

**UNIVERSIDAD NACIONAL DE EDUCACION A DISTANCIA.**

**MÁSTER UNIVERSITARIO DE INVESTIGACIÓN EN INGENIERÍA DE  
SOFTWARE Y SISTEMAS INFORMÁTICOS.**

**ITINERARIO DE INGENIERIA DE SOFTWARE  
CODIGO: 31105128(2C).**



**ARCHE: ARQUITECTURA Y TOMA DE DECISIONES DE UN SISTEMA DE  
INFORMACIÓN PARA LA PUBLICACIÓN DE EVENTOS SÍSMICOS DE  
EMERGENCIA (TERREMOTOS Y TSUNAMIS).**

**ALUMNO: HECTOR HENRY CHAMUCERO GRACIA.  
DIRECTOR: DR. JOSÉ FÉLIX ESTÍVARIZ LÓPEZ.  
CURSO ACADEMICO 2013-2014  
CONVOCATORIA: SEPTIEMBRE DE 2014.**

**UNIVERSIDAD NACIONAL DE EDUCACION A DISTANCIA.**

**MÁSTER UNIVERSITARIO DE INVESTIGACIÓN EN INGENIERÍA DE  
SOFTWARE Y SISTEMAS INFORMÁTICOS.**

**ITINERARIO DE INGENIERIA DE SOFTWARE  
CODIGO: 31105128(2C).**

**ARCHE: ARQUITECTURA Y TOMA DE DECISIONES DE UN SISTEMA DE  
INFORMACIÓN PARA LA PUBLICACIÓN DE EVENTOS SISMICOS DE  
EMERGENCIA (TERREMOTOS Y TSUNAMIS).**

**TRABAJO FIN DE MASTER.**

**ALUMNO: HECTOR HENRY CHAMUCERO GRACIA.  
DIRECTOR: DR. JOSÉ FÉLIX ESTÍVARIZ LÓPEZ.**

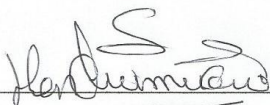
## CALIFICACIÓN



**Autorización de publicación y difusión del Trabajo de Fin de Máster para fines académicos.**

**Autorización**

Autorizo a la Universidad Nacional de Educación a Distancia (UNED) a difundir y utilizar, con fines académicos, no comerciales y mencionando expresamente al autor, tanto la memoria de este Trabajo Fin de Máster, como el código, la documentación y/o el prototipo desarrollado.

Firma: 

Héctor Henry Chamucero Gracia

Juan del Rosal, 16  
28040, Madrid  
Tel: 91 398 89 10  
Fax: 91 398 89 09  
[www.issi.uned.es](http://www.issi.uned.es)

## RESUMÉN.

En este trabajo de investigación se hace la propuesta del diseño de una arquitectura orientada a servicios (SOA) de un sistema de información para la publicación de eventos sísmicos y posteriormente se evalúa la arquitectura con la herramienta ArchE, teniendo en cuenta los escenarios para el atributo de calidad de modificabilidad.

En primer lugar se realiza el proceso de negocio para la publicación de los eventos sísmicos, se analizan los requerimientos funcionales, se realizan los casos de uso y se diseña la arquitectura SOA.

En segundo lugar, a partir de los casos de uso se definen las funcionalidades. Se instala la herramienta ArchE, se crea el proyecto publicarsismos en ArchE, se alimenta la herramienta ArchE con las funcionalidades y sus relaciones. Con esta información ArchE propone el primer diseño Arquitectónico.

En tercer lugar se alimenta ArchE con el primer escenario, pero antes se debe aplicar el marco de razonamiento de modificabilidad. El propósito de este primer escenario es que el sistema sea más flexible para el cálculo de varias magnitudes para el evento sísmico; es decir que se puedan adicionar, modificar y borrar las configuraciones para el cálculo de varias magnitudes. ArchE Automáticamente satisface este primer escenario y propone un segundo diseño arquitectónico.

En cuarto lugar se alimenta ArchE con el segundo escenario. El propósito de este segundo escenario es que el sistema permita publicar no solo eventos sísmicos tectónicos si no también eventos sísmicos volcánicos. En primera instancia ArchE no satisface los escenarios y propone varias tácticas. Se realizan varias iteraciones de aplicación de tácticas hasta que ArchE satisface los escenarios y al final propone un tercer diseño arquitectónico.

En quinto y último lugar se realiza un análisis del último diseño arquitectónico propuesto por ArchE desde el punto de vista de la arquitectura SOA.

**Palabras claves:** ArchE, Atributos de calidad, Tácticas arquitectónicas, Diseño arquitectónico, Arquitectura de software, arquitectura orientada a servicios SOA, patrones arquitectónicos, eventos sísmicos, eventos volcánicos, SEISAN, Estaciones sismológicas, tectónicos.

## TABLA DE CONTENIDO

<b>1</b>	<b>INTRODUCCIÓN.....</b>	<b>11</b>
<b>2</b>	<b>OBJETIVO DE LA INVESTIGACION. ....</b>	<b>12</b>
2.1	OBJETIVO GENERAL.....	12
2.2	Objetivos específicos.....	12
<b>3</b>	<b>MARCO CONCEPTUAL.....</b>	<b>13</b>
3.1	Definición de arquitectura de software.....	13
3.2	Diseño arquitectónico. ....	13
3.3	Vistas arquitectónicas. ....	14
3.4	Patrones arquitectónicos.....	14
3.4.1	Arquitectura en capas. ....	15
3.4.2	Arquitectura de repositorio.....	15
3.4.3	Arquitectura cliente servidor. ....	16
3.4.4	Arquitectura de tubería y filtro. ....	16
3.5	Arquitectura orientada a servicios SOA.....	17
3.6	Eventos sísmicos. ....	18
3.7	Ondas sísmicas.....	19
3.8	Componentes de una estación sismológica. ....	20
<b>4</b>	<b>ESCENARIOS PARA ATRIBUTOS DE CALIDAD Y TÁCTICAS. ....</b>	<b>21</b>
<b>5</b>	<b>ARCHITECTURE EXPERT DESIGN ASSISTANT (ARCHE).....</b>	<b>27</b>
5.1	Descripción de ArchE.....	27
5.2	Instalación de ArchE.....	28
5.2.1	Instalación de java runtime 5.0.....	29
5.2.2	Instalación de eclipse-SDK-3.3-win32. ....	30
5.2.3	Instalación de MySQL mysql-essential-5.0.67-win32.....	31
5.2.4	Instalación de GEF-runtime-3.3.....	32
5.2.5	Instalación de Jess71p1.....	33
5.2.6	Instalación de xmlBlaster_REL_1_6_1.....	36
5.2.7	Instalación de ArchE 3.0.....	37

<b>6</b>	<b>SISTEMA DE INFORMACION PARA LA PUBLICACION DE EVENTOS SISMICOS TECTONICOS DE EMERGENCIA.....</b>	<b>46</b>
6.1	PLANTEAMIENTO DEL PROBLEMA. ....	46
6.2	ALCANCE DEL PROYECTO. ....	46
6.3	OBJETIVO GENERAL.....	46
6.4	OBJETIVOS ESPECIFICOS. ....	47
6.5	PROCESOS DE NEGOCIO. ....	47
6.6	REQUERIMIENTOS DE SOFTWARE.....	48
6.7	REQUERIMIENTOS DE HARDWARE.....	49
6.8	REQUERIMIENTOS FUNCIONALES. ....	49
6.8.1	REQ01: Extracción de las trazas. ....	49
6.8.2	REQ02: Localización del sismo.....	49
6.8.3	REQ03: Generar información para la publicación del sismo. ....	49
6.9	IDENTIFICACION DE SERVICIOS SOA. ....	50
6.9.1	Servicio para localizar el epicentro del sismo.....	50
6.9.2	Servicio para generar los mapas GMT.....	51
6.9.3	Servicio para enviar correos electrónicos. ....	52
6.9.4	Servicio para enviar fax. ....	52
6.9.5	Servicio para realizar llamadas telefónicas. ....	52
6.10	DIAGRAMA GENERAL DE CASOS DE USO.....	52
6.11	DESCRIPCION CASOS DE USO.....	54
6.11.1	Nombre del caso de uso: Extraer trazas del sismo.....	54
6.11.2	Nombre del caso de uso: Localización del sismo.....	56
6.11.3	Nombre del caso de uso: publicar sismo. ....	59
6.12	ARQUITECTURA SOA PROPUESTA PARA EL S.I. PARA LA PUBLICACION DE EVENTOS SISMICOS TECTONICOS DE EMERGENCIA. ....	61
6.12.1	Gestión de procesos de negocio BPM (orquestración de procesos de negocio). ....	61
6.12.2	Gestión de la toma de decisiones empresariales EDM.....	61
6.12.3	Bus de servicio de empresa ESB. ....	61
6.12.4	Procesador de flujos de eventos ESP.....	61
6.12.5	Registro.....	62
6.12.6	Componentes de servicio y su composición.....	62
6.12.7	Mediador de servicios web.....	62
6.12.8	DIAGRAMA DE PROCESOS EN LENGUAJE JPDL (JBOSS JBPM). ....	63
<b>7</b>	<b>EVALUACION DE LA ARQUITECTURA DEL S.I. PARA LA PUBLICACION DE EVENTOS SISMICOS TECTONICOS DE EMERGENCIA CON LA HERRAMIENTA ARCHE.....</b>	<b>66</b>
7.1	Requisitos Funcionales. ....	66
7.2	Requisitos de calidad de modificabilidad. ....	67
7.2.1	Escenario 1: configuración de magnitudes.....	67
7.2.2	Escenario 2: Publicación de eventos sísmicos volcánicos. ....	67

- 7.3 Operación de ArchE.....68**
  - 7.3.1 Funcionamiento de Mysql.....68
  - 7.3.2 Ejecución de XmlBlaster. ....68
  - 7.3.3 Creación del proyecto ArchE publicarsismos. ....70
  - 7.3.4 Creación de las funcionalidades. ....71
  - 7.3.5 Manejo de responsabilidades. ....73
  - 7.3.6 Manejo del mapeo entre funcionalidades y responsabilidades.....74
  - 7.3.7 Creación de los escenarios de calidad de modificabilidad .....76
  
- 8 CONCLUSIONES Y TRABAJOS FUTUROS. .... 100**
  - 8.1 CONCLUSIONES..... 100**
  - 8.2 TRABAJOS FUTUROS..... 100**
  
- 9 LISTADO DE SIGLAS, ABREVIATURAS Y ACRÓNIMOS..... 101**
  
- 10 BIBLIOGRAFIA. .... 105**



## TABLA DE ILUSTRACIONES.

ILUSTRACIÓN 1. REPRESENTACIÓN DE TÁCTICAS.	22
ILUSTRACIÓN 2. APLICACIONES PARA INSTALAR ARCHE.	29
ILUSTRACIÓN 3. INSTALACIÓN DE JAVA RUNTIME.	29
ILUSTRACIÓN 4. FINALIZACIÓN DE LA INSTALACIÓN DE JAVA RUNTIME.	30
ILUSTRACIÓN 5. INSTALACIÓN DE ECLIPSE-SDK-3.3-WIN32	30
ILUSTRACIÓN 6. INICIO DE LA INSTALACIÓN DE MYSQL.	31
ILUSTRACIÓN 7. INSTALACIÓN TYPICA DE MYSQL.	31
ILUSTRACIÓN 8. PROGRESO DE INSTALACIÓN DE MYSQL	32
ILUSTRACIÓN 9. FINALIZACIÓN DE LA INSTALACIÓN DE MYSQL.	32
ILUSTRACIÓN 10. INSTALACIÓN DE GEF-RUNTIME-3.3.	33
ILUSTRACIÓN 11. DESCOMPRIMIR JESS71P1.ZIP	33
ILUSTRACIÓN 12. ESTRUCTURA DE LA CARPETA JESS71P1	34
ILUSTRACIÓN 13. CARPETAS PLUGINS Y FEATURES DE JES71P1.	34
ILUSTRACIÓN 14. COPIADO DE ARCHIVOS A LA CARPETA PLUGINS DE ECLIPSE.	35
ILUSTRACIÓN 15. COPIA DE ARCHIVOS A LA CARPETA FEATURES DE ECLIPSE.	36
ILUSTRACIÓN 16. INSTALACIÓN DE XMLBLASTER.	37
ILUSTRACIÓN 17. DESCOMPRIMIR ARCHE_3_0_0_INSTALLER.ZIP.	37
ILUSTRACIÓN 18. INICIO INSTALACIÓN DE ARCHE.	38
ILUSTRACIÓN 19. TIPO DE INSTALACIÓN DE ARCHE	38
ILUSTRACIÓN 20. SELECCIONAR LA UBICACIÓN DE LA INSTALACIÓN DE ARCHE.	39
ILUSTRACIÓN 21. UBICACIÓN DEL ICONO DE ARCHE.	39
ILUSTRACIÓN 22. COMPROBACIÓN DE LOS PAQUETES QUE SE INSTALARON.	40
ILUSTRACIÓN 23. PROCESO DE CONFIGURACIÓN DE MYSQL DESDE ARCHE	40
ILUSTRACIÓN 24. PROGRESO DE CONFIGURACIÓN DE MYSQL DESDE ARCHE.	41
ILUSTRACIÓN 25. ADVERTENCIA DE ELIMINACIÓN DE LA CARPETA DE DATOS.	41
ILUSTRACIÓN 26. VOLVER A CONFIGURAR MYSQL.	42
ILUSTRACIÓN 27. CONFIGURACIÓN DE MYSQL TERMINADA CON ÉXITO.	42
ILUSTRACIÓN 28. DIGITACIÓN DE LOS PARÁMETROS PARA CREAR LA BASE DE DATOS PARA ARCHE.	43
ILUSTRACIÓN 29. PROCESO PARA LA CREACIÓN DE LA BASE DE DATOS PARA ARCHE.	43
ILUSTRACIÓN 30. CONFIGURACIÓN DE XMLBLASTER.	44
ILUSTRACIÓN 31. INSTALACIÓN COMPLETA DE ARCHE	45
ILUSTRACIÓN 32. PROCESO PARA LA PUBLICACIÓN DE UN EVENTO SÍSMICO.	48
ILUSTRACIÓN 33. INFORMACIÓN ASOCIADA A LA LONGITUD Y LATITUD DEL EVENTO SÍSMICO.	51
ILUSTRACIÓN 34. DIAGRAMA GENERAL DE CASOS DE USO.	53
ILUSTRACIÓN 35. AUTENTICACIÓN DE USUARIOS	55
ILUSTRACIÓN 36. EXTRAER TRAZAS DEL SISMO.	56
ILUSTRACIÓN 37. OPCIÓN LOCALIZAR EL EVENTO SÍSMICO.	58
ILUSTRACIÓN 38. ARQUITECTURA SOA PROPUESTA PARA EL S.I. DE PUBLICACIÓN DE EVENTOS SÍSMICOS TECTÓNICOS.	63
ILUSTRACIÓN 39. DIAGRAMA DE PROCESOS EN LENGUAJE JPD.L.	64

ILUSTRACIÓN 40. CÓDIGO FUENTE DEL DIAGRAMA DE PROCESOS JPDL.	65
ILUSTRACIÓN 41. EJECUCIÓN DE XMLBLASTER.	69
ILUSTRACIÓN 42. PASOS PARA CREAR UN PROYECTO ARCHE.	70
ILUSTRACIÓN 43. CREACIÓN DEL PROYECTO PUBLICARSISMOS.	70
ILUSTRACIÓN 44. PASOS PARA CREAR LAS FUNCIONALIDADES EN ARCHE.	71
ILUSTRACIÓN 45. CAJA DE DIALOGO CREAR NUEVA FUNCIÓN.	72
ILUSTRACIÓN 46. FUNCIONALIDADES PARA LA PUBLICACIÓN DE SISMOS.	72
ILUSTRACIÓN 47. RESPONSABILIDADES PARA EL PROYECTO PUBLICAR SISMOS.	73
ILUSTRACIÓN 48. MENÚ EMERGENTE PARA EL MANEJO DE RESPONSABILIDADES.	74
ILUSTRACIÓN 49. MAPEO ENTRE FUNCIONALIDADES Y RESPONSABILIDADES DEL PROYECTO PUBLICAR SISMOS.	75
ILUSTRACIÓN 50. CREACIÓN DE UN NUEVO MAPEO.	76
ILLUSTRATION 51. ICONO SHOW VIEW AS A FAST VIEW.	77
ILUSTRACIÓN 52. MENÚ EMERGENTE.	77
ILUSTRACIÓN 53. SELECCIONAR MARCO DE RAZONAMIENTO DE MODIFICABILIDAD.	78
ILUSTRACIÓN 54. EJECUCIÓN DEL MARCO DE RAZONAMIENTO DE MODIFICABILIDAD.	79
ILUSTRACIÓN 55. DEPENDENCIA ENTRE RESPONSABILIDADES.	81
ILUSTRACIÓN 56. MENÚ EMERGENTE PARA EL MANEJO DE RELACIONES ENTRE RESPONSABILIDADES.	81
ILUSTRACIÓN 57. CREACIÓN DE RELACIONES.	82
ILUSTRACIÓN 58. RELACIONES PARA EL S.I. PUBLICAR EVENTOS SÍSMICOS.	83
ILUSTRACIÓN 59. MENÚ EMERGENTE PARA EL MANEJO DE ESCENARIOS.	84
ILUSTRACIÓN 60. CREACIÓN DEL ESCENARIO CONFIGURACIÓN DE MAGNITUDES.	84
ILUSTRACIÓN 61. ARQUITECTURA INICIAL PROPUESTA POR ARCHE.	85
ILUSTRACIÓN 62. CREACIÓN DE LA FUNCIONALIDAD PARAMETRIZAR FUNCIONALIDADES.	85
ILUSTRACIÓN 63. CREACIÓN DEL MAPEO ENTRE EL ESCENARIO "CONFIGURACIÓN MAGNITUDES" Y RESPONSABILIDADES RESPECTIVAS.	86
ILUSTRACIÓN 64. EVALUACIÓN DEL ESCENARIO "CONFIGURACIÓN DE MAGNITUDES."	86
ILUSTRACIÓN 65. SEGUNDO DISEÑO ARQUITECTÓNICO PROPUESTO POR ARCHE	87
ILUSTRACIÓN 66. CREACIÓN DEL ESCENARIO "PUBLICAR EVENTOS SÍSMICOS VOLCÁNICOS".	88
ILUSTRACIÓN 67. CREACIÓN DEL MAPEO ENTRE EL ESCENARIO "PUBLICAR EVENTOS SÍSMICOS VOLCÁNICOS" CON SUS RESPECTIVAS RESPONSABILIDADES.	89
ILUSTRACIÓN 68. PRIMERA ITERACIÓN: EVALUACIÓN DE LOS ESCENARIOS "CONFIGURACIÓN DE MAGNITUDES " Y "PUBLICAR EVENTOS SÍSMICOS VOLCÁNICOS"	89
ILUSTRACIÓN 69. PRIMERA ITERACIÓN: APLICACIÓN DE LA SEGUNDA TÁCTICA "ABSTRACTCOMMONRESPONSIBILITIES"	90
ILUSTRACIÓN 70. SEGUNDA ITERACIÓN: EVALUACIÓN DE ESCENARIOS.	91
ILUSTRACIÓN 71. SEGUNDA ITERACIÓN: APLICACIÓN DE LA TÁCTICA 1 "ABSTRACTCOMMONRESPONSIBILITIES"	92
ILUSTRACIÓN 72. TERCERA ITERACIÓN: EVALUACIÓN DE LOS ESCENARIOS.	92
ILUSTRACIÓN 73. TERCERA ITERACIÓN: APLICACIÓN DE CUARTA TÁCTICA "ADJUSTREFINERESPOSABILITIES"	93
ILUSTRACIÓN 74. TERCERA ITERACIÓN: COSTO DE LA CUARTA TÁCTICA "ADJUSTREFINERESPOSABILITIES"	94
ILUSTRACIÓN 75. CUARTA ITERACIÓN: EVALUACIÓN DE LOS ESCENARIOS.	94
ILUSTRACIÓN 76. QUINTA ITERACIÓN: EVALUACIÓN DE LOS ESCENARIOS.	95
ILUSTRACIÓN 77. SEXTA ITERACIÓN: EVALUACIÓN DE LOS ESCENARIOS.	95
ILUSTRACIÓN 78. SEXTA ITERACIÓN: APLICACIÓN DE LA TÁCTICA 2 "INSERTINTERMEDIARY".	96
ILUSTRACIÓN 79. SÉPTIMA EVALUACIÓN: LOS ESCENARIOS SE SATISFACEN	97
ILUSTRACIÓN 80. DISEÑO ARQUITECTÓNICO FINAL PROPUESTO POR ARCHE.	98

## 1 INTRODUCCIÓN.

En la mayoría de casos, el software no es perfecto debido a que su desarrollo es muy complejo para lograr una buena calidad. Los atributos de calidad ayudan a tomar decisiones arquitectónicas, por lo tanto se deben tener en cuenta para el diseño arquitectónico.

