Identificación de los niveles de Integridad de la Seguridad en el desarrollo de software crítico para sistemas ferroviarios

Eterovic, Jorge Esteban
Donadello, Domingo
Universidad Nacional de La Matanza,
Departamento de Ingeniería e Investigaciones Tecnológicas

Abstract

Los sistemas críticos, cuyo buen funcionamiento condiciona nuestra vida cotidiana, necesitan cumplir con determinados niveles de Integridad de la Seguridad (SIL), antes de su puesta en funcionamiento. La criticidad de estos sistemas reside cada vez más en los productos de software que contienen.
En la industria ferroviaria hay una gran cantidad de sistemas críticos que tienen productos de software. Éste software debe cumplir con los criterios de Confiabilidad, Disponibilidad, Mantenibilidad y Seguridad (RAMS, por sus siglas en inglés) establecidos en las normas internacionales, en particular en la norma EN 50126 [1], utilizada en la Unión Europea.
Cuanto más peligrosas sean las consecuencias de una falla del software, mayor será el nivel requerido de Integridad de Seguridad del mismo.
El objetivo de este trabajo es identificar éstos niveles de Integridad de la Seguridad en el desarrollo de software crítico para sistemas ferroviarios.

Palabras Clave

Software crítico ferroviario. Niveles de Integridad de la Seguridad. SIL.

Introducción

El desarrollo de sistemas para la industria ferroviaria debe ser llevado a cabo con ciertas premisas relativas a la seguridad, tratando de identificar los riesgos para controlarlos y llevarlos a un nivel aceptable. Los requisitos de seguridad pueden ser cuantitativos o cualitativos.
En algunas áreas, tales como el software, donde se pueden producir fallas sistemáticas, las buenas prácticas de ingeniería señalan que para cumplir con los requisitos de integridad se deben establecer y aplicar adecuados niveles de Integridad de la Seguridad (SIL) [2].
Existen técnicas estandarizadas para evaluar y controlar el riesgo derivado de las fallas aleatorias. El riesgo derivado de las fallas sistemáticas se controla, en distintas actividades de la ingeniería, a través de la comprobación rigurosa y la aplicación de normas, códigos y del uso de las buenas prácticas generalmente aceptadas.
Sin embargo, como la complejidad de los diseños aumenta, las fallas sistemáticas contribuyen, en una mayor proporción, en el incremento del riesgo.
Para el software, todas las fallas son sistemáticas. En el software y algunas otras áreas donde los diseños pueden ser particularmente complejos, tales como el diseño de la electrónica, las mejores prácticas actuales aconsejan hacer uso de niveles de Integridad de la Seguridad para controlar dichas las fallas sistemáticas.
Los SILs se describen en la serie de normas IEC 61508 [3], ampliamente utilizadas en el desarrollo de software crítico. Además, específicamente para la industria ferroviaria, las normas EN 50128 [4] y EN 50129 [5] definen los sistemas o partes de sistemas para los cuales es aplicable el uso de los SILs.
Los SILs representan diferentes niveles de rigurosidad en el proceso de desarrollo. Se definen cinco niveles, que van desde SIL 4, el más riguroso, al SIL 1, el menos riguroso. Las funciones que no se basan en los SILs para controlar el riesgo, pueden ser descriptas como SIL 0.
Cada nivel de Integridad se asocia con una probabilidad de falla. En la Tabla 1 se muestra la probabilidad de falla para cada uno de los SILs, de acuerdo con la norma IEC 61508.

Modo de Operación de baja demanda (probabilidad de falla en la demanda)	Modo de Operación de Continua / Alta demanda (tasa por hora de falla peligrosa)	Nivel de Integridad de Seguridad (SIL)
$\geq 10^{-5}$ a 10^{-4}	$\geq 10^{-9}$ a 10^{-8}	4
$\geq 10^{-4}$ a 10^{-3}	$\geq 10^{-8}$ a 10^{-7}	3
$\geq 10^{-3}$ a 10^{-2}	$\geq 10^{-7}$ a 10^{-6}	2
$\geq 10^{-2}$ a 10^{-1}	$\geq 10^{-6}$ a 10^{-5}	1

Tabla1

En la mayoría de los casos, se debe utilizar la columna "**Modo de Operación Continua / Alta demanda**". La columna "**Baja demanda**" sólo se debería utilizar si se espera que la demanda se produzca:

- No más de 1 vez al año; y
- No es mayor a 2 veces que cuando la función se pone a prueba.

