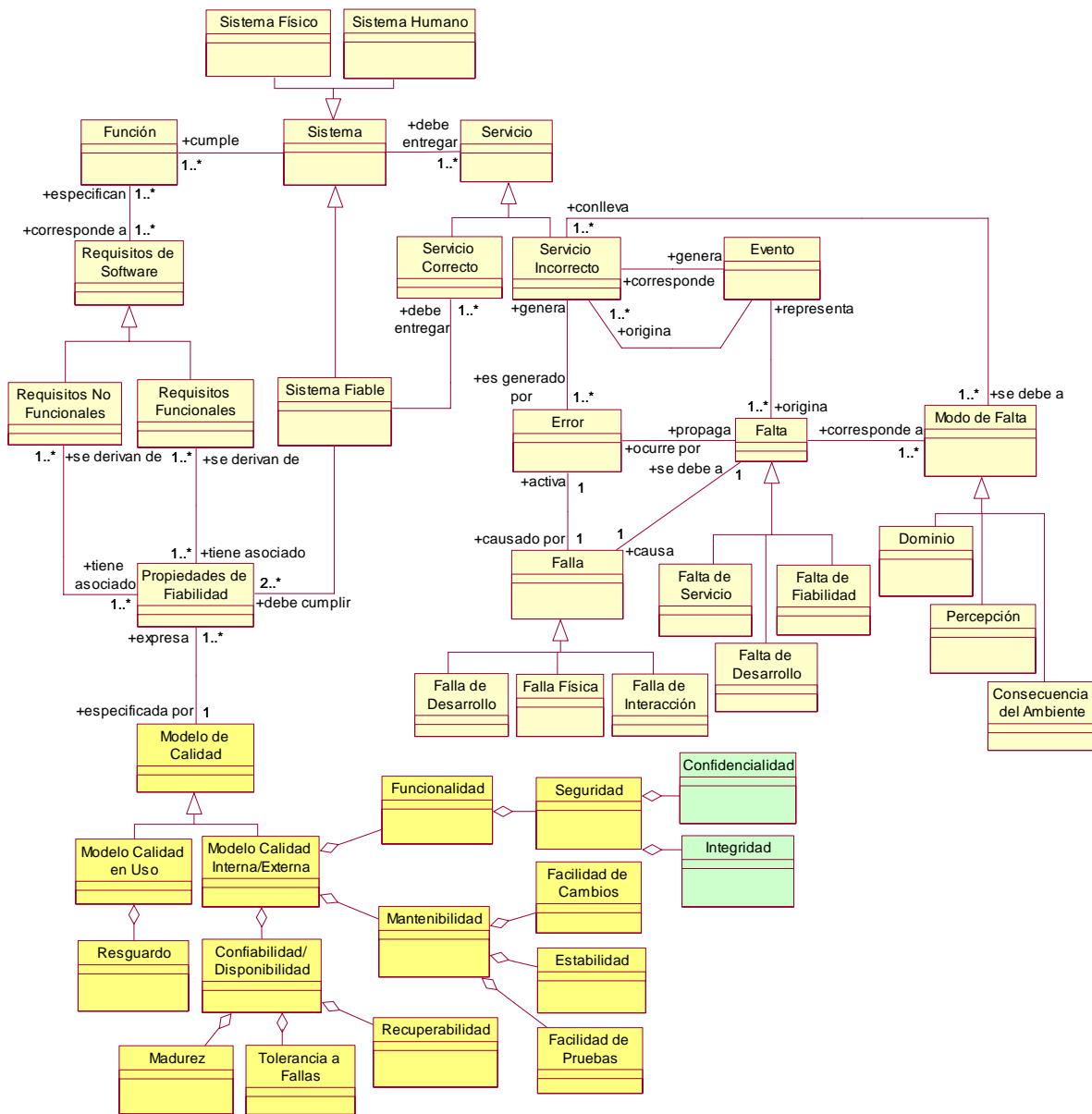


Figura 5: Refinamiento de DeSCMO utilizando el estándar ISO/IEC 9126-1.