ArchE tiene en cuenta dos atributos de calidad (modificabilidad, rendimiento), para asistir al arquitecto en el diseño de la arquitectura. Para comprender el funcionamiento de ArchE en este trabajo fin de master se realizó una serie de pasos:

En el numeral 3. “Marco conceptual” se define la arquitectura de software, el diseño arquitectónico, vistas arquitectónicas, patrones arquitectónicos, arquitectura orientada a servicios SOA, eventos sísmicos y ondas sísmicas.

En el numeral 4, se explica que es un escenario para atributos de calidad, se definen los atributos de calidad y sus tácticas según Bass, Clements, y Kazman.

En el numeral 5, se hace una descripción de la herramienta ArchE y se explica el proceso de su instalación.

En el numeral 6, Se hace el diseño de una arquitectura orientada a servicios SOA para un sistema de información, que realiza la publicación de eventos sísmicos. El objetivo es evaluar esta arquitectura con la herramienta ArchE, teniendo en cuenta el atributo de calidad de modificabilidad.

En el numeral 7, Se definen las funcionalidades, las dependencias entre funcionalidades, se crean los escenarios de modificabilidad y se evalúa la arquitectura para el S.I. de publicación de eventos sísmicos con la herramienta ArchE.

## **2 OBJETIVO DE LA INVESTIGACION.**

### **2.1 OBJETIVO GENERAL.**

El objetivo de la presente investigación es la gestión, en el diseño arquitectónico de los requisitos de calidad de modificabilidad del sistema de información para la publicación de eventos sísmicos de emergencia y su incorporación en la herramienta ArchE para su evaluación y apoyo en la toma de decisiones del desarrollo arquitectónico.

### **2.2 Objetivos específicos.**

Realizar el diseño arquitectónico orientado a servicios (SOA) del sistema de información para la publicación de eventos sísmicos de emergencia, para posterior evaluación con la herramienta ArchE.

Obtener el conocimiento del funcionamiento de la herramienta ArchE: Instalación y su operación.

Incorporación en la herramienta ArchE de las funcionalidades, relaciones entre funcionalidades, escenarios para atributos de calidad de modificabilidad y la aplicación de tácticas del sistema de información para la publicación de eventos sísmicos.

Análisis del diseño arquitectónico final, propuesto por ArchE desde el punto de vista de la arquitectura orientada a servicios SOA.

### 3 MARCO CONCEPTUAL.

#### 3.1 Definición de arquitectura de software.

Según el libro “Software Architecture in Practice, Bass, Clements y Kazman”: define la Arquitectura de software como una Estructura o estructuras de un sistema, lo que incluye sus componentes de software, las propiedades observables de dichos componentes y las relaciones entre ellos.

Según el libro “Agile Software Architecture” define la arquitectura de software como un medio de capturar las características estructurales más importantes y las técnicas para desarrollar un sistema software.

Según el libro “Pattern-Oriented Software Architecture” la definición de Arquitectura de software es diferente para cada persona. Para un desarrollador es la estructura del sistema que se va a construir, para el desarrollador del framework es la forma del sistema que se va a crear, Para todos los interesados es la estructura de alto nivel de la solución a un problema que se quiere resolver.

David Garlan y Mary Shaw en su libro "An introduction to Software Architecture", definen que la Arquitectura es un nivel de diseño que hace foco en aspectos "más allá de los algoritmos y estructuras de datos de la computación; el diseño y especificación de la estructura global del sistema es un nuevo tipo de problema".

#### 3.2 Diseño arquitectónico.

Es un proceso creativo en el cual se diseña un sistema que cubrirá tanto los requerimientos funcionales y no funcionales. El diseño arquitectónico es un conjunto de decisiones a tomar, algunas de ellas son decisiones estructurales que afectaran profundamente el sistema y su proceso de desarrollo.

Los sistemas cubren muchas necesidades de una empresa u organización. Todas las empresas tienen muchas cosas en común, como la contratación de personal, emitir facturas, llevar la contabilidad, liquidación de nóminas, etc. Estos factores en común condujeron al desarrollo de arquitecturas de software que describen la estructura y la organización de tipos particulares de sistemas software.

Cuando se diseña una arquitectura se debe tomar la decisión, que se puede reutilizar de las arquitecturas de software según los factores en común que se tengan con la nueva arquitectura a diseñar.

Debido a la estrecha relación entre los requerimientos no funcionales y la arquitectura de software, el estilo arquitectónico que se elija para un sistema dependerá de los requerimientos no funcionales: Rendimiento, seguridad, protección, disponibilidad y modificabilidad.

### 3.3 Vistas arquitectónicas.

Múltiples vistas arquitectónicas representan el diseño de la arquitectura de un sistema y su documentación. Según Krutchen (1995) en su modelo de vista 4+1 de la arquitectura de software, sugiere que deben existir cuatro vistas arquitectónicas, que se relacionan usando casos de uso o escenarios:

- Vista lógica: Representa los objetos o clases de objeto. En esta vista se relacionan los requerimientos del sistema con entidades.
- Vista de proceso: Muestra en tiempo de operación la interacción de los procesos en que está compuesto el sistema. Esta vista es útil para evaluar las características no funcionales del sistema, como el rendimiento y la disponibilidad.
- Vista de desarrollo: Muestra la descomposición del software para su desarrollo. Esta vista es útil para administradores y programadores de software.
- Vista física: Muestra el hardware del sistema y la distribución del software en los diferentes procesadores. Esta vista es útil para los ingenieros de sistemas que planean la implementación del sistema.

Hofmeister y sus colaboradores (2000) sugieren el uso de vistas similares, pero a estas le agrega una vista conceptual abstracta del sistema, que puede ser la base para descomponer los requerimientos de alto nivel en especificaciones más detalladas y para tomar decisiones sobre componentes que pueden reutilizarse.

### 3.4 Patrones arquitectónicos.

Los patrones arquitectónicos se propusieron en la década de 1990, con el nombre de "Estilos arquitectónicos" (Shaw y Garlan, 1996), Un patrón arquitectónico es una descripción de los elementos y tipos de relaciones junto con un conjunto de restricciones sobre la forma en que se pueden utilizar.

### 3.4.1 Arquitectura en capas.

Un sistema en capas se organiza jerárquicamente, cada capa proporciona servicio a la capa de encima y actúa como cliente de la capa de abajo.

Se usa Cuando se construyen nuevas facilidades por encima de los sistemas existentes; se usa cuando el desarrollo se realiza por varios equipos de trabajo y cada equipo es responsable por una capa; se usa cuando se requiere seguridad multinivel.

Ventajas: Permite el cambio de capas completas. Aumenta la confiabilidad del sistema, cuando se incluye redundancia en cada capa, por ejemplo la autenticación de los usuarios.

Desventajas: no hay una separación limpia entre capas, una capa de nivel superior interactúa directamente con capas de nivel inferior, en vez de que sea con la capa inmediatamente inferior. Suele tener bajo rendimiento por la interpretación de una solicitud de servicio por cada capa.

### 3.4.2 Arquitectura de repositorio.

En esta arquitectura hay un repositorio central de datos, los componentes del sistema no interactuaban directamente, si no tan solo a través del repositorio. Los componentes son programas independientes que tienen acceso al repositorio por referencia directa a la memoria o a través de una llamada a un procedimiento o a una consulta de base de datos.

Esta arquitectura se usa cuando grandes volúmenes de información deban almacenarse durante mucho tiempo. También se usa cuando se deba activar una acción o herramienta cuando se actualice el repositorio.

Ventajas: Los programas son independientes, no necesitan conocer la existencia de otros componentes. Los cambios hechos por un componente se propagan hacia todos los componentes. La totalidad de los datos se pueden gestionar de manera consistente.

Desventaja: Una falla en el repositorio afecta a todo el sistema. La comunicación es ineficiente a través del repositorio. Es difícil que el repositorio sea distribuido en varios computadores.

### 3.4.3 Arquitectura cliente servidor.

Esta arquitectura son máquinas virtuales de dos capas con conexiones en red. Las máquinas virtuales son servidores que están por debajo de los clientes, cada uno de los cuales se comunica a través de llamadas a procedimientos remotos o a métodos de acceso a la red. El servidor representa un proceso que provee servicios a otros procesos (clientes). Los servidores no conocen de antemano la cantidad y la identidad de los clientes, pero los clientes si conocen la identidad de los servidores.

Esta arquitectura se usa cuando, desde varias ubicaciones se tiene que acceder a la base de datos o cuando la carga de un sistema es variable.

La principal ventaja de esta arquitectura es que los servidores se pueden distribuir a través de una red.

Desventajas: Es susceptible a ataques de rechazos de servicios o a fallas del servidor. El rendimiento depende de la red. Cuando los servidores son de diferentes organizaciones pueda que haya problemas administrativos.

### 3.4.4 Arquitectura de tubería y filtro.

Son programas independientes llamados filtros. Cada programa procesa sus entradas y produce sus salidas. Los flujos de datos se transmiten de uno a otro programa y se transforman conforme se desplazan.

El nombre de “tubería y filtro” proviene del sistema Unix, donde era posible vincular procesos empleando “tuberías”. Por ellas pasaba una secuencia de texto de un proceso a otro.

Se suelen utilizar en aplicaciones de procesamiento de datos tanto en Batch y en transacciones, donde las estradas se procesan en etapas separadas para generar salidas relacionadas.

Ventajas: Soporta reutilización de transformaciones. El flujo de trabajo coincide con la estructura de muchos procesos empresariales. El proceso puede ser secuencial o recurrente.

Desventajas: La carga del sistema aumenta debido a que cada componente debe analizar y transformar los datos de acuerdo a un formato establecido, para la transferencia de los datos.



### 3.5 Arquitectura orientada a servicios SOA.

SOA es una arquitectura basada en servicios web reutilizables con interfaces públicas, donde los proveedores de servicios y consumidores interactúan en forma desacoplada para realizar los procesos de negocio.

La arquitectura SOA se basa principalmente en servicios web que proveen la lógica del negocio y los datos. Utiliza el estándar SOAP (Simple Object Access Protocol) para comunicarse con objetos remotos mediante el intercambio de mensajes en formato XML.

Los servicios tienen que tener una interfaz o contrato muy bien definido. Un contrato consiste en la especificación completa de un servicio entre un proveedor y un cliente. Este contrato debe identificar que operaciones va a realizar el servicio, definir qué datos serán necesarios para su intercambio y una forma detallada de como invocar el servicio.

Los siguientes son las tecnologías de una plataforma SOA:

- Gestión de procesos de negocio. BPM: Su objetivo es gestionar y orquestar los procesos de una organización (modelar, integrar, monitorizar y optimizar). Estos procesos orquestados y gestionados pueden utilizar los servicios definidos y gestionados por SOA. La comunicación entre SOA y BPM se realiza por medio del bus de servicios de empresa ESB. Algunas herramientas para BPM son: JBoss jBPM, Apache ODE.
- Bus de servicio de empresa. ESB: Es una aplicación middleware cuya función es proporcionar la interoperabilidad entre distintos protocolos de comunicación. También realiza el procesamiento y coreografía del flujo de datos, clasificación y conmutación de errores, transformaciones de un formato XML a otro. Algunas herramientas para ESB son: JBoss ESB, OpenESB.
- Registro de servicios y su publicación: La función es registrar las especificaciones (contrato) que claramente identifique los parámetros de entrada, de salida, los posibles fallos y las operaciones disponibles de los servicios web. Algunas herramientas de registro son: WSO2, OpenLDAP, ApacheDS.
- Gestión de la toma de decisiones empresariales. EDM: La función es gestionar y definir cómo funciona el negocio a través de sentencias escritas en forma sencilla y comprensible. Estas sentencias son llamadas reglas de negocio, que son ejecutadas por medio del motor BRE (Business Rule Engine). Algunas herramientas para EDM son: JBoss Rules (Drools), Mandarax, openLexicon.
- Procesador de flujo de eventos. ESP: La función de un ESP es recibir en tiempo real la información de las bases de datos, de las aplicaciones, de los

sistemas basados en mensajes y en general flujo de datos en tiempo real para analizarlos y detectar patrones entre ellos. Los resultados arrojados por este análisis sirve para alimentar la monitorización de la actividad del negocio con el fin de detectar oportunidades o anomalías. Algunas herramientas para ESP son: tecnología ESP/CEP (Event Stream / Complex Event Processing) y el proyecto Esper.

- Componentes de servicio y su composición. SCA: Define un mecanismo para crear componentes de servicio que son las piezas fundamentales para SOA. Un servicio puede construirse mediante una función de inteligencia de negocio de forma que combine parte de datos y parte de lógica. Se pueden crear servicios de grano fino que son aquellos que cumplen una sola funcionalidad. También se pueden crear componentes de grano grueso que son servicios compuestos por varios servicios. Un ejemplo de una herramienta para SCA es Apache Tuscany.
- Mediación de servicios web. WSM: Un mediador de servicios utiliza perfiles de seguridad. Controla las diferentes versiones de las API. Gestiona el tráfico dependiendo de cada perfil. Traduce mensajes de cualquier protocolo a SOAP para su uso interno. Ejemplo de herramienta para SCA es Apache Synapse.

### 3.6 Eventos sísmicos.

Un evento sísmico es un movimiento caótico de la corteza terrestre, con direcciones y amplitudes dependientes del tiempo. Un terremoto empieza con un choque que se produce a cierta profundidad bajo la corteza terrestre, en un punto llamado hipocentro (Buffa, 2007).

Históricamente los sismos eran atribuidos a las acciones de los dioses o a la lucha entre deidades maléficas y protectoras. Científicamente también intentaron explicar estos fenómenos como los filósofos presocráticos y a Aristóteles que consideraban que los terremotos eran producidos por masas de aire caliente que intentaban escapar del interior de la tierra. (Sarachaga, 2008).

La ciencia que estudia los aspectos relacionados con la ocurrencia de temblores de tierra, terremotos o sismos se denomina sismología. La sismología ha ayudado a definir la estructura interna de la tierra. (Castro & Jiménez Jiménez, 1986).

Según las causas que generan los sismos, estos se pueden clasificar en las siguientes clases:

- Sismos de colapso: son pequeños terremotos que ocurren debido al hundimiento de cavidades subterráneas o minas.

- Sismos volcánicos: Las explosiones de los gases durante las erupciones volcánicas pueden producir sismos que en general afectan a superficies pequeñas y tienen poca intensidad.
- Sismos tectónicos: Son los más fuertes y los más frecuentes. Su causa es la rotura de las capas de roca formando las fallas geológicas.
- Terremotos interplaca: La ruptura súbita de la corteza terrestre sucede en los bordes de las placas.

### 3.7 Ondas sísmicas.

La ruptura de la roca libera energía. De esta manera, la tierra es puesta en vibración. Esta vibración es debido a la propagación de las ondas sísmicas que son registradas por sismógrafos. Estos son instrumentos que registran tres componentes del movimiento del terreno: dos horizontales y uno vertical. Los sismógrafos se pueden diseñar para que registren la aceleración, la velocidad o el desplazamiento del terreno. Los más utilizados son aquellos que registran la aceleración, llamados acelerógrafos. (Castro & Jiménez Jiménez, 1986).

Las ondas sísmicas tienen su origen en el foco del sismo y se propagan a través de las capas de la tierra hasta la superficie.

En un medio infinito, homogéneo, sólido, elástico e isotrópico, solo se pueden propagar dos tipos de ondas: ondas P (Primarias, puesto que llegan primero y ondas S (Secundarias).

Las ondas P se desplazan a través de sólidos, líquidos, o gas a diferentes velocidades según el medio. En una roca granítica la onda recobra su volumen original más rápidamente que en un líquido o gas. De aquí que las ondas P se aceleran cuando pasan a través de sólidos de mayor densidad y se desaceleran cuando viajan a través de materiales líquidos o gaseosos.

Las ondas S desaparecen totalmente al entrar en los líquidos. Estas ondas S se mueven a través del material mediante un movimiento de cizalla que es una deformación en la que se desplaza lateral y temporalmente, después de que pasa la onda el material regresa a su posición original. (Valdivia, Plaza, & Valdés, 1996).

### 3.8 Componentes de una estación sismológica.

Una estación sismológica registra eventos sísmicos a través de instrumento o dispositivos electrónicos muy sensibles. Consta de los siguientes dispositivos electrónicos:

- **Sensor, sismómetro, acelerógrafos:** Registra el movimiento de la tierra (ondas sísmicas) y las convierte en señales eléctricas (análogas) estas señales análogas son enviadas al digitalizador. Hoy en día existen muchos sismógrafos en todo el planeta para poder registrar las ondas que se propagan por la tierra.
- **Digitalizador:** Convierte las señales análogas que fueron enviadas por el sismómetro a formato digital, es decir convierte el voltaje eléctrico a binario y envía esta información al sistema de comunicaciones.
- **Modem:** Hace parte del sistema de comunicación, modula la información que recibe del digitalizador para poderla enviar vía satélite.
- **Antena parabólica:** Transmite vía satélite la información modulada y llega al centro de adquisición de datos por medio de una antena parabólica y de allí la información es almacenada en servidores.

## 4 ESCENARIOS PARA ATRIBUTOS DE CALIDAD Y TÁCTICAS.

Un escenario para atributos de calidad es un requerimiento específico para un atributo de calidad. Los atributos de calidad ayudan a tomar decisiones arquitectónicas, por esta razón se deben tener en cuenta desde el momento en que se genera la arquitectura de un sistema.

Al implantar los atributos de calidad en un sistema, surge el problema de que ellos tienden a solaparse unos con otros. Otro problema es que cuando un atributo de calidad tiene éxito es posible que afecte positivamente o negativamente en el logro de otros atributos de calidad.

Los escenarios ayudan a solucionar estos problemas, y son la caracterización de los atributos de calidad. Podemos decir entonces que los atributos de calidad son una colección de escenarios generales y que para un sistema específico, estos escenarios generales se convierten en escenarios concretos que serán los requisitos de calidad para el sistema a diseñar.

Los escenarios constan de seis partes:

- Fuente del estímulo: Puede ser un usuario, una fuente automática u otro sistema.
- Estímulo: Es una condición que debe ser considerada cuando llega a un sistema.
- Medio ambiente: El estímulo se produce dentro de ciertas condiciones. El sistema puede estar en una condición de sobrecarga o puede estar en ejecución cuando se produce el estímulo, o alguna otra condición puede ser cierto.
- Artefacto: Algunos artefactos se estimulan. Esto puede ser todo el sistema o algunas piezas de la misma.
- Respuesta: La respuesta es la actividad que se realice después de la llegada del estímulo.
- Medida de la respuesta: Cada vez que ocurre una respuesta, ésta debe ser medida con objeto de evaluar si el requisito ha sido alcanzado.

Los atributos de calidad sirven para evaluar la arquitectura de un sistema, esta evaluación consiste en detectar a tiempo los posibles errores en la arquitectura propuesta, con el propósito de modificarla antes de comenzar el desarrollo del sistema.

Un escenario es la definición de un requisito específico de un atributo de calidad, que permitirá poder evaluar el sistema para observarlo si lo cumple o no. En el caso que la evaluación sea negativa, habría que adoptar ciertas decisiones para

conseguir los objetivos deseados. Estas decisiones son llamadas tácticas y que se pueden representar de acuerdo con la propuesta de Bass, Clements, y Kazman como aparece en la ilustración No. 1. El conjunto de tácticas es lo que se llama estrategia arquitectónica.

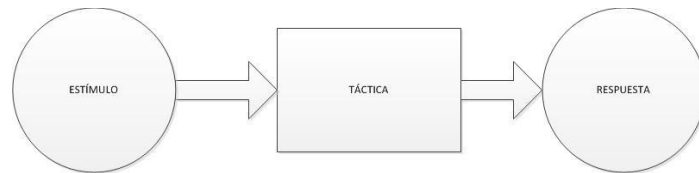


Ilustración 1. Representación de tácticas.

A continuación se describe los atributos de calidad con sus respectivas tácticas según Bass, Clements, y Kazman.