No se establecen probabilidades para SIL 0.
Luego, deberían establecerse las probabilidades de falla para las funciones sistemáticas a efectos de lograr un nivel de riesgo aceptable para el sistema en general.
Para los sistemas ferroviarios, las funciones que el software debería realizar estarán definidas en las Especificaciones de Requisitos.
De la misma manera que los requisitos de seguridad se fijan a nivel de sistema y forman parte de los requisitos generales del mismo, es habitual establecer una Especificación de Requisitos de Seguridad del Software ya sea como un subconjunto de la Especificación de Requisitos del Software o como un documento separado.
La norma EN 50128 proporciona una guía sobre las características de tolerancia a fallas.
Como el software no se desgasta ni se rompe, todas las fallas de software serán sistemáticas. La mayoría de las fallas de software son el resultado de errores en la programación, que a su vez son el resultado de fallas en el proceso de desarrollo, tales como una especificación incorrecta, o un error en la implementación de una especificación.
En general, si un sistema ferroviario incluye software, entonces la Integridad de la Seguridad del sistema dependerá de la Integridad de la Seguridad del software. Entonces, la confiabilidad del sistema deberá ser tratada mediante la especificación del SIL del software, que será el mismo que el SIL para el sistema.
Se requiere una Especificación de Requisitos de Seguridad del Software y una Especificación de Requisitos del Software para el software. Siempre los requisitos relacionados con la seguridad deberán ser claramente identificados.
La Especificación de Requisitos de Seguridad del Software tendrá un papel fundamental en los casos de seguridad del sistema. Tendrá que demostrar que los requisitos de seguridad del software son suficientes y que el software cumple con sus requisitos de seguridad.
Para lograr esto, la Especificación de Requisitos de Seguridad del Software debe ser completa, precisa e inteligible tanto para aquellos que desarrollan el software como para los que los usuarios.
Idealmente se debería poder asegurar que la Especificación de Requisitos del Software tenga todos estos atributos. Sin embargo, no se ha encontrado un consenso dentro de la comunidad de la ingeniería del software sobre métodos de predicción de la probabilidad de fallas de software [6].
Por lo tanto, si no se puede estimar la probabilidad de falla del software, no es posible estimar la probabilidad de falla del sistema que contiene el software.
Es posible, sin embargo, estimar la probabilidad de falla del sistema por causas que no sean del software y presentar este cálculo, cuidadosamente explicado, junto con los SILs de las funciones del sistema en un "Caso de Seguridad" [7]. Si se utilizaran árboles de fallas, la probabilidad que el sistema falle por causas que no sean del software, pueden ser calculados estableciendo que la probabilidad de falla de software sea cero, aunque debe entenderse

que se trata de un una forma de excluir la falla del software en el proceso de cálculo, y no que el software no falla.
El objetivo de este trabajo es identificar los niveles de Integridad de la Seguridad en el desarrollo de software crítico para sistemas ferroviarios.

Elementos del Trabajo y metodología

La documentación del sistema debe identificar las funciones del sistema relacionado con la seguridad tanto para el software, como para las interfaces del mismo. El sistema en el cual el software está integrado debe estar totalmente definido en relación a los siguientes elementos:

- funciones e interfaces;
- condiciones de aplicación;
- configuración o arquitectura del sistema;
- situaciones peligrosas a controlar;
- requisitos de Integridad de Seguridad;
- asignación de requisitos y del SIL al software y al hardware;
- restricciones de tiempo