En particular para el modelo de calidad en uso, se redefine la característica Resguardo desde la perspectiva de los sistemas fiables como la capacidad del sistema de software para alcanzar niveles aceptables de riesgos a daño sobre los usuarios, el ambiente y otros sistemas. En el modelo de calidad interna/externa se redefine a la Confiabilidad como la capacidad del sistema de software para mantener un nivel específico de desempeño y continuidad de un servicio correcto, cuando el sistema es utilizado en condiciones específicas; la confiabilidad implica madurez (capacidad del sistema para evitar faltas como resultado de fallas en el software), tolerancia a fallas (capacidad del sistema para mantener un nivel específico de desempeño en casos de fallas o trasgresiones de interfaces específicas) y recuperabilidad (capacidad del sistema para reestablecer un nivel específico de desempeño y recuperar información en caso de una falta). La Mantenibilidad se redefine como la capacidad del sistema para experimentar modificaciones; la mantenibilidad implica Facilidad de Cambios (capacidad del sistema para permitir la implementación de modificaciones específicas), Estabilidad (capacidad del sistema para evitar efectos inesperados por las modificaciones) y Facilidad de Pruebas (capacidad que permite al sistema ser validado). La Seguridad en el refinamiento efectuado al

modelo conceptual DeSCMO utilizando el marco de ISO/IEC 9126-1, viene dado por la incorporación de la característica Funcionalidad redefinida como la capacidad del sistema para proveer las funciones que permitan cumplir con necesidades implícitas y declaradas cuando el sistema es utilizado en condiciones específicas. La funcionalidad implica por tanto seguridad y define esta última como la capacidad del sistema para proteger la información y los datos de personas no autorizadas; la seguridad a su vez implica confidencialidad (capacidad del sistema para denegar el acceso a personas no autorizadas), integridad (capacidad del sistema para evitar modificaciones no autorizadas) y disponibilidad (capacidad del sistema para garantizar la continuidad de un servicio correcto). En el estándar ISO/IEC 9126-1 no se encuentran per se definidas confidencialidad e integridad, pero el marco de trabajo permite su definición como subcaracterísticas de seguridad; tampoco la característica disponibilidad se encuentra definida de manera explícita, sin embargo en [6] se señala que es una combinación de las subcaracterísticas madurez, tolerancia a fallas y recuperabilidad y por tanto también viene dada en nuestro modelo por la confiabilidad. La tabla 1 resume la correspondencia descrita, efectuada a las propiedades de fiabilidad utilizando el estándar ISO/IEC 9126-1.

Tabla 1: Correspondencia de las propiedades de Fiabilidad utilizando el estándar ISO/IEC 9126-1.

<i>Propiedades de Fiabilidad según la comunidad de investigadores</i>	<i>Propiedades de Fiabilidad adaptadas según ISO/IEC 9126-1</i>
Confiabilidad: continuidad de un servicio correcto.	Confiabilidad: capacidad para mantener un nivel específico de desempeño y continuidad de un servicio correcto. La confiabilidad implica: <ul style="list-style-type: none"> ▪ <i>Madurez, Tolerancia a fallas y Recuperabilidad (disponibilidad)</i>
Resguardo: ausencia de consecuencias catastróficas sobre el o los usuarios y el ambiente.	Resguardo: capacidad para alcanzar niveles aceptables de riesgo sobre los usuarios, ambiente y otros sistemas.
Mantenibilidad: habilidad para experimentar modificaciones y reparaciones.	Mantenibilidad: capacidad para experimentar modificaciones, implica: <ul style="list-style-type: none"> ▪ <i>Facilidad de Cambios, Estabilidad y Facilidad de Pruebas.</i>
Seguridad: ausencia de acceso no autorizado para el manejo del estado del sistema. <ul style="list-style-type: none"> ▪ Confidencialidad: habilidad para mantener en reserva la información. ▪ Integridad: ausencia de alteraciones inadecuadas del sistema. ▪ Disponibilidad: predisposición para un servicio correcto. 	Funcionalidad: capacidad para proveer las funciones que permitan cumplir con necesidades implícitas y declaradas; implica: <ul style="list-style-type: none"> ▪ Seguridad: capacidad para proteger la información y los datos de personas no autorizadas; esta a su vez implica: <i>Confidencialidad, integridad y disponibilidad.</i> <ul style="list-style-type: none"> ○ Confidencialidad: capacidad del sistema para denegar el acceso a personas no autorizadas. ○ Integridad: capacidad para evitar modificaciones no autorizadas. ○ Disponibilidad: está incluida en confiabilidad