- **Atributo de calidad de Disponibilidad:** Es la posibilidad de que se produzcan fallas en un sistema. Una falla ocurre cuando el sistema deja de entregar su servicio. Las siguientes son las tácticas utilizadas:
  - Detección de fallas: Monitoreo de actividades y reporte de fallas.
    - Ping/eco: Un componente envía una petición (Ping) y recibe una respuesta (eco).
    - Heartbeat: se emite un mensaje periódicamente a un componente, si este mensaje falla se asume que el origen a fallado.
    - Excepciones: Estas son activadas cuando ocurre una falla.
  - Recuperación de fallas: Recuperación y reparación del sistema.
    - Voting: se usa para corregir fallos en el funcionamiento de los algoritmos o de un procesador.
    - Redundancia activa: Todos los componentes redundantes responden a los eventos al mismo tiempo, pero se utiliza el valor del primer componente.
    - Redundancia pasiva: El primer componente responde a los eventos y después informe a los otros componentes
    - Spare: Hay una plataforma configurada para reemplazar al componente que falle.
    - Shadow: Se coloca el componente que ha fallado o que se ha actualizado en la sombra para comprobar su funcionamiento antes de colocarlo en funcionamiento
    - Re-sincronización de estados: Actualizar el estado de un componente antes de restaurar el servicio.
    - Checkpoint/Rollback: Se utilizan puntos de control para poder devolver el sistema cuando falle al punto más consistente.
  - Prevención de fallas.

- Supresión del servicio: Esta táctica elimina el componente que fallo, para evitar fallos potenciales al sistema.
  - Transacciones: Las transacciones se deshacen en caso que el sistema falle.
  - Monitorización del proceso: El proceso de monitoreo elimina el componente que fallo y crea una nueva instancia del mismo componente en un estado apropiado.
- **Atributo de calidad de modificabilidad:** Se refiere a la facilidad que tiene el sistema de adaptarse a los cambios de funcionalidad, a los cambios de la plataforma y a los cambios en la comunicación con otros sistemas. Tiene que ver mucho con el coste del cambio. Las tácticas que se emplean son las siguientes:
  - Localizar modificaciones
    - Mantener la coherencia semántica: Son las relaciones entre las responsabilidades dentro de un módulo. Se debe asegurar que estas responsabilidades no dependan excesivamente de otros módulos. Una táctica es la abstracción de servicios comunes en módulos especializados.
    - Anticipar los posibles cambios: Esta táctica ayuda a minimizar los cambios en el sistema y se utiliza junto con la táctica de mantener la coherencia semántica.
    - Generalizar el módulo: Un módulo generalizado permite más funciones y será más fácil de ajustarlo.
    - Limitar las posibles opciones: Restringir las posibles opciones reducirá el efecto de estas modificaciones.
  - Prevenir el efecto de onda: Es la necesidad de hacer cambios en módulos no directamente afectados por la modificación. Por ejemplo, si el módulo A se cambia para llevar a cabo una modificación particular, entonces el módulo B sólo se cambia debido al cambio en el módulo A. El B se tiene que modificar porque depende, en algún sentido, en A.
    - Ocultar información: Es la descomposición de las responsabilidades de un sistema en partes más pequeñas, se debe seleccionar que partes serán públicas o privadas, con el fin de aislar los cambios.
    - Mantener las interfaces existentes: Si A depende de la interface de B y B se modifica pero sin modificar su interfaz no es necesario cambiar A.
    - Limitar las rutas de comunicación: Restringir los módulos con los cuales el modulo central se comunica. Esto reducirá el efecto onda.
    - Usar un intermediario: Si el módulo A depende del módulo B pero la dependencia nos es semántica, se puede usar un

- módulo intermedio entre A y B para que gestione las actividades asociadas con la dependencia.
- Deferir el tiempo de entrega: Deferir el tiempo para realizar el cambio. Este cambio lo puede realizar el desarrollador lo cual lo probara y lo instalara o el cambio lo puede hacer el usuario final en tiempo de ejecución. Para esto el sistema debe estar diseñado para ello utilizando algunas de las siguientes tácticas: Registro en tiempo de ejecución (funcionamiento plug-and-play), Archivos de configuración, polimorfismo, reemplazo de componentes y protocolos definidos.
  - **Atributo de calidad de rendimiento:** Se refiere a la respuesta del sistema sobre la ocurrencia de eventos como el envío de mensajes, solicitudes de usuarios, el paso del tiempo y otros. Esta respuesta se puede dar en velocidad, en asignación de recursos, en precisión, y más que todo en un tiempo determinado. Hay dos factores que contribuyen al tiempo de respuesta a un evento, el primero es el consumo de recursos como los procesadores, base de datos, redes de comunicación, la memoria entre otros. El segundo factor es el tiempo de bloqueo que se puede dar por la disputa de los recursos, disponibilidad de recursos y procesos de otros sistemas. La latencia es el tiempo entre la ocurrencia de un evento y la respuesta al mismo. Las siguientes son las categorías de las tácticas para mejorar la arquitectura según el rendimiento:
    - Demanda de recursos:
      - Aumentar la eficiencia computacional: Mejoramiento de algoritmos, cambio de componentes.
      - Reducir la sobre carga computacional: Los intermediarios aumentan la demanda de recursos lo cual es necesario encontrar el equilibrio entre las tácticas de modificabilidad y de rendimiento.
      - Reducir el número de eventos procesados: Esto se puede hacer gestionando las tasas de eventos y controlando la frecuencia de muestreo de llegada de los eventos.
      - Control del uso de recursos: Se logra poniendo tiempos de ejecución y controlando el número de eventos en cola.
    - Gestión de recursos:
      - Fomentar la concurrencia: Ejecutar varios procesos simultáneamente.
      - Mantener varias copias de los datos: Reduce la disputa de los recursos.
      - Incrementar la disponibilidad de los recursos: Utilizar procesadores más rápidos, adicionar procesadores, adicionar más memoria



- Arbitraje de recursos: Se debe planear los recursos para reducir la disputa. Para la planificación de los recursos se utilizan las siguientes políticas:
  - FIFO: Primeros que entran primeros que salen. Las peticiones tienen la misma prioridad y se asignan por turnos.
  - Prioridad Fija: Asigna prioridades a cada petición y de acuerdo al orden de prioridad se asignan los recursos. Las estrategias para asignar prioridades son por la importancia semántica, tiempo máximo de ejecución y periodo de ejecución.
  - Prioridad dinámica.
    - Round robin: Ordena las peticiones y les asigna la prioridad.
    - Earliest Deadline First: Asigna las prioridades de acuerdo a las peticiones pendientes.
  - Planificación estática: La asignación de recursos se hace de una forma autónoma.
  
- **Atributo de calidad de seguridad:** Es la capacidad del sistema para resistir el uso no autorizado mientras se continúa proporcionando los servicios a usuarios legítimos. La violación de la seguridad puede ser por un ataque, acceso no autorizado a datos o servicios, o negar servicios a usuarios legítimos. La seguridad de un sistema se puede caracterizar por las siguientes propiedades:
  - No rechazo: La transacción no puede ser negada por ninguna de las partes.
  - Confidencialidad: Los datos y servicios están protegidos contra acceso no autorizados.
  - Integridad: Los datos o los servicios han sido entregados sin modificaciones desde su origen.
  - Aseguramiento: Las partes involucradas en una transacción aseguran que son las que dicen ser.
  - Disponibilidad: El sistema estará disponible para su uso legítimo.
  - Auditoría: El sistema deja el rastro de todas las transacciones que se realizan, permitiendo después reconstruirlas.
  
- **Atributo de calidad de testeabilidad (capacidad de prueba):** Es la facilidad con que se puede hacer software para demostrar sus defectos a través de pruebas.

Para que un sistema sea debidamente comprobable, debe ser posible controlar las entradas y el estado interno de cada componente para luego observar sus salidas.

Las pruebas del software la realizan los desarrolladores, probadores, verificadores o por los usuarios. Algunas partes del código, el diseño o el software completo se puede probar.

- **Atributo de calidad de usabilidad:** La usabilidad de un sistema es la facilidad que tenga para usarse y de aprenderse. Es la facilidad y el apoyo que tiene un sistema para que un usuario realice sus tareas. Se puede dividir en las siguientes áreas:
  - Aprendizaje: ¿Qué puede hacer el sistema para facilitar el aprendizaje?
  - Minimizar el impacto de los errores: ¿Qué puede hacer el sistema para minimizar el impacto de un error?
  - Sistema eficiente: ¿Qué puede hacer el sistema para que el trabajo del usuario sea más eficiente?
  - Adaptación del sistema a las necesidades del usuario: ¿Qué se debe hacerle al sistema para adaptarlo a las necesidades del usuario?
  - Aumento de la confianza y la satisfacción: ¿Cómo se debe hacer el sistema para que el usuario tenga confianza y quede satisfecho?

El proceso de desarrollo detecta problemas de usabilidad a través de la construcción de prototipos y pruebas a estos. Los escenarios deben ser correctos para el diseño de la arquitectura para que estos errores no salgan en las pruebas del sistema.

## 5 ARCHITECTURE EXPERT DESIGN ASSISTANT (ARCHE).

### 5.1 Descripción de ArchE.

Architecture Expert (ArchE) es un asistente para el arquitecto, desarrollado por SEI (Carnegie Mellon **S**oftware **E**ngineering **I**nstitute). ArchE pretende a ayudar a generar el diseño arquitectónico que satisfaga los requisitos de calidad.

Para ArchE los atributos de calidad tales como modificabilidad, rendimiento, seguridad, usabilidad, testeabilidad y disponibilidad tienen una influencia dominante en la arquitectura de un software.

ArchE se basa en cuatro conceptos diferentes:

- Escenarios para atributos de calidad: ArchE requiere que los escenarios se especifiquen en una estructura de seis partes que implica un estímulo, una fuente del estímulo, un ambiente, un artefacto estimulado, una respuesta y una medida de respuesta.
- Responsabilidades: Las actividades dentro del sistema diseñado son representadas por responsabilidades. Mientras que el proceso de diseño progresa, las responsabilidades se añaden, se parten o se modifican para apoyar el logro de los atributos de calidad.
- Marcos de razonamiento: El conocimiento de los atributos de calidad están encapsulados en marcos de razonamiento. Un marco de razonamiento incluye conocimientos relevantes de los atributos de calidad de una arquitectura, evaluando si la arquitectura cumple con los requisitos para ese atributo de calidad y propuesta de la táctica como mecanismo de mejora.
- Tácticas arquitectónicas: Una táctica arquitectónica es un medio de satisfacer los atributos de calidad después de la manipulación de algún aspecto de la arquitectura tales como el cambio de las responsabilidades o relaciones.

Durante el proceso de diseño, el arquitecto proporciona el conocimiento del diseño y ArchE proporciona el conocimiento de los atributos de calidad. El dialogo entre el arquitecto y ArchE procede de la siguiente manera:

- Entradas del arquitecto.
  - Características que necesitan ser calculadas por el sistema que se está diseñando tales como funcionalidades y dependencias entre funcionalidades.
  - Requisitos de calidad especificados como escenarios.

- ArchE pide información necesaria para determinar el comportamiento de los atributos de calidad, tales como tiempo de ejecución para diversas funciones o costo de cambio de distintas características.
- ArchE.
  - Propone un diseño.
  - Identifica escenarios que no satisfacen.
  - Presenta una lista de tácticas que podrían mejorar el diseño.
- El arquitecto elige una o más tácticas y ArchE aplica estas tácticas y propone un nuevo diseño.
- El arquitecto tendría que proporcionar la información adicional al nuevo diseño como un nombre significativo para un servicio común o tiempo de cálculo para un intermediario.
- Este proceso se repite hasta.
  - Hasta que se cumplan los requisitos de calidad.
  - Hasta que el arquitecto quede satisfecho con el resultado.
  - Hasta que ArchE no tenga más propuestas.

ArchE incluye el conocimiento de los atributos de calidad de modificabilidad y rendimiento en tiempo real. ArchE depende del siguiente software libre para uso no comercial:

- JESS: Es un motor de reglas escrito en Java. JESS se utiliza para administrar las reglas que ArchE utiliza para el proceso de diseño y para interactuar con los marcos de razonamiento.
- MySQL: Es un sistema de gestión de base de datos. Se utiliza para almacenar información de las diferentes alternativas de diseño. También sirve para interactuar con herramientas externas.
- Eclipse: Es una plataforma de desarrollo de código abierto. ArchE usa eclipse para gestionar su interfaz de usuario.
- xmlBlaster: Servidor de comunicaciones.
- Hibernate: Framework para mapeo de objetos relacionales en la base de datos MySQL.
- GEF (Graphical Editing Framework): Es un plug-in de Eclipse para el desarrollo de editores visuales. Tiene capacidad de modelado UML.

## 5.2 Instalación de ArchE.

Se instaló la herramienta ArchE en Windows XP. Para poderlo instalar se deben instalar las aplicaciones: java runtime 5.0, eclipse-sdk-3.3-win32, mysql-essential-5.0.67-win32, GEF-runtime-3.3, xmlBlaster\_REL\_1\_6\_1, y por último ArchE\_3\_0\_0\_Installer. Ver ilustración No 2.

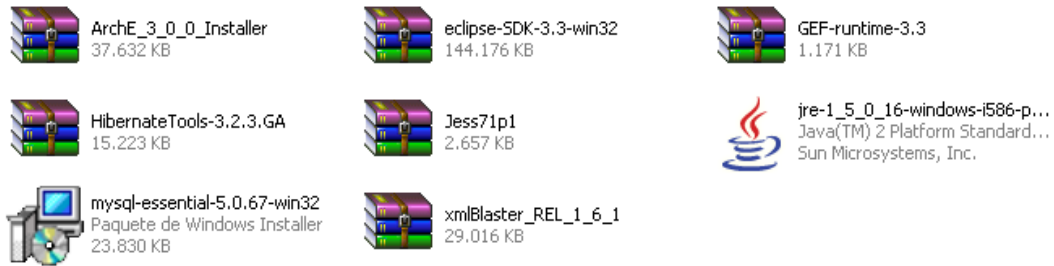


Ilustración 2. Aplicaciones para instalar ArchE.

### 5.2.1 Instalación de java runtime 5.0.

En primer lugar se debe descargar el instalador de la dirección web:

<http://www.oracle.com/technetwork/java/javasebusiness/downloads/java-archive-downloads-javase5-419410.html>.

En segundo lugar se hace doble click sobre la aplicación. Seguidamente se selecciona la opción “Instalación típica” y se oprime el botón Aceptar. Ver ilustración No 3.

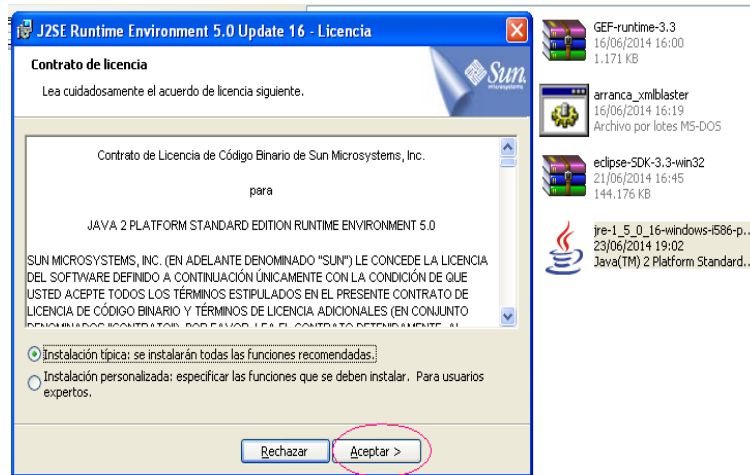


Ilustración 3. Instalación de java runtime.

La aplicación comienza a instalarse y al finalizar se debe oprimir el botón Finalizar. Ver ilustración No 4.

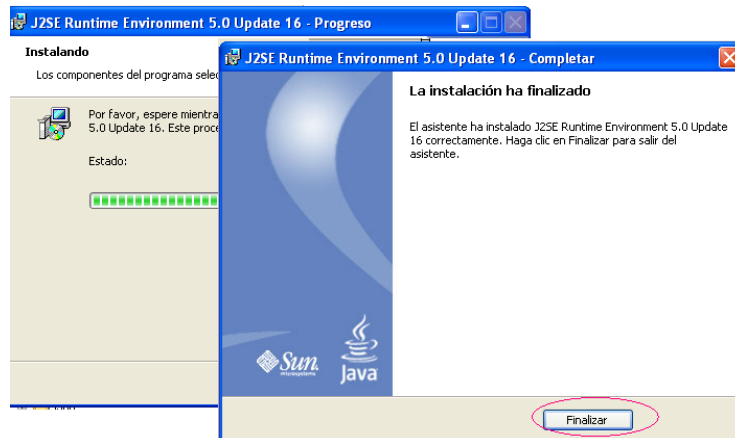


Ilustración 4. Finalización de la instalación de Java Runtime.

### 5.2.2 Instalación de eclipse-SDK-3.3-win32.

La instalación de eclipse consiste únicamente en descomprimir eclipse-SDK-3.3-win32.zip en la carpeta raíz C:\. Ver ilustración No 5.

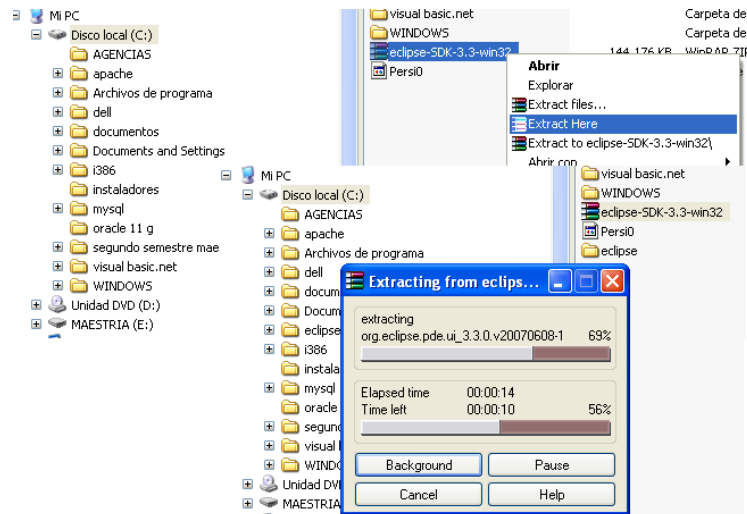


Ilustración 5. Instalación de eclipse-SDK-3.3-win32

### 5.2.3 Instalación de MySQL mysql-essential-5.0.67-win32.

Para instalar MySQL se debe hacer doble click en mysql-essential-5.0.67-win32 y oprimir el botón Next. Ver ilustración No. 6.



Ilustración 6. Inicio de la instalación de MySQL.

Se debe seleccionar la instalación typical, oprimir el botón Next y el botón Install. Ver ilustración No 7.

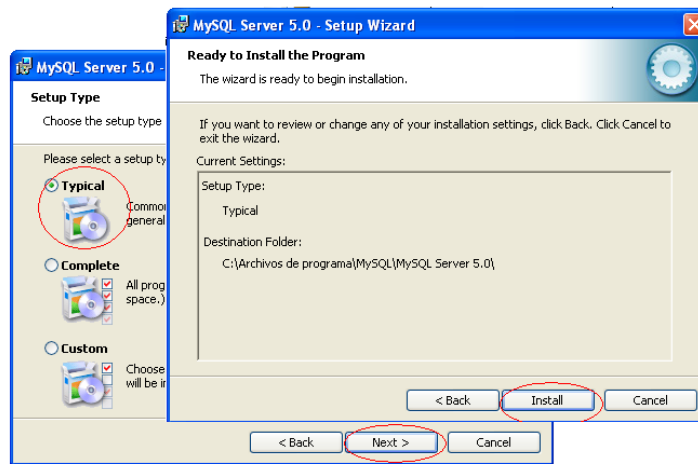


Ilustración 7. Instalación típica de MySQL.

El sistema comienza la instalación y después se debe oprimir el botón Next dos veces más. Ver ilustración No 8.

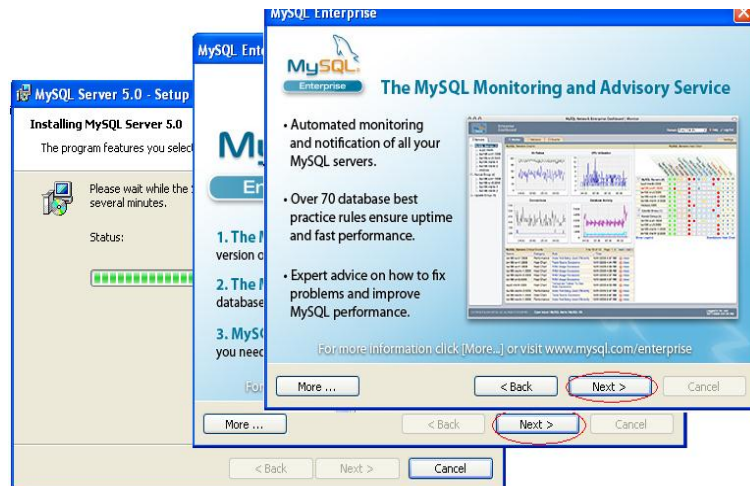


Ilustración 8. Progreso de instalación de MySQL

Antes de oprimir el botón Finish se debe asegurar que no esté seleccionada la opción “Configure the MySQL server now” ya que esta configuración se realiza cuando se instale ArchE\_3\_0\_0\_Installer. Ver ilustración No 9.