La Integridad de la Seguridad del software debe especificarse como uno de cinco niveles, que como se vió anteriormente, van desde el SIL 0 al SIL 4.
El nivel requerido de Integridad de la Seguridad del software se debe decidir y evaluar a nivel del sistema, tomando como base el nivel de Integridad de la Seguridad del sistema y el nivel de riesgo asociado con el uso del software en el sistema.
Según las normas europeas, para la parte de software de un sistema que tengan un impacto en la seguridad por debajo del SIL 1, se deben cumplir, por lo menos, los requisitos asociados al SIL 0. Esto ocurrirá cuando exista una incertidumbre en la evaluación del riesgo, e incluso en la identificación de situaciones peligrosas.
Si trabajamos con un alto grado de incertidumbre en la evaluación del riesgo, es recomendable aplicar un nivel bajo de Integridad de la Seguridad, representado por SIL 0, en lugar de no utilizar ninguno.
Para trabajar de acuerdo con las normas europeas, se debe demostrar que se ha satisfecho cada uno de los requisitos definidos respecto al nivel de Integridad de la Seguridad del software y que por lo tanto se ha cumplido con el objetivo en cuestión.
En los casos en los que se califique a un requisito con las palabras "dentro del alcance requerido por el nivel de Integridad de Seguridad del software", se estará indicando que se han utilizado una serie de técnicas y medidas para satisfacer dicho requisito.
En los casos en los que se aplique el punto anterior, se deberán usar las Tablas 2, 3, y 4 para hacer la selección de las técnicas y medidas adecuadas al nivel de Integridad de Seguridad del software. Dicha selección debe documentarse en el Plan de Garantía de Calidad del Software o en otro documento al que haga referencia el Plan de Garantía de Calidad del Software [8].
A modo de ejemplo, en los Anexos A, se describen cada una de las técnicas o medidas referenciadas en las tablas.
Si no se utiliza una técnica o medida calificada como altamente recomendable (AR) en las tablas, se deben detallar entonces las razones que justifiquen el uso de técnicas alternativas y se deben registrar en el Plan de Garantía de Calidad del Software. Esto no será necesario si se utiliza una combinación homologada de técnicas que estén especificadas en la tabla correspondiente. Se debe demostrar que se han aplicado las técnicas seleccionadas de forma correcta [9].
Si se propone utilizar una técnica o una medida que no aparezca en las tablas, se debe justificar entonces su efectividad e idoneidad para cumplir los requisitos particulares y los objetivos globales del apartado.
Debe verificarse, mediante la inspección de los documentos requeridos por las normas europeas, la conformidad con los requisitos de cada apartado en particular y sus técnicas y medidas respectivas detalladas en

DOCUMENTACIÓN	SIL 0	SIL 1	SIL 2	SIL 3	SIL 4
Planificación					
1. Plan de Garantía de Calidad del Software	AR	AR	AR	AR	AR
2. Informe de Verificación de la Garantía de Calidad del Software	AR	AR	AR	AR	AR
3. Plan de Gestión de la Configuración del Software	AR	AR	AR	AR	AR
4. Plan de Verificación del Software	AR	AR	AR	AR	AR
5. Plan de Validación del Software	AR	AR	AR	AR	AR
Requisitos del Software					
6. Especificación de Requisitos del Software	AR	AR	AR	AR	AR
7. Especificación de Ensayos del Software en Conjunto	AR	AR	AR	AR	AR
8. Informe de Verificación de los Requisitos del Software	AR	AR	AR	AR	AR
Arquitectura y diseño					
9. Especificación de la Arquitectura del Software	AR	AR	AR	AR	AR
10. Especificación del Diseño del Software	AR	AR	AR	AR	AR
11. Especificación de las Interfaces del Software	AR	AR	AR	AR	AR
12. Especificación de Ensayos de Integración del Software	AR	AR	AR	AR	AR
13. Especificación de Ensayos de Integración del Software / Hardware	AR	AR	AR	AR	AR
14. Informe de Verificación de Diseño y Arquitectura del Software	AR	AR	AR	AR	AR
Diseño de Componentes					
15. Especificación de Diseño de los Componentes Software	R	AR	AR	AR	AR
16. Especificación de Ensayos de los Componentes Software	R	AR	AR	AR	AR
17. Informe de Verificación del Diseño de los Componentes Software	R	AR	AR	AR	AR
Implementación y Ensayos de Componentes					
…..					
…..					

Tabla 2. SILs relativos al Ciclo de Vida y la Documentación

las tablas.
Cuando sea procedente, se deberán tener en cuenta también otras pruebas objetivas, auditorías y ensayos.

Criterios para la selección de técnicas y medidas

Para establecer los niveles de SIL requeridos, se comienza trabajando con la documentación asociada al Ciclo de Vida del desarrollo del software.
En la Tabla 2, a manera de ejemplo, se muestran los SILs para las primeras 4 etapas del Ciclo de Vida.
Luego se desarrollan tablas de nivel inferior para cada apartado de la tabla anterior. Por ejemplo, el apartado "6. Especificación de Requisitos del Software" de la Tabla 2 se desarrolla en la Tabla 3.