6. CONCLUSIONES

Este artículo ha presentado a DeSCMO, un modelo conceptual para el dominio de los sistemas fiables y su posterior refinamiento utilizando el modelo de calidad estándar ISO/IEC 9126-1 para especificar los atributos de fiabilidad, principal propiedad de calidad de estos sistemas. DeSCMO permite representar conceptualmente los principales elementos y la terminología básica que ha sido utilizada en la comunidad de investigadores, constituye una herramienta útil para iniciar una ontología dentro del área de los sistemas fiables o computación fiable. El manejo de un vocabulario común en un contexto particular resulta de gran importancia en el desarrollo de los sistemas, debido a que permite superar serías deficiencias en el manejo de los términos relacionados con el sistema de software que se desea obtener, permitiendo entre otras cosas implementar efectivamente los diversos requisitos del software y reducir errores conceptuales. El modelo de calidad puede ser utilizado como herramienta para la caracterización del dominio de los sistemas fiables. Como trabajo futuro se ha propuesto el refinamiento de DeSCMO con la incorporación de nuevos elementos, la cuantificación de los atributos de calidad que corresponden a las propiedades de fiabilidad, así como la adaptación del concepto de fiabilidad utilizando el enfoque de la orientación a aspectos, y en particular su relación con la noción de incumbencia (concern), característica en este paradigma emergente de desarrollo de sistemas de software.

Referencias

- [1] AVIZIENIS A., LAPRIE J_C. y RANDELL B. Fundamental Concepts of Dependability. Technical report, LAAS CNRS, October 2000.
- [2] AVIZIENIS A., LAPRIE J_C. y RANDELL B. Fundamental Concepts of Dependability. Research report N°1145, LAAS CNRS, April 2001.
- [3] AVIZIENIS A., LAPRIE J_C., RANDELL B. y LANDWEHR C. Basic concepts and taxonomy of dependable and secure computing. IEEE Transaction on dependable and Secure Computing Vol. 1, issue 1, pp 11-33, IEEE 2004.
- [4] BERTOIA M., TROYA J. y VALECILLO A. Aspectos de Calidad en el Desarrollo de Software Basado en Componentes. Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga, 2002.
- [5] CHIRINOS L., LOSAVIO F y MATTEO A. Identifying Quality-Based Requirements. Information Systems Management. Editorial: Auerbach Publications, 21(1),15-26, 2004.
- [6] ISO/IEC: FCD 9126-1. Information Technology - Software Engineering Product Quality. Part 1: Quality Model. 2001.
- [7] ISO/IEC: FCD 25030. Software engineering: Software product Quality Requirements and Evaluation (SQuaRE) - Quality Requirements. June 2005.
- [8] JACOBSON I., BOOCH G. y RUMBAUGH J. El Lenguaje Unificado de Modelado. Segunda Edición. Madrid: Addison Wesley. 2000.
- [9] LAPRIE J.C. Dependable computing and fault tolerance: concepts and terminology. In Digest of FTCS-15, 2-11,1985.
- [10] LAPRIE J.C. Dependability: Basic Concepts and terminology. IFIP WG 10.4 - Dependable Computing and Fault Tolerance, August 1994.
- [11] LOSAVIO F., CHIRINOS L., LEVY N. y RAMDANE-CHERIF A. Quality Characteristics for Software Architectures. Journal of Object Technology, 2(2), 133-150. 2003.
- [12] RANDELL B. Facing up to Faults. ICSE'04 Workshop on Architecting Dependable Systems, May 2004.