Ilustración 9. Finalización de la instalación de MySQL.

#### 5.2.4 Instalación de GEF-runtime-3.3.

Para la instalación de GEF-runtime-3.3.zip solo se descomprime desde la raíz C:\. Para que actualice la carpeta C:\eclipse. Ver ilustración 10.



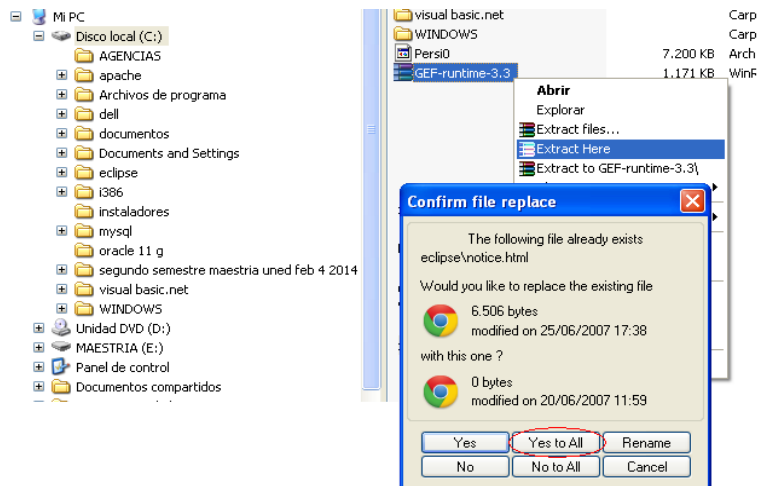


Ilustración 10. Instalación de GEF-runtime-3.3.

### 5.2.5 Instalación de Jess71p1.

Para comenzar a instalar Jess71p1 Se debe descomprimir en la raíz C:\. Ver ilustración No. 11.

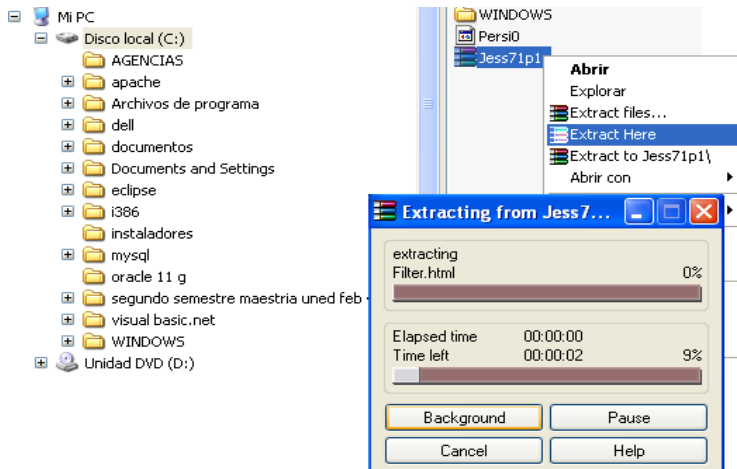


Ilustración 11. Descomprimir Jess71p1.zip

Al descomprimir Jess71p1.zip crea la carpeta Jess71p1 con la estructura que se muestra en la ilustración No 12.

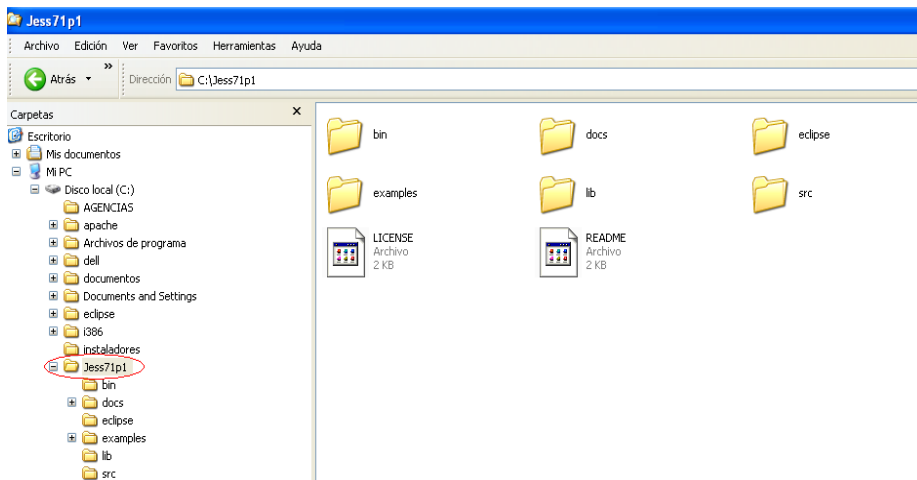


Ilustración 12. Estructura de la carpeta Jess71p1

En la carpeta C:\Jess71p1\eclipse quedan archivos comprimidos que se deben extraer. Al extraerlos se crean las carpetas plugins y features. Ver ilustración No 13.

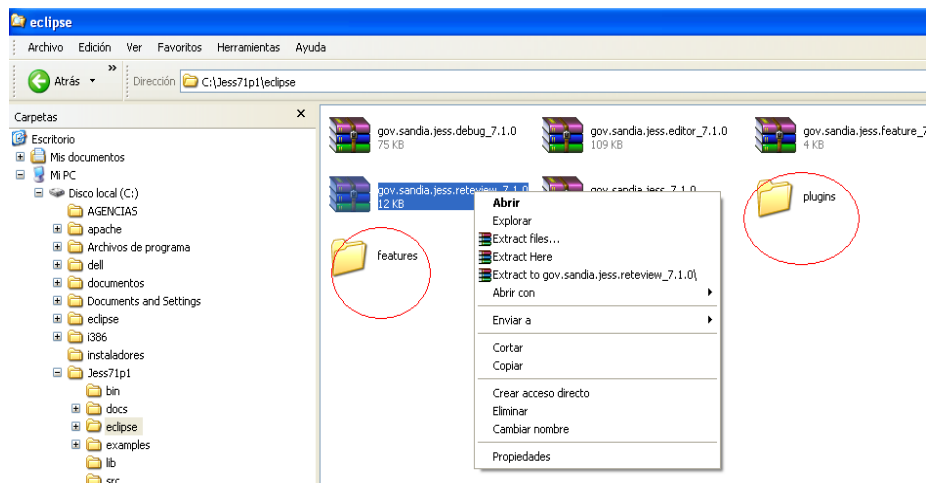


Ilustración 13. Carpetas plugins y features de Jes71p1.

Los archivos de la carpeta C:\Jess71p1\eclipse\plugins se deben copiar a la carpeta C:\eclipse\plugins. Ver ilustración No 14.

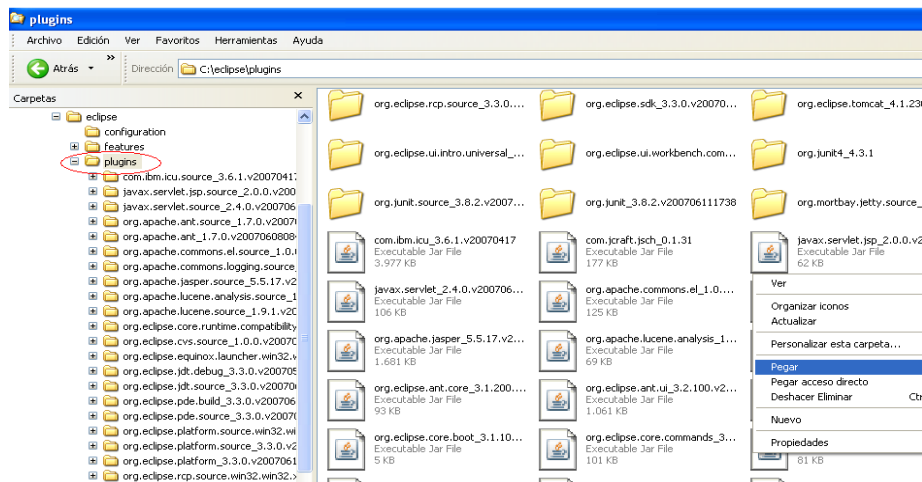
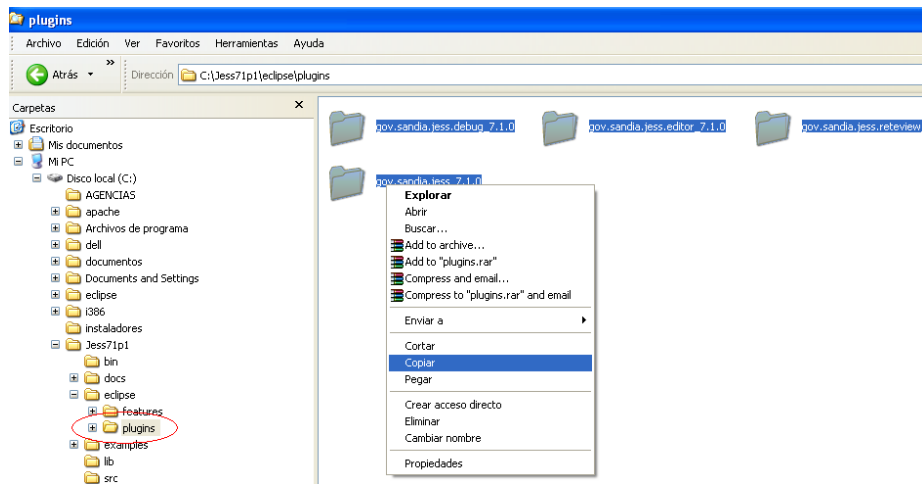


Ilustración 14. Copiado de archivos a la carpeta plugins de eclipse.

El archivo de la carpeta C:\Jess71p1\eclipse\features se deben copiar a la carpeta C:\eclipse\features. Ver ilustración No 15.

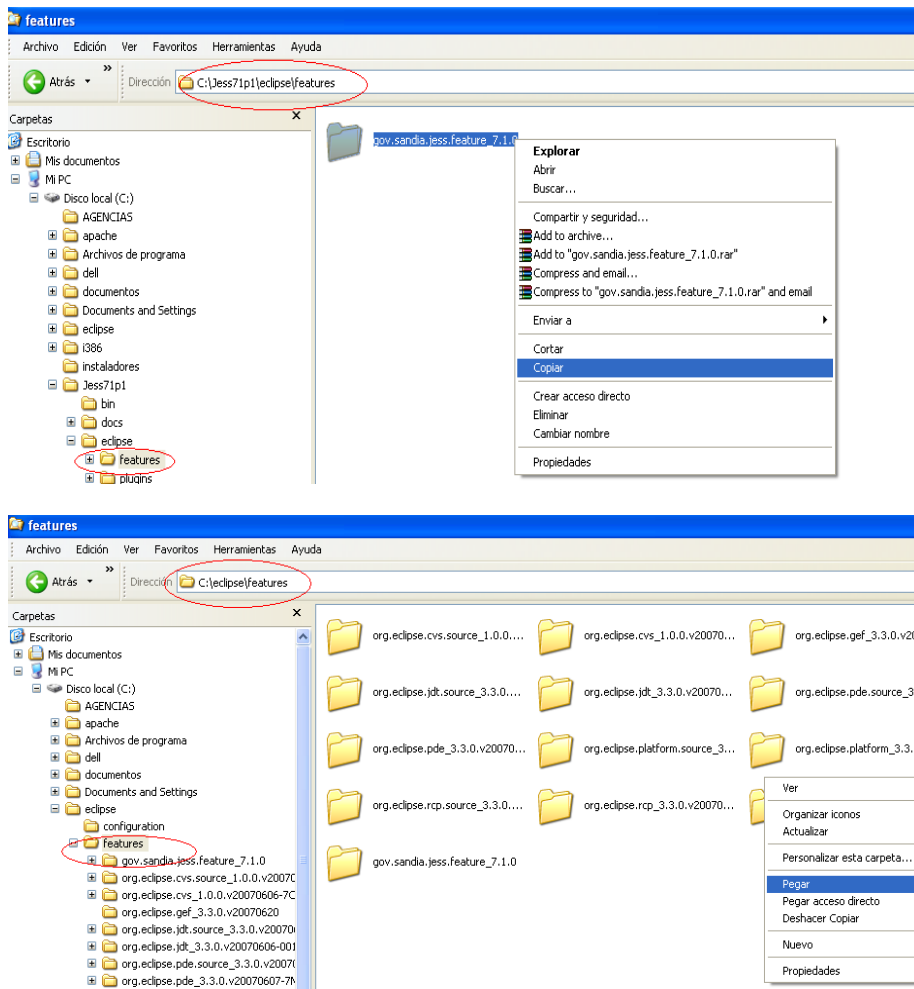


Ilustración 15. Copia de archivos a la carpeta features de eclipse.

### 5.2.6 Instalación de xmlBlaster\_REL\_1\_6\_1.

XmiBlaster es un gestor de comunicaciones entre ArchE y Mysql. Se descomprimió XmiBlaster.zip en la raíz C:\. Ver ilustración No 16.

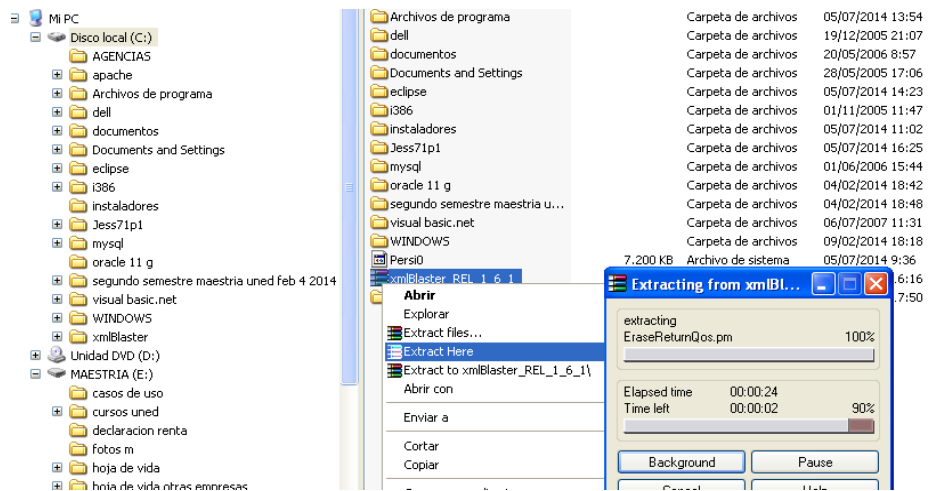


Ilustración 16. Instalación de XmlBlaster.

### 5.2.7 Instalación de ArchE 3.0

Se descomprimió ArchE\_3\_0\_0\_Installer.zip en la raíz C:\ ver ilustración No. 17.

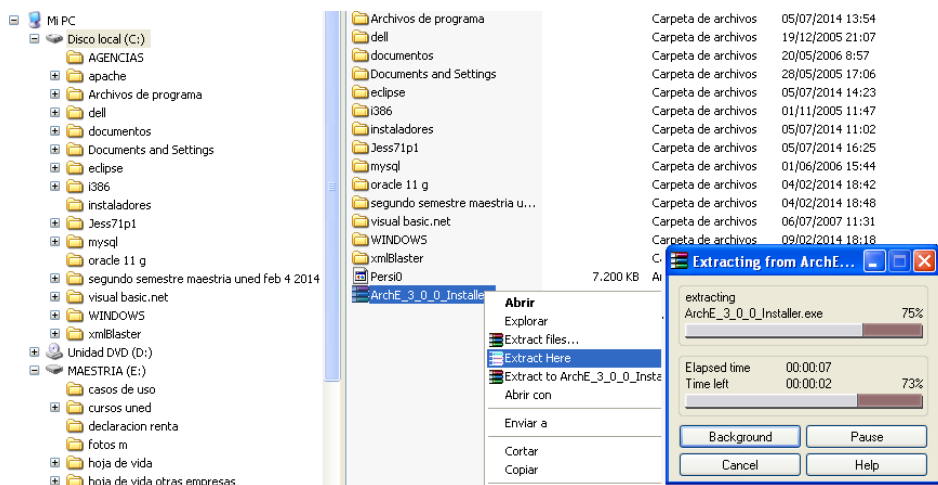


Ilustración 17. Descomprimir ArchE\_3\_0\_0\_Installer.zip.

Al ejecutar el instalador ArchE\_3\_0\_0\_Installer.exe nos muestra una interfaz donde se deberá oprimir el botón Next. Ver Ilustración No 18.



Ilustración 18. Inicio instalación de ArchE.

En seguida se despliega una interfaz en donde se deberá seleccionar el tipo de instalación “completa” para que se instalen todos los componentes de ArchE. Ver ilustración No 19.

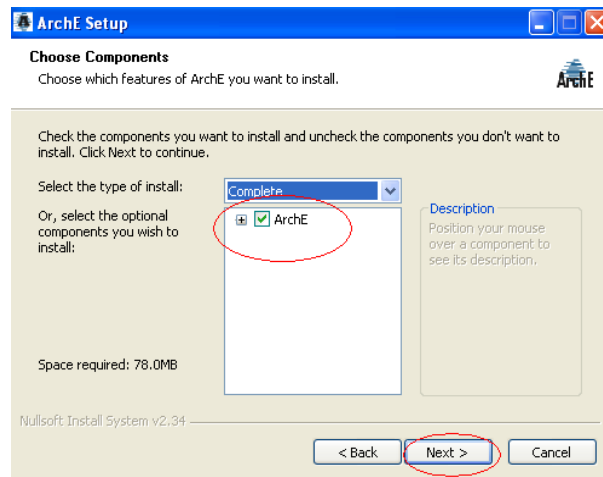


Ilustración 19. Tipo de instalación de ArchE

Al oprimir el botón Next se desplegará la interfaz para seleccionar la ubicación de la instalación. Se recomienda que la ubicación sea donde esté instalado eclipse, en este caso C:\eclipse y que por default tiene el instalador. Se debe oprimir el botón Next. Ver ilustración No. 20.

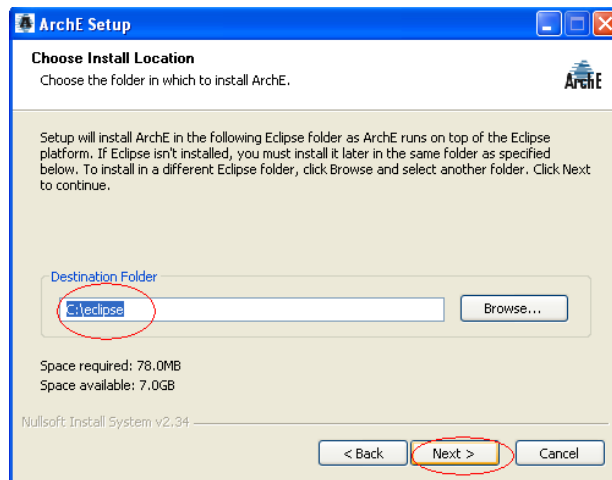


Ilustración 20. Seleccionar la ubicación de la instalación de ArchE.

En seguida se desplegará la interfaz para seleccionar la carpeta en donde quedará el icono para ejecutar la herramienta. Ver ilustración No. 21.

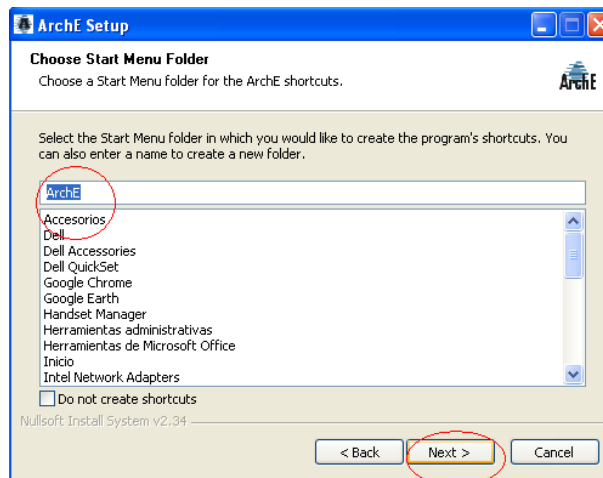


Ilustración 21. Ubicación del icono de ArchE.