TÉCNICA / MEDIDA	Ref.	SIL 0	SIL 1	SIL 2	SIL 3	SIL 4
1. Métodos Formales	A.1	-	R	R	AR	AR
2. Modelado	Tabla 3	R	R	R	AR	AR
3. Metodología Estructurada	A.2	R	R	R	AR	AR
4. Tablas de Decisión	A.3	R	R	R	AR	AR

Requisitos:
1) La Especificación de Requisitos del Software debe incluir una descripción del problema en lenguaje natural y todas las notaciones formales o semiformales necesarias.
2) La tabla refleja requisitos adicionales para definir la especificación de forma clara y precisa. Se deben seleccionar una o más de estas técnicas para satisfacer el nivel de Integridad de Seguridad del software utilizado.

Tabla 3. Especificación de Requisitos del Software

TÉCNICA / MEDIDA	Ref.	SIL 0	SIL 1	SIL 2	SIL 3	SIL 4
1. Modelado de Datos	A.4	R	R	R	AR	AR
2. Diagrama de Flujo de Datos	A.5	-	R	R	AR	AR
3. Diagrama de Flujo de Control	A.6	R	R	R	AR	AR
4. Máquinas de Estado Finitos o Diagramas de Transición de Estados	A.7	-	AR	AR	AR	AR
5. Redes de Petri Temporizadas	A.8	-	R	R	AR	AR
6. Tablas de Decisión/Tablas de Verdad	A.9	R	R	R	AR	AR
7. Métodos Formales	A.10	-	R	R	AR	AR
8. Modelado de las Prestaciones	A.11	-	R	R	AR	AR
9. Prototipado/Animación	A.12	-	R	R	AR	AR
10. Diagramas de Estructura	A.13	-	R	R	AR	AR
11. Diagramas de Secuencias	A.14	R	AR	AR	AR	AR
Requisitos: 1) Se deben definir y utilizar directrices de modelado. 2) Se debe seleccionar al menos una de las técnicas “AR”.						

Tabla 4. Modelado

En ésta tabla tenemos la Técnica/Medida “2. Modelado”, que se desarrolla en la Tabla 4.
En cada una de las tablas se hace referencia a un anexo “A”, que es de carácter informativo. A manera de ejemplo se muestra en el Anexo A.4 la descripción del apartado “Modelado de Datos” de la Tabla 4.
Junto a cada técnica o medida que aparece en las tablas, habrá un requisito para cada nivel de Integridad de Seguridad del software. En este ejemplo, los requisitos para los niveles 1 y 2 de Integridad de Seguridad del software son los mismos para cada técnica. Del mismo modo, cada técnica tiene los mismos requisitos en los niveles 3 y 4 de Integridad de Seguridad del software. Estos requisitos pueden ser:

– "M" este símbolo significa que el uso de una técnica es obligatoria (Mandatorio).
– "AR" este símbolo significa que la técnica o la medida es altamente recomendable (Altamente Recomendado) para ese nivel de Integridad de la Seguridad. Si no se utiliza esa técnica o medida, se debe proporcionar una justificación detallada en el Plan de Garantía de Calidad del Software de por qué se han utilizado técnicas

– alternativas.

Objetivo:
Crear un modelo de datos.
Descripción
En informática se llama modelado de datos al proceso de creación de un modelo de datos mediante la aplicación de descripciones de modelos de datos formales utilizando las técnicas de modelado de datos. En ingeniería del software, un modelo de datos es un modelo abstracto que describe los modos de representación y acceso a los datos. Los modelos de datos definen formalmente objetos de datos y las relaciones entre los objetos de datos para un ámbito de interés determinado. Algunas aplicaciones típicas de modelos de bases de datos incluyen soportar el desarrollo de bases de datos y permitir el intercambio de datos para un ámbito de interés determinado. Los modelos de datos se especifican en un lenguaje de modelado de datos.

- "R" este símbolo significa que la técnica o la medida es recomendable (Recomendado) para ese nivel de Integridad de la Seguridad. Éste es un nivel de recomendación inferior al "AR", y se pueden combinar dichas técnicas para formar parte de un paquete.
- "-" este símbolo significa que no existen recomendaciones ni a favor ni en contra relativas a la utilización de esa técnica o medida.
- "NR" este símbolo significa que la técnica o la medida NO es recomendable (No Recomendado) para ese nivel de integridad de la seguridad. La utilización de esta técnica o medida debe justificarse de forma detallada en el Plan de Garantía de Calidad del Software.