Al oprimir Next se muestra una interfaz en donde se puede comprobar los paquetes que se instalaron. En este caso muestra que XmiBlaster no se ha instalado y esto sucede posiblemente por no haber definido las variables de entorno. De todas maneras el instalador deja continuar al oprimir el botón Next. Las variables de entorno se deben definir cuando se vaya a utilizar la herramienta ArchE. Ver ilustración No 22.

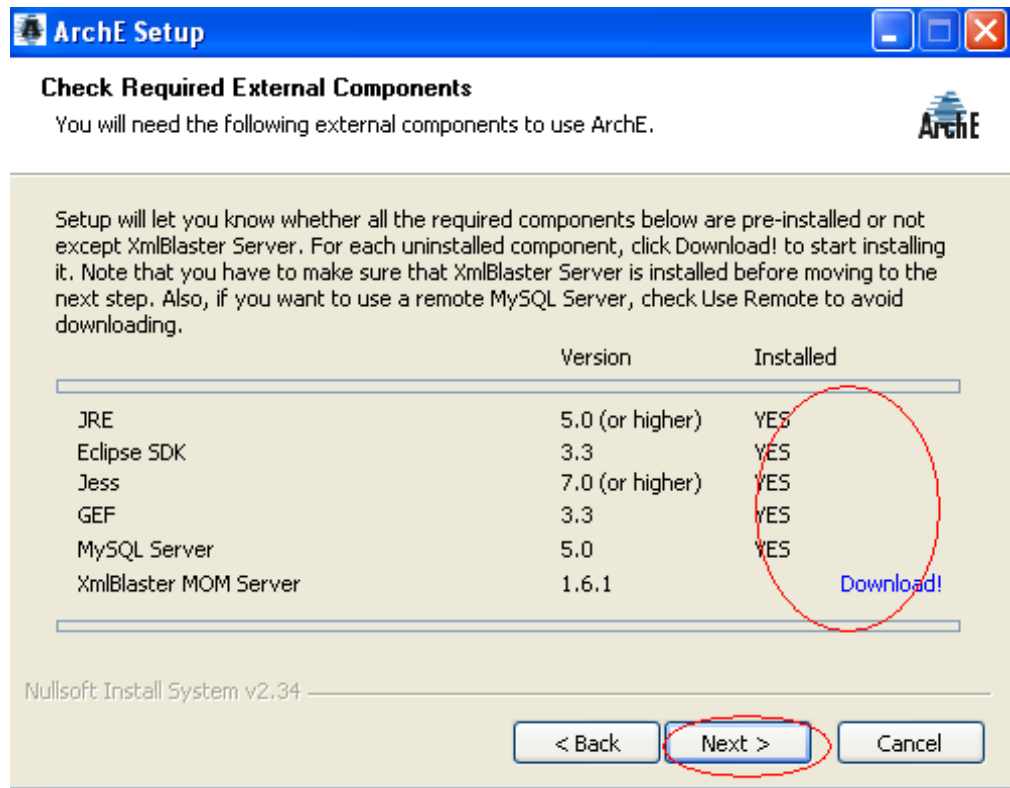


Ilustración 22. Comprobación de los paquetes que se instalaron.

Al continuar se muestra la interfaz de configuración de MySQL. En esta interfaz se deberá colocar el puerto de conexión (default el puerto 3306) y un nuevo password para el usuario root de la base de datos. Se deberá oprimir el botón “configure”. Ver ilustración No. 23.

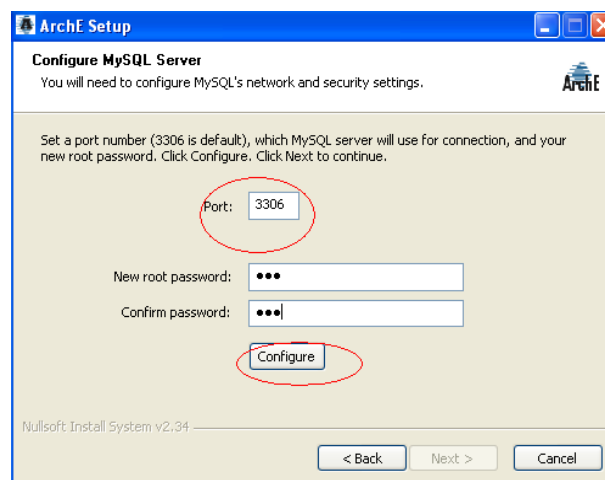


Ilustración 23. Proceso de configuración de MySQL desde ArchE



Al oprimir el botón “Configure” comienza la configuración de Mysql server. Ver ilustración No. 24.

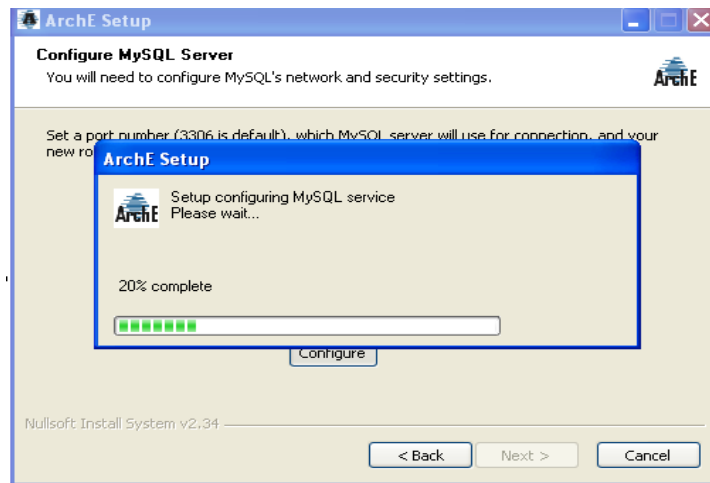


Ilustración 24. Progreso de configuración de MySQL desde ArchE.

La configuración detecta que se ha cambiado el password de root, por lo tanto muestra la advertencia que el directorio de datos no se eliminara automáticamente y que es necesario el antiguo password para conectarse. Ver Ilustración No 25.

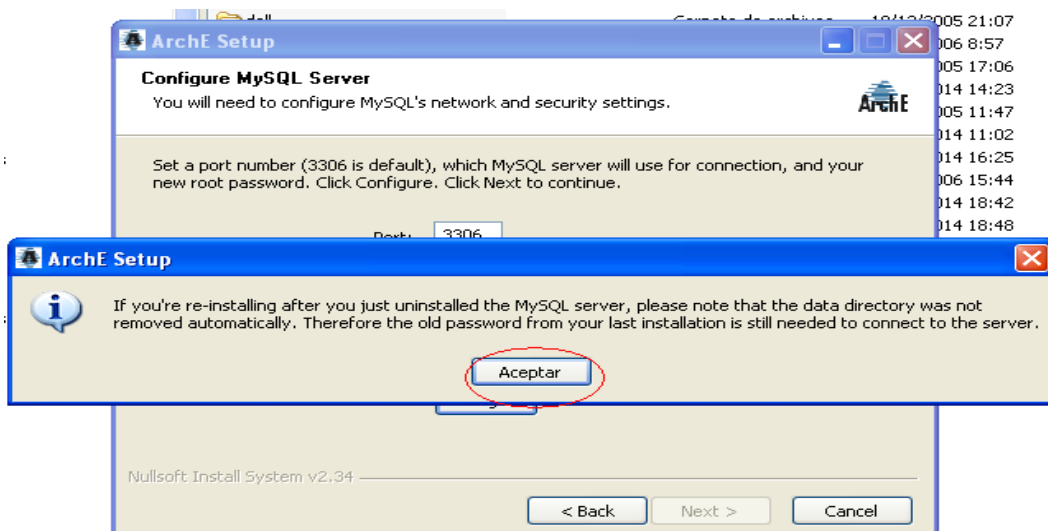


Ilustración 25. Advertencia de eliminación de la carpeta de datos.

Se debe oprimir el botón “Aceptar” y volver a oprimir el botón “Configure” ver ilustración No. 26.

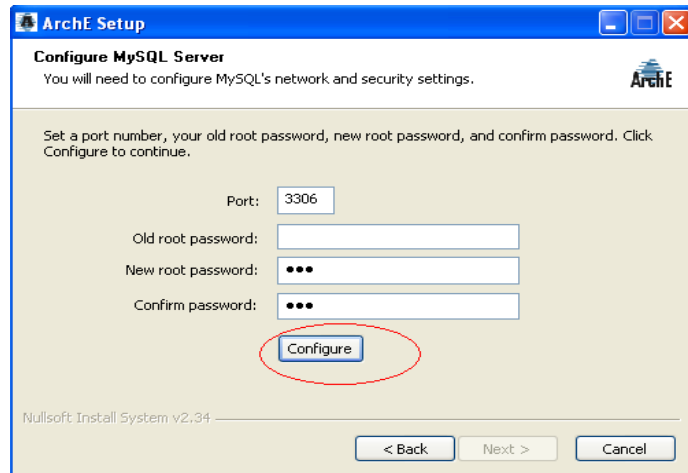


Ilustración 26. Volver a configurar MySQL.

El sistema vuelve a reconfigurar MySQL logrando terminar con éxito. Ver ilustración No 27.

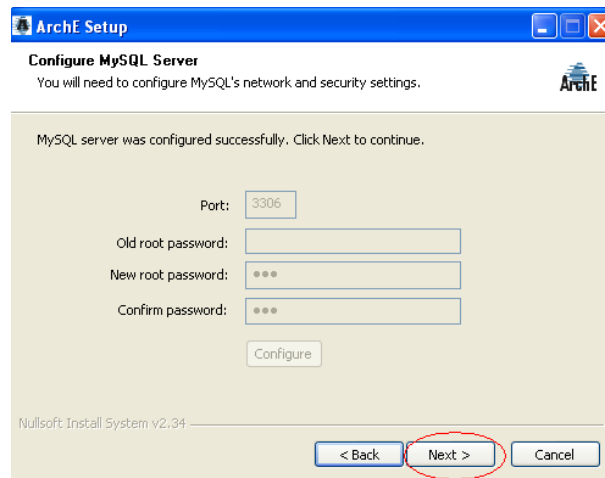


Ilustración 27. Configuración de MySQL terminada con éxito.

Al oprimir el botón "Next" se desplegará la interfaz para crear la base de datos para ArchE. Se debe digitar el password de root y oprimir el botón "Create the ArchE database". Ver ilustración No. 28.

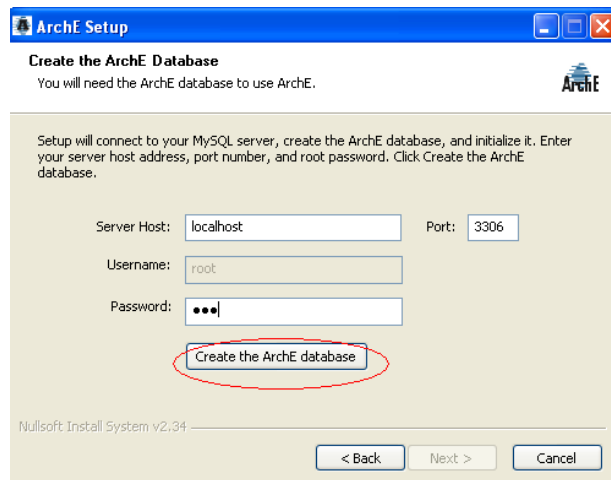


Ilustración 28. Digitación de los parámetros para crear la base de datos para ArchE.

El sistema comienza el proceso de creación de la base de datos ArchE. Para continuar con el proceso de instalación se debe oprimir el botón Next. Ver ilustración No 29.

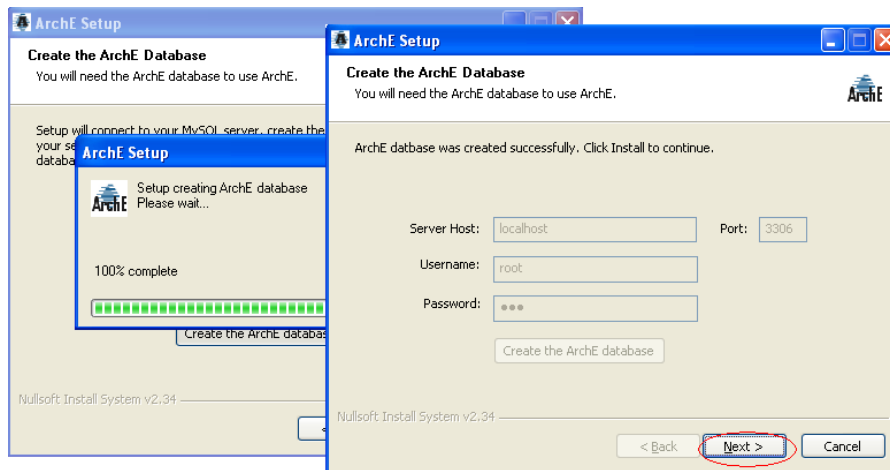


Ilustración 29. Proceso para la creación de la base de datos para ArchE.

A continuación se desplegará la interfaz para la configuración de xmlBlaster. En este caso se dejó los valores que por default tiene el instalador. Host: localhost y el puerto 7607 y se oprime el botón "Install". Ver ilustración No 30.

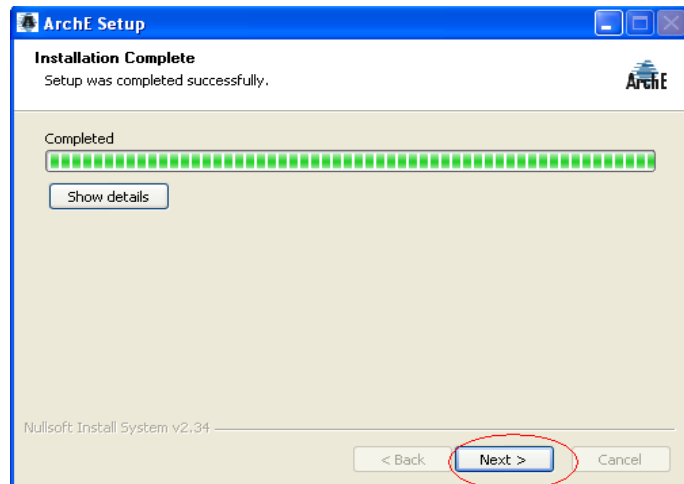
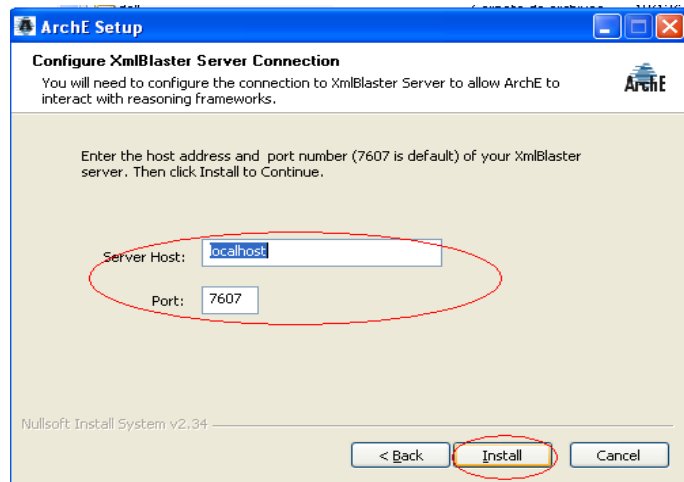


Ilustración 30. Configuración de xmBlaster.

En seguida se mostrara la interfaz de instalación completa de ArchE. Se deberá seleccionar la opción de run ArchE Antes de oprimir el botón “Finish” para ejecutar ArchE desde eclipse. Ver ilustración No. 31.

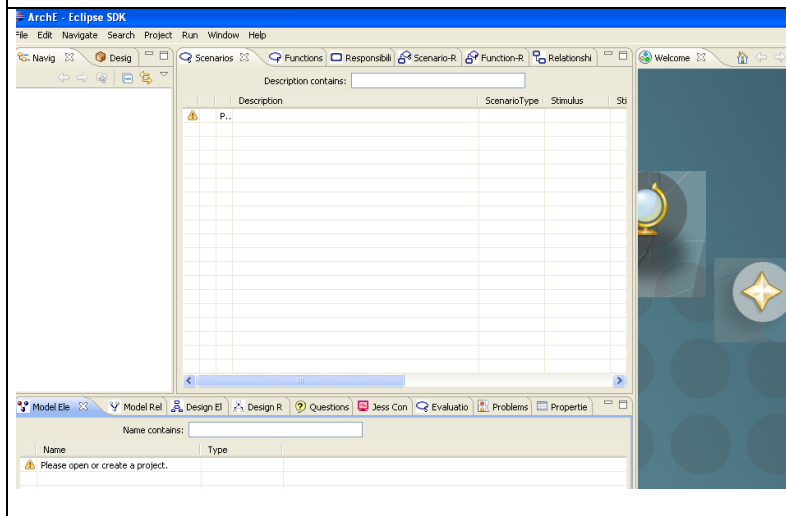
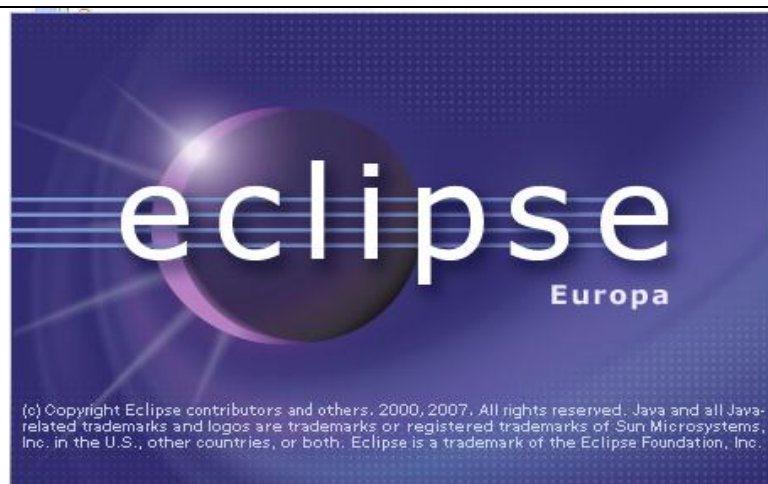
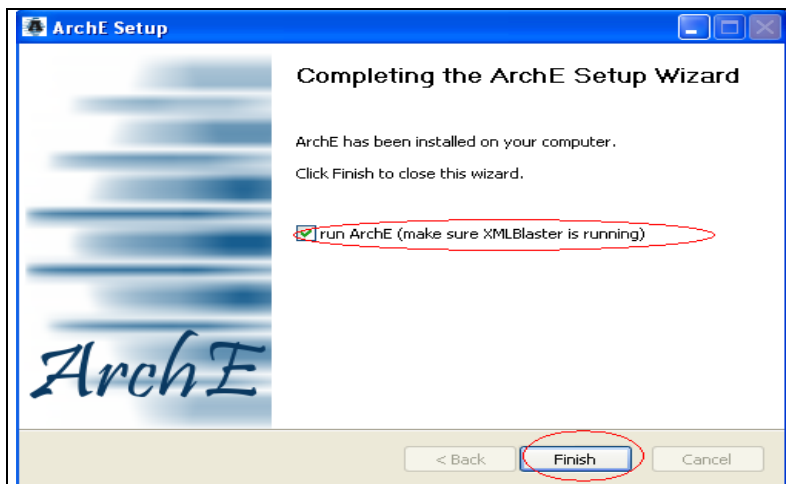


Ilustración 31. Instalación completa de ArchE

## **6 SISTEMA DE INFORMACION PARA LA PUBLICACION DE EVENTOS SISMICOS TECTONICOS DE EMERGENCIA.**

### **6.1 PLANTEAMIENTO DEL PROBLEMA.**

La Red Sismológica es el instrumento que ha implementado el gobierno para brindar información oportuna al Sistema Nacional del Riesgo, a las entidades de protección a la comunidad y a las asociaciones científicas acerca de la actividad sísmica del país. Para enfrentar en mejores condiciones las emergencias que se puedan presentar por un evento sísmico, se debe realizar en forma permanente, planeada y organizada acciones adecuadas para proporcionar información rápida y confiable sobre el evento ocurrido.

Según el protocolo de emergencia, se estipula que después de alrededor de 10 minutos después de registrado un evento sísmico, se debe tener y diseminar información confiable sobre las características del sismo. Para llevar a cabo esta labor se utilizan aplicativos de procesamiento obsoletos, los cuales no se ajustan a las condiciones actuales, se ha triplicado el número de estaciones y seguirá en incremento, y ha mejorado y ampliado los sistemas de adquisición y procesamiento; esto sumado al hecho de que el aplicativo es vulnerable a omisiones y errores por parte del analista de turno, hace necesario y justifica el desarrollo de una aplicación que permita la publicación de información relacionada con evento sísmicos, con el fin de mejorar la calidad y los tiempos de ejecución del proceso y de esta forma garantizar el suministro de información veraz y oportuna.

### **6.2 ALCANCE DEL PROYECTO.**

Especificación de requisitos y diseño técnico de un sistema de información para la publicación de eventos sísmicos tectónicos de emergencia.

### **6.3 OBJETIVO GENERAL.**

Realizar el diseño de un sistema de información para la publicación de eventos sísmicos de emergencia, basado en la arquitectura orientada a servicios SOA.

#### 6.4 OBJETIVOS ESPECIFICOS.

- Minimizar los tiempos de procesamiento y publicación de los sismos de emergencia.
- Minimizar los errores de procesamiento y publicación de los sismos y así obtener datos verídicos y confiables para los organismos de socorro
- Integrar y mejorar los procesos actuales de procesamiento y publicación de sismos de emergencia.

#### 6.5 PROCESOS DE NEGOCIO.

Cuando ocurre un evento sísmico tectónico, las estaciones sismológicas captan el movimiento de la corteza terrestre y envían la información vía satélite al sistema de adquisición de datos. Esta información es almacenada en los servidores, en las correspondientes carpetas de cada estación y en la base de datos Winston (Mysql).

El analista que es la persona encargada de localizar y publicar el sismo, extrae las trazas del sistema de adquisición de datos ejecutando la rutina "extraew". Para procesar el sismo ejecuta la rutina "mulplt" de la aplicación SEISAN, en este proceso se visualizan los sismogramas (ondas sísmicas) de las estaciones que captaron el sismo. Por cada sismograma pica o enmarca el sismo en la onda primaria (P) y secundaria (S), después ejecuta el comando L para que el sistema realice la localización del sismo, esta localización consiste en calcular la longitud, latitud, profundidad, magnitud y otros valores más, dejando esta información en un archivo plano llamado Sfile, después ejecuta el comando update de la aplicación SEISAN para confirmar la información en la base de datos de SEISAN.

SEISAN es un software para análisis de terremotos. Fue desarrollado en 1992 por los Dres.: Jens Havskov y Terje UTHEIM, de la universidad de Bergen, Noruega.

Para publicar el evento sísmico en la página web de la red sismológica y dar aviso a los organismos de socorro, el analista realiza los siguientes pasos:

- Ejecuta la rutina "publicar", esta rutina lee el archivo Sfile para subir los datos a la base de datos "sismología" (Mysql).
- Ejecuta la rutina ReadS que lee los datos de la base de datos "sismología": para calcular la hora local del sismo, para localizar el epicentro del sismo, para generar el boletín con su respectivo mapa GMT, para generar dos mapas GMT para la página web.
- Ejecuta la rutina emergencia1: para enviar los correos electrónicos, para imprimir el boletín con su respectivo mapa y para enviar el fax a los organismos de socorro. Ver Ilustración No 32.

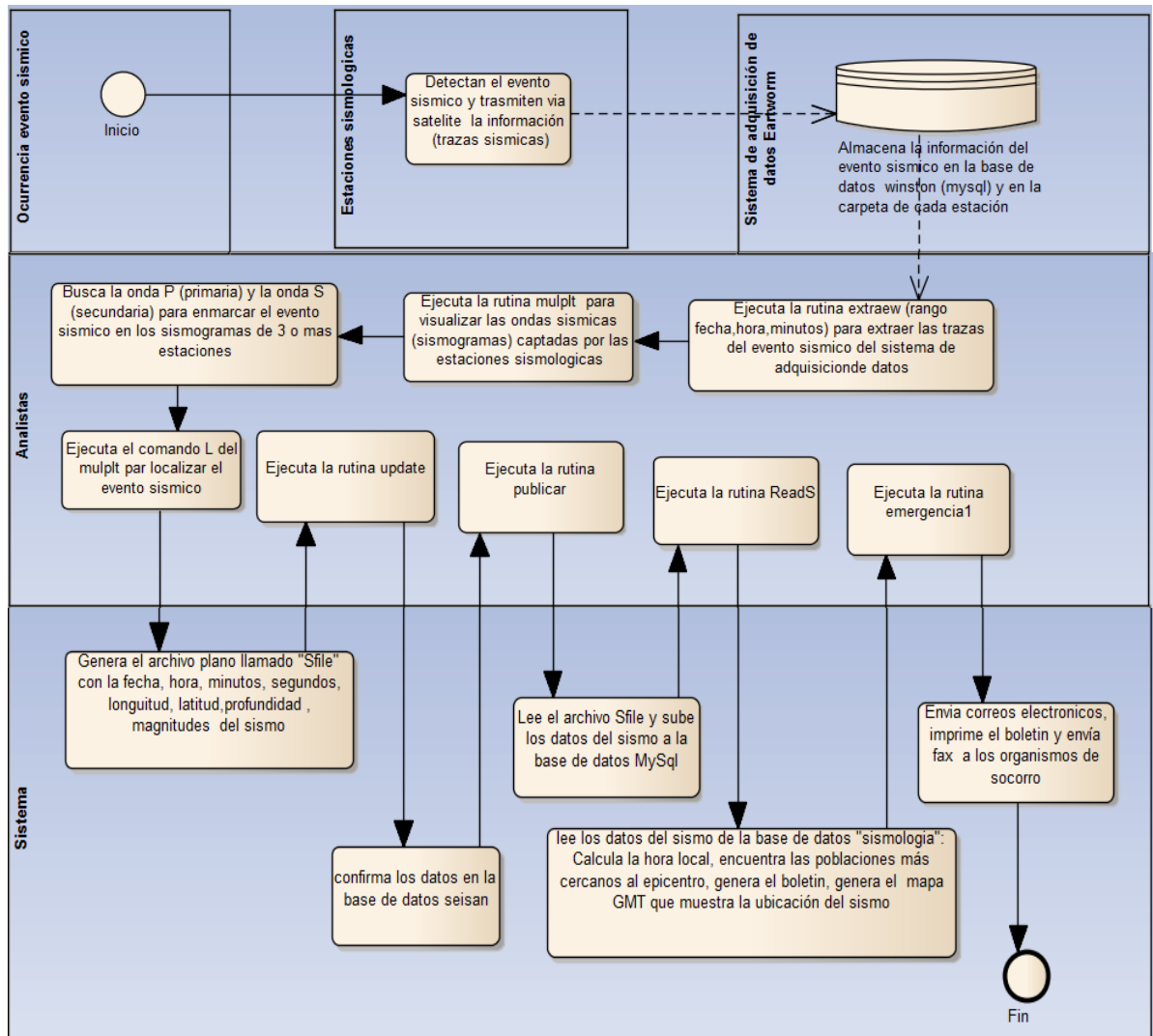


Ilustración 32. Proceso para la publicación de un evento sísmico.

## 6.6 REQUERIMIENTOS DE SOFTWARE.

- La aplicación se debe desarrollar para una arquitectura SOA.
- El software para la arquitectura SOA debe ser free (libre).
- El lenguaje de programación debe ser java.
- La aplicación debe ser web.



- Se debe utilizar la misma base de datos que actualmente está en funcionamiento.
- Se debe dejar la misma página web actual que publica las emergencias sísmicas.
- Desde la aplicación SOA se debe invocar las rutinas de la aplicación SEISAN que procesan y localizan el sismo.

## 6.7 REQUERIMIENTOS DE HARDWARE.

Se utilizara el mismo computador de procesamiento actual que es una Workstation Dell con procesador Intel E5-1620, Sistema operacional LINUX. En este computador está instalada la aplicación SEISAN y aquí deberá quedar la nueva interfaz web para el procesamiento y publicación de los sismos de emergencia.

## 6.8 REQUERIMIENTOS FUNCIONALES.

### 6.8.1 REQ01: Extracción de las trazas.

La interfaz de usuario permitirá extraer las trazas (ondas sísmicas) de los sismos en un rango de tiempo del sistema de adquisición de datos.

### 6.8.2 REQ02: Localización del sismo.

La interfaz de usuario, tras efectuar la extracción de las trazas, permitirá ejecutar la rutina mulplt para localizar el evento sísmico.

### 6.8.3 REQ03: Generar información para la publicación del sismo.

Tras efectuar la localización del sismo, la interfaz permitirá generar automáticamente la siguiente información:

- Crear los datos de la localización del sismo en la base de datos "sismología".
- Generar los mapas de la ubicación del sismo para que la página web actual los publique.
- Realizar automáticamente las llamadas telefónicas a los organismos de socorro.

- Generar e imprimir el boletín con el mapa respectivo.
- Enviar el boletín por fax a los organismos de socorro.
- Enviar automáticamente correos electrónicos a las cuentas de correo que estén inscritas.

## 6.9 IDENTIFICACION DE SERVICIOS SOA.

### 6.9.1 Servicio para localizar el epicentro del sismo.

Para localizar el epicentro del sismo (Lugar donde ocurrió el sismo), se va a utilizar el servicio "Bing Maps REST Services" de Microsoft (<http://msdn.microsoft.com/en-us/library/ff701710.aspx>). Este servicio devuelve la información asociada con una latitud y una longitud. La información que devuelve este servicio se puede ver en la Ilustración No. 33. Donde el país y la ciudad es la que nos interesa. Este servicio se ejecuta con la siguiente dirección: <http://dev.virtualearth.net/REST/v1/Locations/47.64054,-122.12934?o=xml&key=BingMapsKey> , donde 47.64054 es la latitud y -122.12934 es la longitud donde ocurrió el sismo.

```
<Response xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.microsoft.com/search/local/ws/rest/v1">
  <Copyright>
    Copyright © 2011 Microsoft and its suppliers. All rights reserved.
    This API cannot be accessed and the content and any results may not be
    used, reproduced or transmitted in any manner without express written
    permission from Microsoft Corporation.
  </Copyright>
  <BrandLogoUri>
    http://dev.virtualearth.net/Branding/logo_powered_by.png
  </BrandLogoUri>
  <StatusCode>200</StatusCode>
  <StatusDescription>OK</StatusDescription>
  <AuthenticationResultCode>ValidCredentials</AuthenticationResultCode>
  <TraceId>
    dd31ffaf098f4406b7ecdd0da36680ff
  </TraceId>
  <ResourceSets>
    <ResourceSet>
      <EstimatedTotal>1</EstimatedTotal>
      <Resources>
        <Location>
          <Name>1 Microsoft Way, Redmond, WA 98052</Name>
```

```

<Point>
  <Latitude>47.640568390488625</Latitude>
  <Longitude>-122.1293731033802</Longitude>
</Point>
<BoundingBox>
  <SouthLatitude>47.636705672917948</SouthLatitude>
  <WestLongitude>-122.137016420622</WestLongitude>
  <NorthLatitude>47.6444311080593</NorthLatitude>
  <EastLongitude>-122.1217297861384</EastLongitude>
</BoundingBox>
<EntityType>Address</EntityType>
<Address>
  <AddressLine>1 Microsoft Way</AddressLine>
  <AdminDistrict>WA</AdminDistrict>
  <AdminDistrict2>King Co.</AdminDistrict2>
  <CountryRegion>United States</CountryRegion>
  <FormattedAddress>1      Microsoft      Way,      Redmond,      WA
98052</FormattedAddress>
  <Locality>Redmond</Locality>
  <PostalCode>98052</PostalCode>
</Address>
<Confidence>Medium</Confidence>
<MatchCode>Good</MatchCode>
<GeocodePoint>
  <Latitude>47.640568390488625</Latitude>
  <Longitude>-122.1293731033802</Longitude>
  <CalculationMethod>Interpolation</CalculationMethod>
  <UsageType>Display</UsageType>
  <UsageType>Route</UsageType>
</GeocodePoint>
</Location>
</Resources>
</ResourceSet>
</ResourceSets>
</Response>

```

Ilustración 33. Información asociada a la longitud y latitud del evento sísmico.

### 6.9.2 Servicio para generar los mapas GMT.

Para generar los mapas GMT (Generic Mapping Tools) se debe crear un servicio web que reciba como parámetros la latitud, longitud, profundidad y el nombre de la

población del epicentro y debe regresar una imagen del mapa con los respectivos datos.

#### 6.9.3 Servicio para enviar correos electrónicos.

Para enviar los correos electrónicos automáticos se va a utilizar el servicio web <http://www.programmableweb.com/api/copernica> . Permite crear una base de datos de contactos, se pueden crear plantillas para envíos inteligentes de correos a todos los contactos.

#### 6.9.4 Servicio para enviar fax.

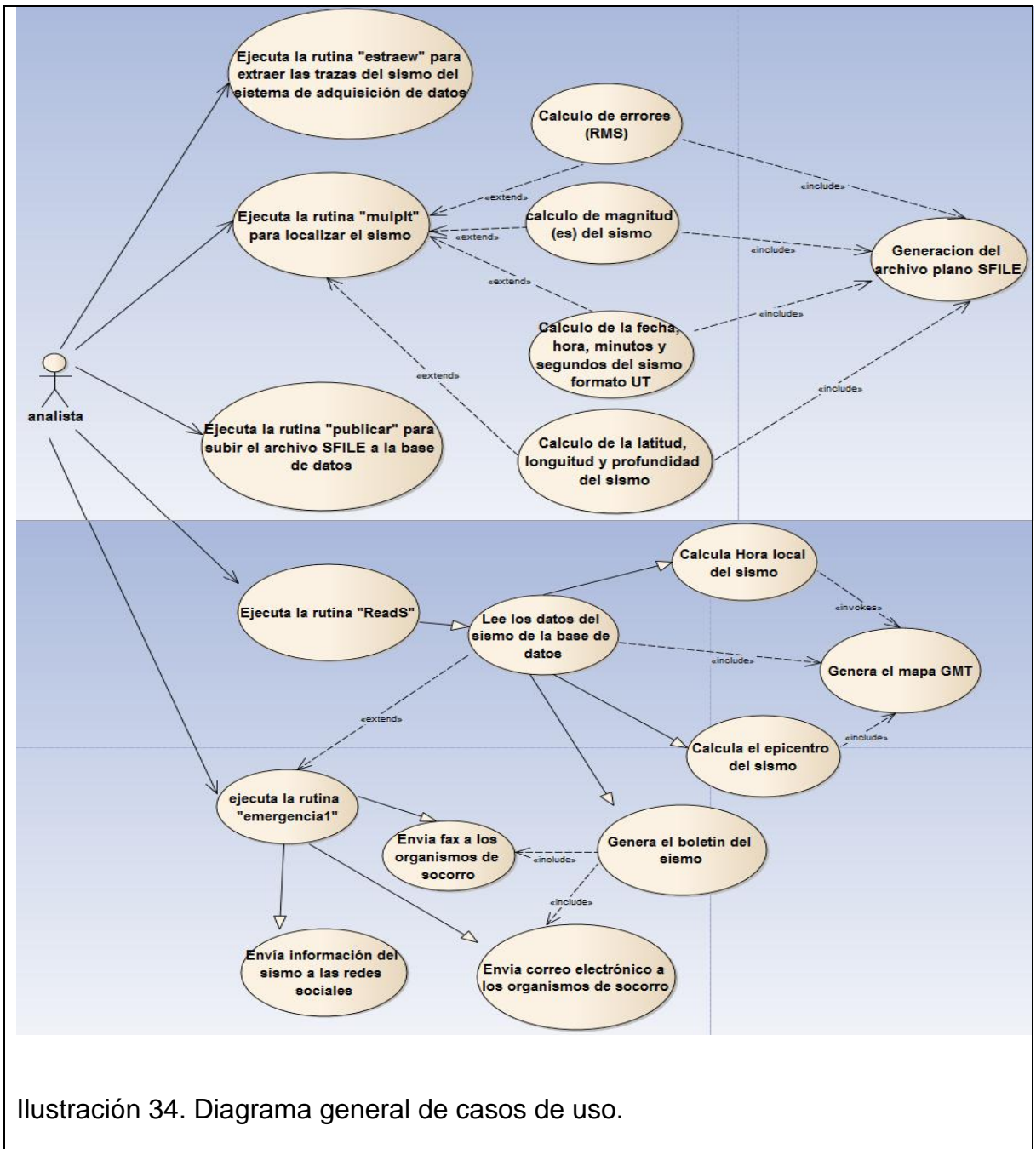
Para enviar los fax, se va a utilizar el servicio web <http://www.programmableweb.com/api/interfax>, permite la transmisión de mensajes de fax en formatos PDF, Word, HTML, etc. Sin instalar ningún tipo de hardware o software.

#### 6.9.5 Servicio para realizar llamadas telefónicas.

Para realizar las llamadas telefónicas automáticas y enviar los mensajes SMS se va a utilizar el servicio web <http://www.programmableweb.com/api/hoiio-voice>, este servicio utiliza para las llamada telefónicas el módulo LWP y para enviar los mensajes SMS utiliza bulk SMS API.

### 6.10 DIAGRAMA GENERAL DE CASOS DE USO.

En este diagrama se visualiza las funciones que realiza el analista para la publicación del evento sísmico. Ver ilustración No 34.



## 6.11 DESCRIPCION CASOS DE USO.

### 6.11.1 Nombre del caso de uso: Extraer trazas del sismo

#### 6.11.1.1 Descripción caso de uso:

Permite extraer las ondas sísmicas del sistema de adquisición de datos del evento sísmico.

#### 6.11.1.2 Requerimiento funcional asociado:

REQ01: Extracción de las trazas.

#### 6.11.1.3 Actores del sistema.

Analistas.

#### 6.11.1.4 Precondiciones.

- El sistema de adquisición de datos debe estar funcionando.
- Haber espacio suficiente en el computador de procesamiento para las trazas.
- El actor del sistema debe tener permiso para localizar y publicar sismos.

#### 6.11.1.5 Post condición.

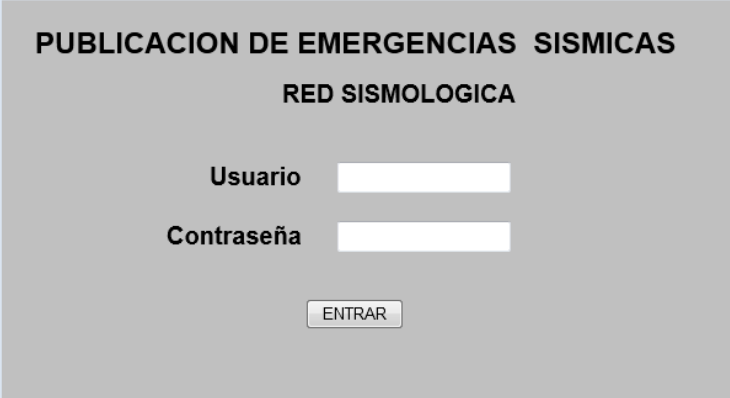
El sistema debe almacenar las trazas en la carpeta EME del computador de procesamiento.

#### 6.11.1.6 Flujo de eventos.

- El actor debe autenticarse. Ver prototipo 1. autenticación. Ver Ilustración 35.
- Seleccionar la opción “extraer trazas del sismo” de la interfaz web”.
- El sistema borrara el contenido de la carpeta EME del computador de

procesamiento.

- El sistema desplegara la ventana “extraer trazas del sismo”. Ver ilustración No 36.
- El sistema automáticamente colocara la fecha, hora, minutos y segundos del sistema, en el cuadro de texto “fecha” en formato YYYYMMDDHHMMSS. Con la opción de que el actor pueda cambiar su contenido.
- El sistema automáticamente colocara 1200 seg. en el cuadro de texto “duración”. Con la opción de que el actor cambie su contenido.
- El actor oprimirá el botón ACEPTAR.
- Al oprimir el botón ACEPTAR el sistema deberá ejecutar la rutina “extraew” enviando los parámetros fecha y duración.
- La rutina “extraew” deberá dejar las trazas en la carpeta EME del computador de procesamiento.
- Al terminar el proceso de extracción, el sistema desplegara el siguiente mensaje: “Las trazas fueron extraídas con éxito”.