La combinación de técnicas o medidas se expondrá en el Plan de Garantía de Calidad del Software, seleccionando una o más técnicas o medidas, a menos que las notas adjuntas a la tabla impongan otros requisitos.
Dichas notas pueden incluir referencias a técnicas homologadas o a combinaciones homologadas de técnicas. Si se utilizan éstas técnicas o combinaciones de técnicas, más todas las técnicas de carácter obligatorio, entonces se deberán aceptar como válidas y solo se deberá verificar que se apliquen correctamente.
Si se utiliza un conjunto de técnicas diferente y puede justificarse su uso, entonces puede considerarlas aceptables.

Resultados

Trabajando de ésta manera se establecen una serie de requisitos que se deben cumplir en el desarrollo, implantación y mantenimiento de cualquier software relacionado con la seguridad destinado a aplicaciones de control y protección de ferrocarriles.
Se definen los requisitos relativos a cada una de las etapas del Ciclo de Vida del desarrollo del software que incluyen las actividades de desarrollo, implantación y mantenimiento.
El concepto clave es el de los niveles de Integridad de Seguridad del software. Se identifican cinco niveles de integridad de seguridad del software, siendo 0 el nivel mínimo y 4 el máximo.
Se establece como norma que cuanto más peligrosas sean las consecuencias de una falla del software, mayor será el nivel requerido de Integridad de la Seguridad del software.
Se han identificado técnicas y medidas para los cinco niveles de Integridad de la Seguridad del software. En las Tablas 1, 2 y 3 se muestran las técnicas y medidas requeridas para dichos niveles de SIL.
En este ejemplo y para simplificar la identificación de los requisitos, las técnicas requeridas para el nivel 1 son las mismas que para el nivel 2 y las técnicas requeridas para el nivel 3 son las mismas que para el nivel 4.
No se dan indicaciones sobre qué nivel de Integridad de la Seguridad del software es apropiado para un riesgo determinado. Este análisis dependerá de muchos factores, incluyendo la naturaleza de la aplicación, del grado en que otros sistemas llevan a cabo funciones de seguridad y de factores sociales y económicos.
Para ello se requiere que se adopte un enfoque sistemático para:

a) identificar peligros, evaluar riesgos y tomar decisiones en función de los criterios de riesgo;
b) identificar la reducción de riesgo necesaria para cumplir con los criterios de aceptación de riesgos;

c) definir una Especificación global de Requisitos de Seguridad del sistema con las protecciones necesarias a efectos de conseguir la reducción de riesgo requerida;
d) seleccionar una arquitectura del sistema adecuada;
e) planificar, supervisar y controlar las actividades técnicas y de gestión necesarias para convertir la Especificación de Requisitos de Seguridad del sistema en un Sistema Relacionado con la Seguridad a partir de las características validadas de Integridad de la Seguridad.

A medida que se descompone la especificación en un diseño que incluye sistemas y componentes relacionados con la seguridad, se produce una nueva asignación de niveles de Integridad de la Seguridad. Y finalmente se llegará a los niveles de Integridad de la Seguridad requeridos para el software.

El estado actual de la técnica es tal que ni la aplicación de métodos para garantizar la calidad, como las medidas para evitar y detectar errores, ni la aplicación de soluciones de software tolerante a errores, pueden garantizar la seguridad absoluta del sistema.

No hay manera conocida de demostrar la ausencia de errores en un software razonablemente complejo, especialmente la ausencia de errores de especificación y diseño.

Por ello, las buenas prácticas aplicadas al desarrollo de software de alta complejidad, recomiendan el uso de:

- métodos de diseño descendentes (arriba-abajo);
- la modularidad;
- la verificación de cada fase del Ciclo de Vida del desarrollo;
- la verificación de los componentes y de las librerías de los componentes;
- una documentación y trazabilidad claras;
- documentos auditables;
- la validación;
- la evaluación;
- la gestión de la configuración y el control de las modificaciones; y
- el estudio apropiado de las competencias del personal y de la organización.

Discusión

La Especificación de Requisitos de Seguridad del sistema identifica todas las funciones de seguridad asignadas al software y determina el nivel de Integridad de la Seguridad del mismo para dichas funciones.

Las etapas funcionales sucesivas en la aplicación de las normas europeas son las que se detallan a continuación:

a) definir la Especificación de Requisitos del Software y en paralelo considerar la arquitectura del software. La arquitectura del software es donde se desarrolla la estrategia de seguridad para el software y para el nivel de Integridad de la Seguridad del software;
b) diseño, desarrollo y ensayo del software de acuerdo con el Plan de Garantía de la Calidad del Software, el nivel de Integridad de la Seguridad del software y el ciclo de vida del software;
c) integrar el software en el hardware objetivo y verificar su funcionalidad;
d) aceptar e implantar el software;
e) si se requiere el mantenimiento del software durante su vida operativa, entonces se ha de hacer de acuerdo con la norma europea.