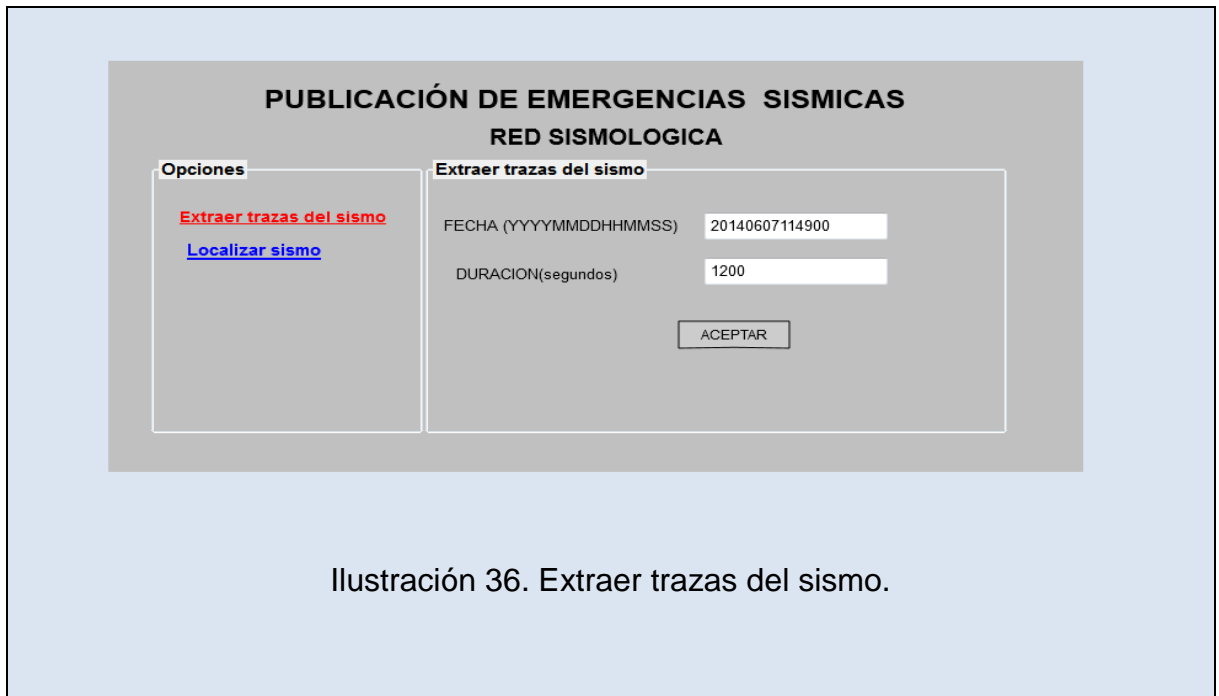


**PUBLICACION DE EMERGENCIAS SISMICAS**  
**RED SISMOLOGICA**

**Usuario**

**Contraseña**

Ilustración 35. Autenticación de usuarios



6.11.2 Nombre del caso de uso: Localización del sismo.

6.11.2.1 Descripción caso de uso:  
Ejecución de la rutina “mulplt” de la aplicación SEISAN para localizar el sismo.

6.11.2.2 Requerimiento funcional asociado:  
  
REQ02: Localización del sismo

6.11.2.3 Actores del sistema.  
Analistas.



#### 6.11.2.4 Precondiciones.

- Las trazas del sismo deben estar almacenadas en la carpeta EME del computador de procesamiento.
- El actor del sistema debe tener permiso para localizar y publicar sismos.

#### 6.11.2.5 Post condición.

En la carpeta EME del computador de procesamiento, la rutina “mulplt” deberá almacenar el archivo plano Sfile.

#### 6.11.2.6 Flujo de eventos.

- Una vez extraídas las trazas del sismo, el actor del sistema oprimirá la opción “localizar sismo” de la interfaz web. Ver Ilustración 37.
- El sistema validara si las trazas se encuentran en la carpeta EME.
- El sistema borrara el contenido de la carpeta “sismo” del computador de procesamiento.
- El Sistema ejecutara la rutina “mulplt” de la aplicación SEISAN.
- Una vez ejecutada la rutina “mulplt” el analista localizara el sismo.
- Después que el analista termine de localizar el sismo la herramienta “mulplt” dejara el archivo plano Sfile en la carpeta “sismo” del computador de procesamiento.
- El sistema deberá regresar a la interfaz web, abriendo el formulario “Datos del sismo”.
- El sistema deberá leer los datos del archivo plano Sfile y subirlos al formulario “Datos del sismo”.
- El analista revisara y analizara los datos del formulario para comprobar que todo esté bien.
- El analista tendrá la opción de cambiar cualquier dato del sismo en el formulario. Uno de los cambios importantes, es que el analista con la decisión del sismólogo de turno y en la eventualidad que el terremoto hubiese ocurrido en el mar deberá colocar en la caja de texto del “epicentro” la palabra “posible tsunami”.
- Si hubo cualquier cambio en los datos del sismo deberá oprimir el botón ACTUALIZAR para que el sistema actualice el archivo Sfile. Ver ilustración No 37..
- Si los datos del formulario están bien, el analista oprimirá el botón “publicar sismo”. Ver ilustración No 37.

## PUBLICACIÓN DE EMERGENCIAS SISMICAS RED SISMOLOGICA

### Opciones

[Extraer trazas del sismo](#)

[Localizar sismo](#)

### Datos del sismo

#### ARCHIVO SFILE

FECHA	7 DE JUNIO DE 2014	HORA LOCAL	11:49:00
HORA UTC	16:49:00	MAGNITUD Mw	5.3
LATITUD	47.64054	LONGITUD	122.12934
PROFUNDIDAD	40 km		
EPICENTRO	ESTADOS UNIDOS. REDMOND		

ACTUALIZAR

PUBLICAR SISMO

Ilustración 37. Opción Localizar el evento sísmico.

### 6.11.3 Nombre del caso de uso: publicar sismo.

#### 6.11.3.1 Descripción caso de uso:

Permite generar la información para publicar el evento sísmico y así poder avisar a los organismos de socorro sobre la ocurrencia del evento.

#### 6.11.3.2 Requerimiento funcional asociado:

REQ03: Generar información para la publicación del sismo

#### 6.11.3.3 Actores del sistema.

Analistas.

#### 6.11.3.4 Precondiciones.

- El archivo plano Sfile debe estar almacenado en la carpeta “sismo” del computador de procesamiento.
- El actor del sistema debe tener permiso para localizar y publicar sismos.

#### 6.11.3.5 Post condición.

El sistema deberá dejar las imágenes de los mapas generados en la carpeta “sismología” del servidor web, de tal manera que la página existente, que publica el evento sísmico en la web pueda desplegar el mapa.

El sistema deberá dejar el boletín con su respectivo mapa en la carpeta “boletín” del computador de procesamiento.

#### 6.11.3.6 Flujo de eventos.

- Este caso de uso comienza cuando el analista oprime el botón “publicar sismo”. Ver Ilustración No 35.
- El sistema deberá subir la información del archivo plano Sfile a la base de datos “sismología” con el fin de que la página existente, que publica el evento sísmico en la web pueda desplegar estos datos.
- Se debe ejecutar el servicio web “Bing Maps REST Services”. <http://msdn.microsoft.com/en-us/library/ff701710.aspx> (Ver numeral 6.9.1) para traer el nombre de la población en donde ocurrió el sismo. Esta información será grabada en la base de datos “sismología”.
- Se debe ejecutar el servicio generar mapa GMT (ver numeral 6.9.2), este servicio debe dejar los mapas en la carpeta “sismología” del servidor web.
- Se debe generar el boletín del sismo en formato PDF con su respectivo mapa, utilizando la rutina emergencia1 que no deberá tener los procesos de enviar correos y de enviar el fax, ya que para esto se utilizan los servicios.
- Se debe ejecutar el servicio <http://www.programmableweb.com/api/hoio-voice> para que el sistema realice automáticamente las llamadas telefónicas a los organismos de socorro; el mensaje de voz se puede especificar en el parámetro 'msg' haciendo una petición POST al módulo LWP. Para enviar un mensaje de texto (móviles) se debe utilizar el componente API de SMS. (Ver numeral 6.9.5).
- Se debe ejecutar el servicio <http://www.programmableweb.com/api/interfax>, para enviar el boletín por fax. (ver numeral 6.9.4).
- Para enviar los correos electrónicos adjuntando el boletín, se debe ejecutar el servicio web <http://www.programmableweb.com/api/copernica>. (ver numeral 6.9.3)

## 6.12 ARQUITECTURA SOA PROPUESTA PARA EL S.I. PARA LA PUBLICACION DE EVENTOS SISMICOS TECTONICOS DE EMERGENCIA.

Después de haber analizado el proceso de negocios, los requerimientos y los casos de uso se procedió a diseñar la arquitectura, seleccionando el mejor software libre para plataformas SOA. Ver Ilustración No. 38.

### 6.12.1 Gestión de procesos de negocio BPM (orquestración de procesos de negocio).

El BPM sirve para construir procesos ejecutables que se encuentran repartidos entre distintas organizaciones o sistemas. Se seleccionó JBoss jBPM por ser un motor de workflow (secuencias de tareas automáticas) sencillo de utilizar y muy potente.

### 6.12.2 Gestión de la toma de decisiones empresariales EDM.

Es un motor que gestiona y ejecuta las reglas de negocio. Una regla de negocio es un sentencia escrita de cómo debe funcionar el negocio. Se seleccionó el motor de JBoss Rules (Drools) por que tiene un desarrollo maduro.

### 6.12.3 Bus de servicio de empresa ESB.

ESB (Enterprise Service Bus) sirve para proporcionar la interoperabilidad entre distintos protocolos de comunicación. Se seleccionó Apache Synapse por ser un bus ligero y por estar diseñado para proporcionar alta disponibilidad y rendimiento.

### 6.12.4 Procesador de flujos de eventos ESP.

ESP recibe flujo de datos con el fin de detectar patrones entre ellos y disponer de una visibilidad en tiempo real de nuestro sistema (monitorear). Cuando ocurren demasiados eventos eso puede significar la ocurrencia del algún problema. Se seleccionó Esper por ser una solución de software libre en java y por tener una buena documentación.

#### 6.12.5 Registro.

La función del registro, es almacenar los distintos artefactos de software que se usaran en la aplicación SOA. Se seleccionó el registro de WSO2 porque es muy sencillo de utilizar, tiene categorización de recursos, monitorización y la gestión del ciclo de vida.

#### 6.12.6 Componentes de servicio y su composición.

Los servicios o la composición de servicios son lo fundamental en una plataforma SOA. Un servicio se crea en función de la inteligencia del negocio, combinando datos y lógica. Un servicio es un componente que está expuesto a través de un protocolo externo como SOAP. Se seleccionó como componente de servicio a Apache Tuscany porque simplifica las tareas de desarrollo SOA. Se basa en la arquitectura SCA (Service Component Architecture).

#### 6.12.7 Mediador de servicios web.

Un mediador web se utiliza para la seguridad de los datos. También puede transformar versiones de API anteriores a un formato consiste con versiones más actuales. El mediador se utiliza para gestionar el tráfico de las conexiones de los clientes. La principal función que realiza el mediador es la habilidad de traducir mensajes de un protocolo a otro.

Se seleccionó como mediador web a Apache Synapse porque ofrece soporte para WS-Security, funciona como un proxy, posee almacenamiento intermedio, balanceo de carga y tolerancia a cambios.

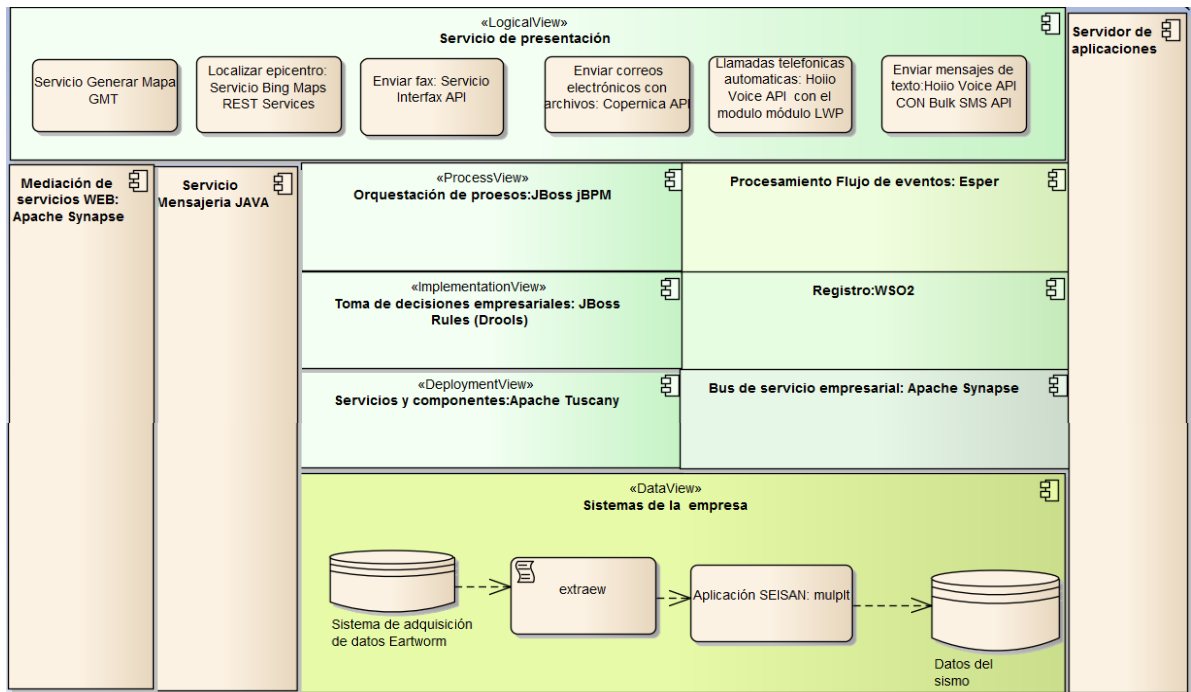


Ilustración 38. Arquitectura SOA propuesta para el S.I. de publicación de eventos sísmicos tectónicos.

#### 6.12.8 DIAGRAMA DE PROCESOS EN LENGUAJE JPDLL (JBOSSE JBPM).

Este diagrama se hizo con BPM de JBPM. Indica la secuencia de pasos que debe realizar los actores y el sistema en el momento de publicar el evento sísmico. Sirve como puente entre los analistas y desarrolladores que van a implementar el sistema. Ver ilustración No 39.

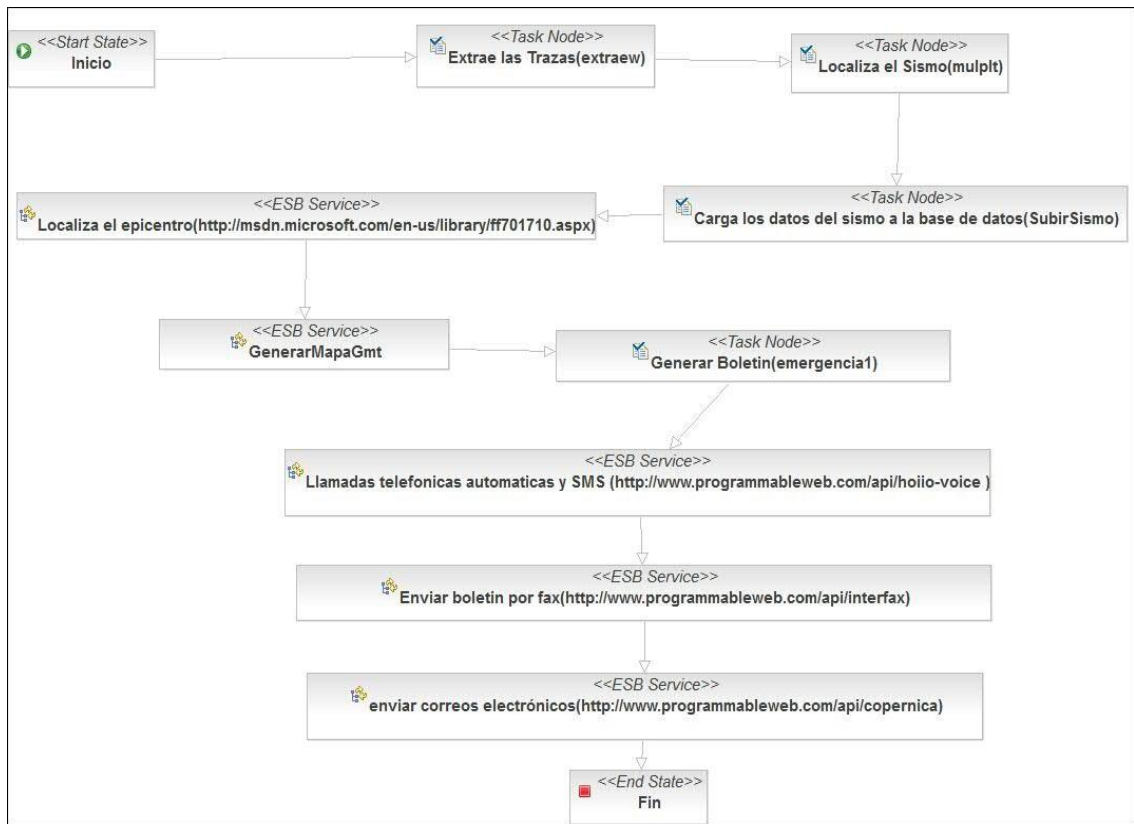


Ilustración 39. Diagrama de procesos en lenguaje JPDL.

El código fuente que genera la herramienta BPM de JBPM se pueda apreciar en la Ilustración No. 40.

```
<?xml version="1.0" encoding="UTF-8"?>

<process-definition
xmlns="urn:jbpm.org:jpdl-3.2"
name="publicarSismo">
<start-state name="Inicio">
<transition to="Extrae las Trazas(extraew)"></transition>
</start-state>
<task-node name="Extrae las Trazas(extraew)">
<transition to="Localiza el Sismo(mulplt)"></transition>
</task-node>

<task-node name="Localiza el Sismo(mulplt)">
<transition to="Carga los datos del sismo a la base de datos(SubirSismo)"></transition>
</task-node>

<task-node name="Carga los datos del sismo a la base de datos(SubirSismo)">
<transition to="Localiza el epicentro(http://msdn.microsoft.com/en-
```



```

us/library/ff701710.aspx"></transition>
</task-node>

<node name="Localiza el epicentro(http://msdn.microsoft.com/en-us/library/ff701710.aspx)">
<action class="org.jboss.soa.esb.services.jbpm.actionhandlers.EsbActionHandler"></action>
<transition to="GenerarMapaGmt"></transition>
</node>

<node name="GenerarMapaGmt">
<action class="org.jboss.soa.esb.services.jbpm.actionhandlers.EsbActionHandler"></action>
<transition to="Generar Boletin(emergencia1)"></transition>
</node>

<task-node name="Generar Boletin(emergencia1)">
<transition to="Llamadas telefonicas automaticas y SMS
(http://www.programmableweb.com/api/hoiio-voice )"></transition>
</task-node>

<node name="Llamadas telefonicas automáticas y SMS
(http://www.programmableweb.com/api/hoiio-voice )">
<action class="org.jboss.soa.esb.services.jbpm.actionhandlers.EsbActionHandler"></action>
<transition to="Enviar boletín por
fax(http://www.programmableweb.com/api/interfax)"></transition>
</node>

<node name="Enviar boletín por fax(http://www.programmableweb.com/api/interfax)">
<action class="org.jboss.soa.esb.services.jbpm.actionhandlers.EsbActionHandler"></action>
<transition to="enviar correos
electrónicos(http://www.programmableweb.com/api/copernica)"></transition>
</node>

<node name="enviar correos electrónicos(http://www.programmableweb.com/api/copernica)">
<action class="org.jboss.soa.esb.services.jbpm.actionhandlers.EsbActionHandler"></action>
<transition to="Fin"></transition>
</node>

<end-state name="Fin"></end-state>
</process-definition><?xml version="1.0" encoding="UTF-8"?>

```

Ilustración 40. Código fuente del Diagrama de procesos JPD.

## 7 EVALUACION DE LA ARQUITECTURA DEL S.I. PARA LA PUBLICACION DE EVENTOS SISMICOS TECTONICOS DE EMERGENCIA CON LA HERRAMIENTA ARCHE.

### 7.1 Requisitos Funcionales.

De acuerdo con los casos de uso se definieron las siguientes funcionalidades:

Número funcionalidad	Descripción funcionalidad
1	Autenticación de usuarios.
2	Borrar contenido de la carpeta EME donde quedaran las trazas del sismo.
3	Extraer las trazas del sismo de una fecha, hora, minutos y segundos con una duración de 1200 segundos.
4	Borrar contenido de la carpeta (sismo) donde quedara el archivo plano Sfile con los datos de localización del sismo.
5	Localizar el sismo.
6	Subir datos de localización del sismo (Sfile) al formulario web.
7	Modificar los datos de localización del sismo en el formulario web.
8	Actualizar el archivo Sfile con los datos del formulario.
9	Subir los datos de localización del sismo del archivo Sfile a la base de datos MySQL (sismología).
10	Localización del epicentro (lugar) del sismo.
11	Generación del mapa GMT.
12	Generación del boletín del sismo.
13	Realizar llamada telefónicas automáticas a los organismos de socorro.
14	Enviar fax del boletín a los organismos de socorro.