Durante el Ciclo de Vida del desarrollo del software se deberán realizar varias actividades. Estas incluyen: ensayos, verificación, validación, evaluación, aseguramiento de la calidad y modificación y control de las modificaciones.
Se deberán establecer requisitos para las herramientas de soporte y para los sistemas configurados mediante datos de aplicación o algoritmos.
Se deberán establecer también requisitos en lo que concierne a la independencia de roles y a la competencia del personal implicado en el desarrollo del software.
Si bien las normas no obligan a utilizar un Ciclo de Vida específico para el desarrollo del software, es muy conveniente adoptar uno y su correspondiente conjunto de documentación asociada.

Conclusión

Una conclusión importante a la que se arribó, luego del análisis de los niveles de Integridad de la Seguridad del software crítico en sistemas ferroviarios, es que el SIL depende de la integridad ante fallas sistemáticas y ante fallas aleatorias.
A las fallas aleatorias, inherentes a la fiabilidad de los equipos, fallas debido a la fatiga, deterioro por el tiempo de uso, etc. no los hemos considerado, ya que la investigación se basó en la identificación de los requisitos de Integridad de la Seguridad del software.
Las fallas sistemáticas son causadas por errores humanos durante el diseño, fabricación, verificación, validación, implantación o mantenimiento del software.
Una forma de minimizar éstas fallas sistemáticas es adoptar un adecuado Ciclo de Vida de desarrollo y técnicas o medidas del software adecuadas para el diseño y desarrollo del producto.
Para cada nivel SIL, las normas EN 5012X son más o menos exigentes, determinando la forma de minimizar las fallas sistemáticas.
Para minimizar los riesgos debidos a las fallas sistemáticas, las técnicas o medidas del software deberán ser de aplicación Mandatoria (M) o Altamente Recomendable (AR) para sistemas con requerimientos de un SIL elevado, como es el caso del software crítico en sistemas ferroviarios.

Referencias

1. Norma EN 50126:2005. Aplicaciones Ferroviarias. Especificación y demostración de la fiabilidad, la disponibilidad, la mantenibilidad y la seguridad (RAMS). 2005.
2. Engineering Safety Management Guidance. The yellow book. Fundamental and Guidance. Volume 2. Published by Rail Safety and Standards Board. 2007
3. Norma BS EN 61508:2010. Functional safety of electrical/electronic/programmable electronic safety-related systems. BSI Standards Publication. 2010
4. Norma EN 50128:2012. Aplicaciones Ferroviarias. Sistemas de comunicaciones, señalización y procesamiento. Software para sistemas y protección del ferrocarril. 2012.
5. Norma EN 50129:2005. Aplicaciones ferroviarias. Sistemas de comunicación, señalización y procesamiento. Sistemas electrónicos relacionados con la seguridad para la señalización. 2005.
6. Engineering Safety Management (The Yellow Book). Volumes 1 and 2 Fundamentals and Guidance. Publicado por Rail Safety and Standards Board. 2007.
7. Mark Charlwood, Shane Turner and Nicola Worsell. A methodology for the assignment of safety integrity levels (SILs) to safety-related control functions implemented by safety-related electrical, electronic and programmable electronic control systems of machines. Health and Safety Executive. 2004
8. Brosseau, Jim. Software Quality Attributes: Following All the Steps, Clarrus Consulting Group Inc. Noviembre de 2010.
9. Kelly T. A Systematic Approach to Safety Case Management, 04AE-149, 2003.

Datos de Contacto:

Mag. Jorge Esteban Eterovic.
Universidad Nacional de La Matanza. Departamento de Ingeniería e Investigaciones Tecnológicas.
Florencio Varela 1903, (B1754JEC) San Justo, Prov. de Buenos Aires, Argentina.
Tel: (54 11) 4480-8900
E-mail. jeterovic@ing.unlam.edu.ar

Mag. Domingo Donadello.
Universidad Nacional de La Matanza. Departamento de Ingeniería e Investigaciones Tecnológicas.
Florencio Varela 1903, (B1754JEC) San Justo, Prov. de Buenos Aires, Argentina.
Tel: (54 11) 4480-8900
E-mail.ddonadel@ing.unlam.edu.ar