15	Publicación en la web.
----	------------------------

## 7.2 Requisitos de calidad de modificabilidad.

### 7.2.1 Escenario 1: configuración de magnitudes.

Las magnitudes miden la energía liberada por el sismo en el hipocentro o foco.

Escenario 1: el sistema deberá tener una opción que permita crear, modificar y borrar los parámetros necesarios para el cálculo de varias magnitudes, como por ejemplo las siguientes magnitudes:  $M_l$  = Magnitud local o de Richter,  $M_c$  = Magnitud Coda,  $M_b$  = Magnitud del cuerpo de la onda,  $M_s$  = Magnitud de ondas superficiales y  $M_w$  = Magnitud del momento.

Configuración de magnitudes	
Estimulo	Configurar parámetros para el cálculo de las magnitudes
Fuente del estimulo	Analista administrador
Entorno	Tiempo de ejecución
Artefacto	Sistema
Respuesta	Modificación de la estructura del archivo plano Sfile y la estructura de la base de datos.
Medida	3 Días.
Valor	20

### 7.2.2 Escenario 2: Publicación de eventos sísmicos volcánicos.

Escenario 2: El sistema deberá modificarse para que permita la publicación de eventos sísmicos volcánicos. La localización de estos eventos sísmicos volcánicos se realiza en los observatorios vulcanológicos. La información de la localización tal

como la magnitud, profundidad, hora, minutos, segundos, y el epicentro son enviados por correo electrónico a la red sismológica para que allí se publiquen.

Publicación de eventos sísmicos volcánicos	
Estimulo	Se recibe la información del evento sísmico volcánico desde un observatorio vulcanológico
Fuente del estímulo	Analista
Entorno	Tiempo de ejecución
Artefacto	Sistema
Respuesta	Se debe digitar en el sistema la magnitud, profundidad, día, hora, minutos, segundos y el epicentro
Medida	5 días
Valor	20.0

### 7.3 Operación de ArchE.

Una vez definido los requisitos funcionales y los requisitos de calidad se ejecuta la herramienta Arche para evaluar la arquitectura. Se deben realizar los siguientes pasos para la operación de Arche:

#### 7.3.1 Funcionamiento de Mysql.

Se debe verificar que el servicio de la base de datos Mysql esté funcionando, al no estar se debe activar, para que ArchE pueda acceder a la base de datos que se creó y se configuro en su instalación.

#### 7.3.2 Ejecución de XmIBlaster.

XmIBlaster permite la comunicación de ArchE con la base de datos Mysql. Para activarlo se debe configurar las siguientes variables de ambiente:

```
set JAVA_HOME=C:\Archivos de programa\Java\jdk1.6.0_03
```

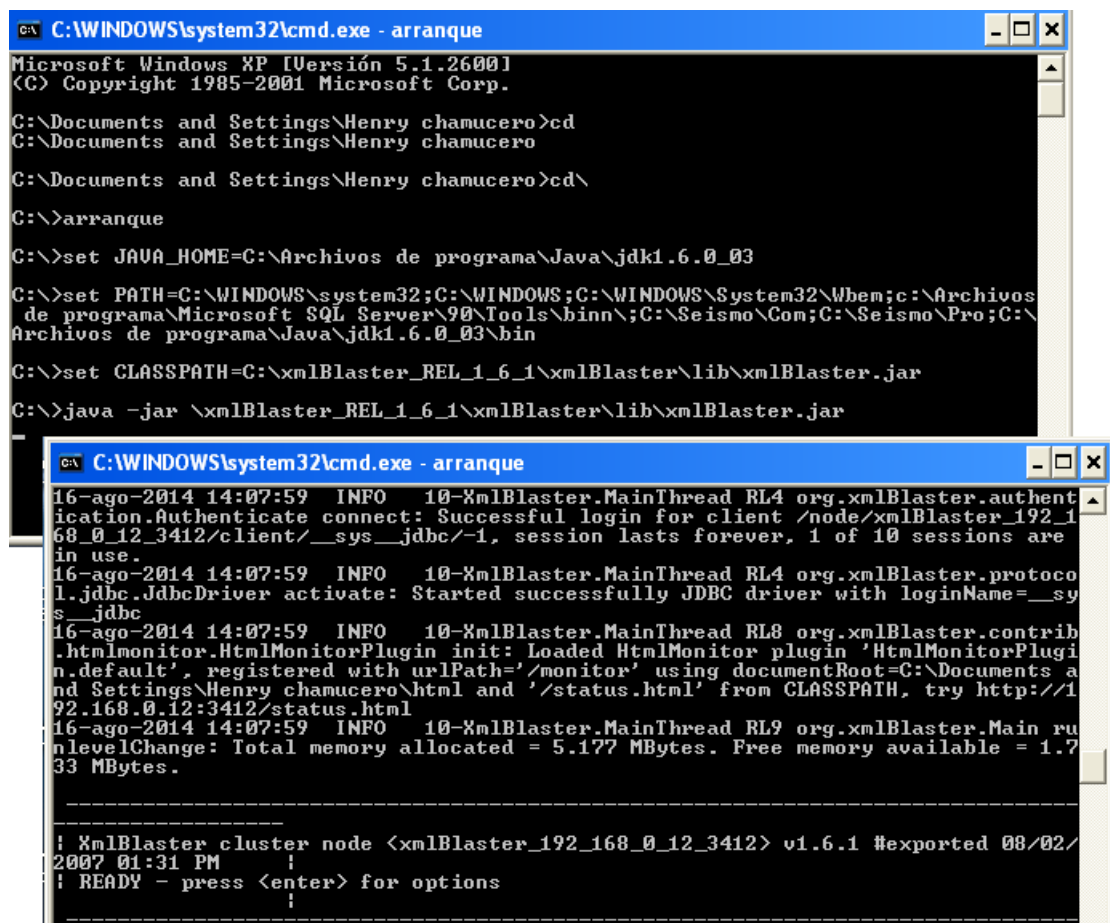
```
set PATH=%PATH%;%JAVA_HOME%\bin
```

```
set CLASSPATH=C:\xmlBlaster_REL_1_6_1\xmlBlaster\lib\xmlBlaster.jar
```

Para activar XmlBlaster se ejecuta la siguiente instrucción:

```
java -jar \xmlBlaster_REL_1_6_1\xmlBlaster\lib\xmlBlaster.jar.
```

También se puede crear un archivo arranque.bat con los comandos anteriores y ejecutarlo. Ver Ilustración No 41



```
C:\WINDOWS\system32\cmd.exe - arranque
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Henry chamucero>cd
C:\Documents and Settings\Henry chamucero
C:\Documents and Settings\Henry chamucero>cd\
C:\>\arranque
C:\>\set JAVA_HOME=C:\Archivos de programa\Java\jdk1.6.0_03
C:\>\set PATH=C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;c:\Archivos
de programa\Microsoft SQL Server\90\Tools\binn\;C:\Seismo\Com;C:\Seismo\Pro;C:\
Archivos de programa\Java\jdk1.6.0_03\bin
C:\>\set CLASSPATH=C:\xmlBlaster_REL_1_6_1\xmlBlaster\lib\xmlBlaster.jar
C:\>\java -jar \xmlBlaster_REL_1_6_1\xmlBlaster\lib\xmlBlaster.jar

16-ago-2014 14:07:59 INFO 10-XmlBlaster.MainThread RL4 org.xmlBlaster.authentic
ication.Authenticate connect: Successful login for client /node/xmlBlaster_192_1
68_0_12_3412/client/_sys_jdbc/-1, session lasts forever, 1 of 10 sessions are
in use.
16-ago-2014 14:07:59 INFO 10-XmlBlaster.MainThread RL4 org.xmlBlaster.protoco
l.jdbc.JdbcDriver activate: Started successfully JDBC driver with loginName=_sy
s_jdbc
16-ago-2014 14:07:59 INFO 10-XmlBlaster.MainThread RL8 org.xmlBlaster.contrib
.htmlmonitor.HtmlMonitorPlugin init: Loaded HtmlMonitor plugin 'HtmlMonitorPlugi
n.default', registered with urlPath='/monitor' using documentRoot=C:\Documents a
nd Settings\Henry chamucero\html and '/status.html' from CLASSPATH, try http://1
92.168.0.12:3412/status.html
16-ago-2014 14:07:59 INFO 10-XmlBlaster.MainThread RL9 org.xmlBlaster.Main ru
nlevelChange: Total memory allocated = 5.177 MBytes. Free memory available = 1.7
33 MBytes.

-----
! XmlBlaster cluster node <xmlBlaster_192_168_0_12_3412> v1.6.1 #exported 08/02/
2007 01:31 PM
! READY - press <enter> for options
!
-----
```

Ilustración 41. Ejecución de xmlBlaster.

### 7.3.3 Creación del proyecto ArchE publicarsismos.

Después de haber ejecutado ArchE desde el icono. Se crea el proyecto seleccionando la opción File/New/Arche Project. Ver ilustración No 42.

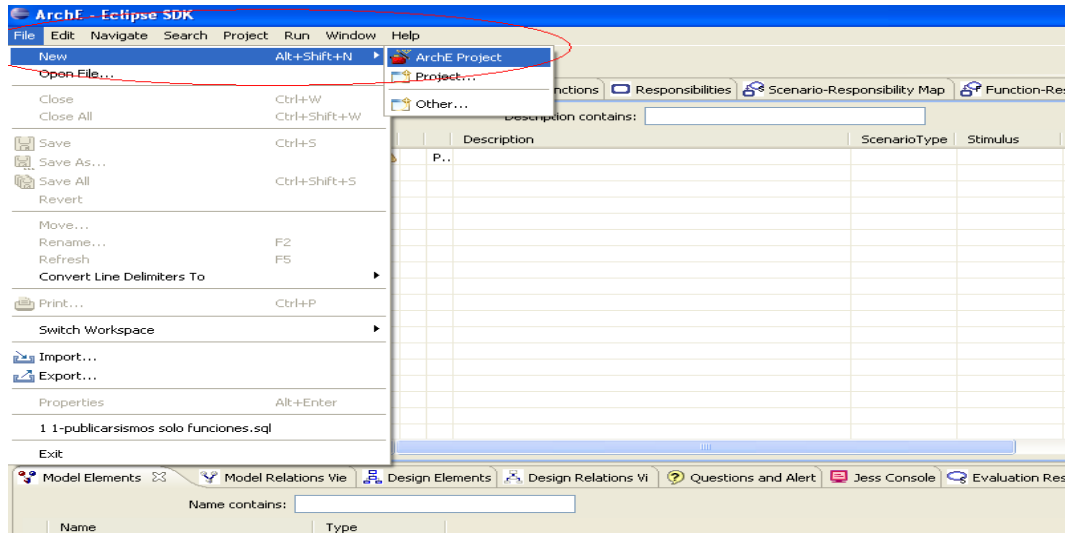


Ilustración 42. Pasos para crear un proyecto ArchE.

En seguida se visualiza un cuadro de dialogo donde se digita el nombre del proyecto y se oprime el botón Finish. Ver ilustraciones No 43.

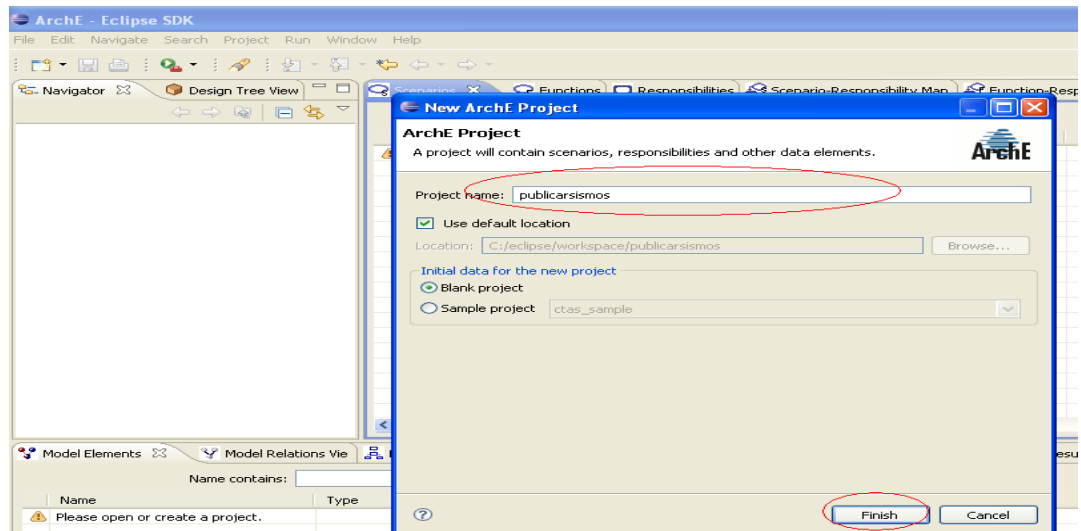


Ilustración 43. Creación del proyecto publicarsismos.

### 7.3.4 Creación de las funcionalidades.

Para poder crear en ArchE los requisitos funcionales se hace click en la pestaña Functions y después con el botón derecho del mouse se hace click en la ventana apareciendo un menú emergente. En este menú se selecciona la opción New function desplegando un cuadro de dialogo. Ver ilustración No 44.

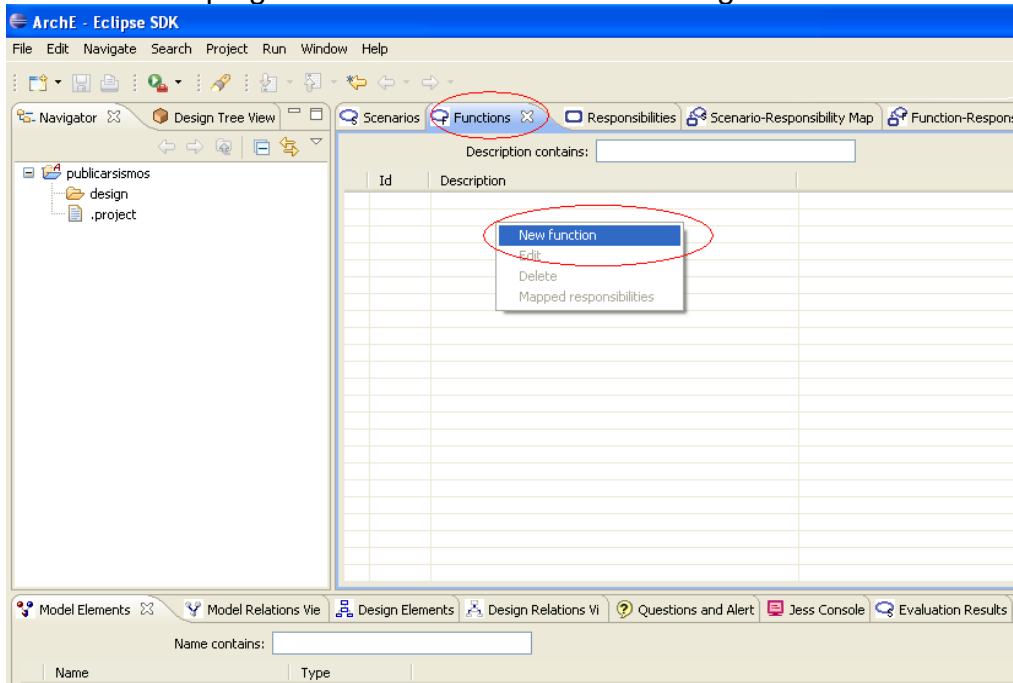


Ilustración 44. Pasos para crear las funcionalidades en ArchE.

En la caja de texto id del cuadro de dialogo se digita el número de identificación de la funcionalidad y en la caja de texto descripción la descripción de la funcionalidad. Ver ilustraciones No. 45 y 46.

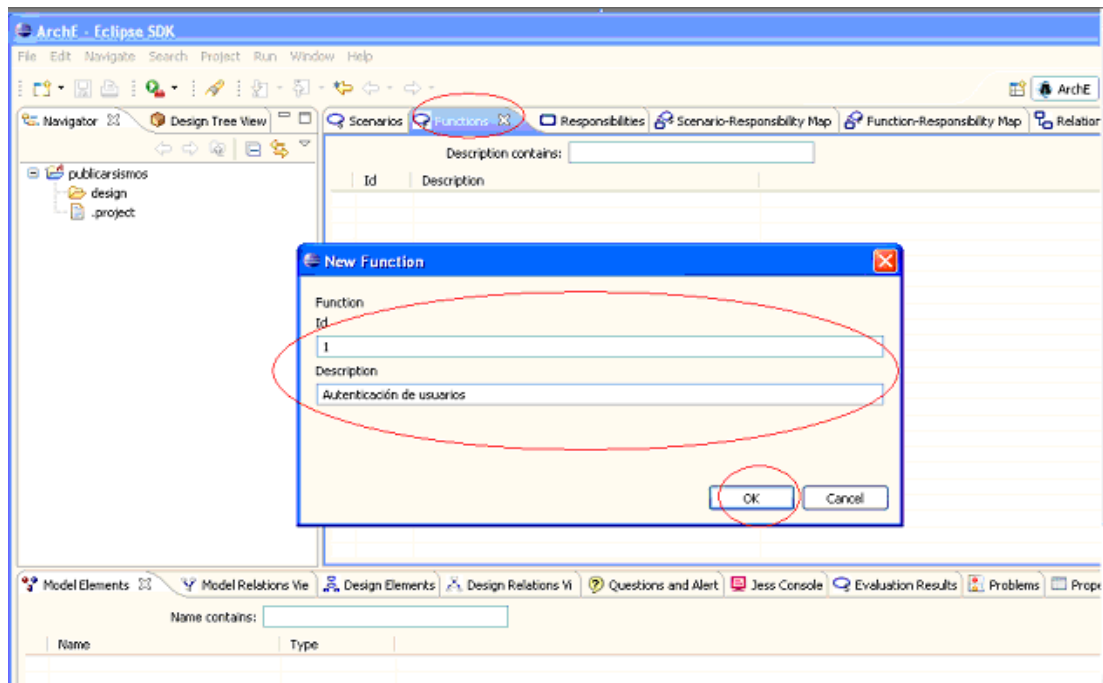


Ilustración 45. Caja de diálogo crear nueva función.

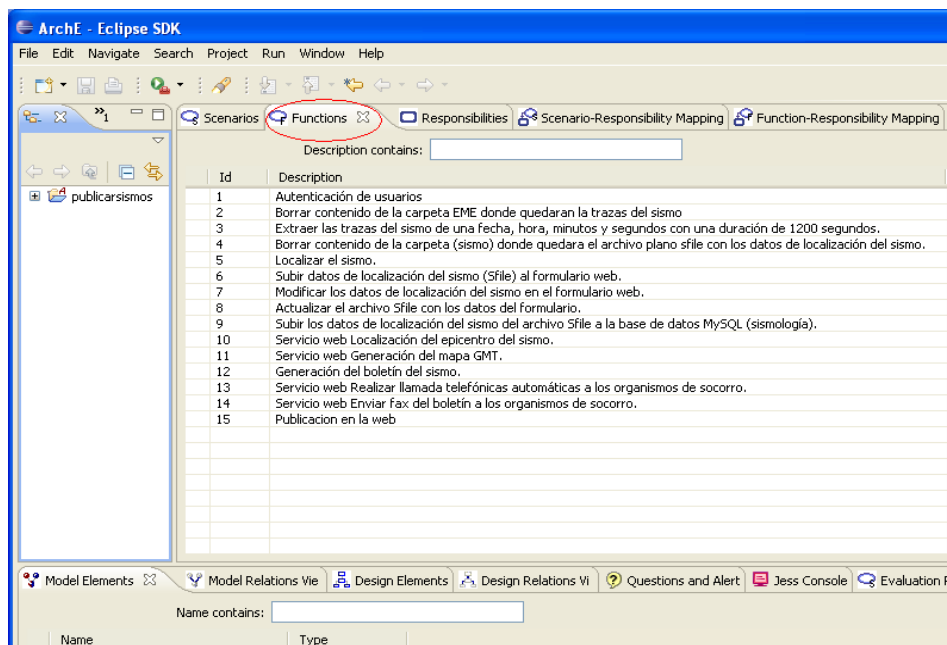


Ilustración 46. Funcionalidades para la publicación de sismos.



### 7.3.5 Manejo de responsabilidades.

Al hacer click en la pestaña Responsibilities se despliegan las responsabilidades. Las responsabilidades automáticamente se crean a medida que se van creando las funcionalidades, por cada funcionalidad se crea una responsabilidad. Ver ilustración No. 47.

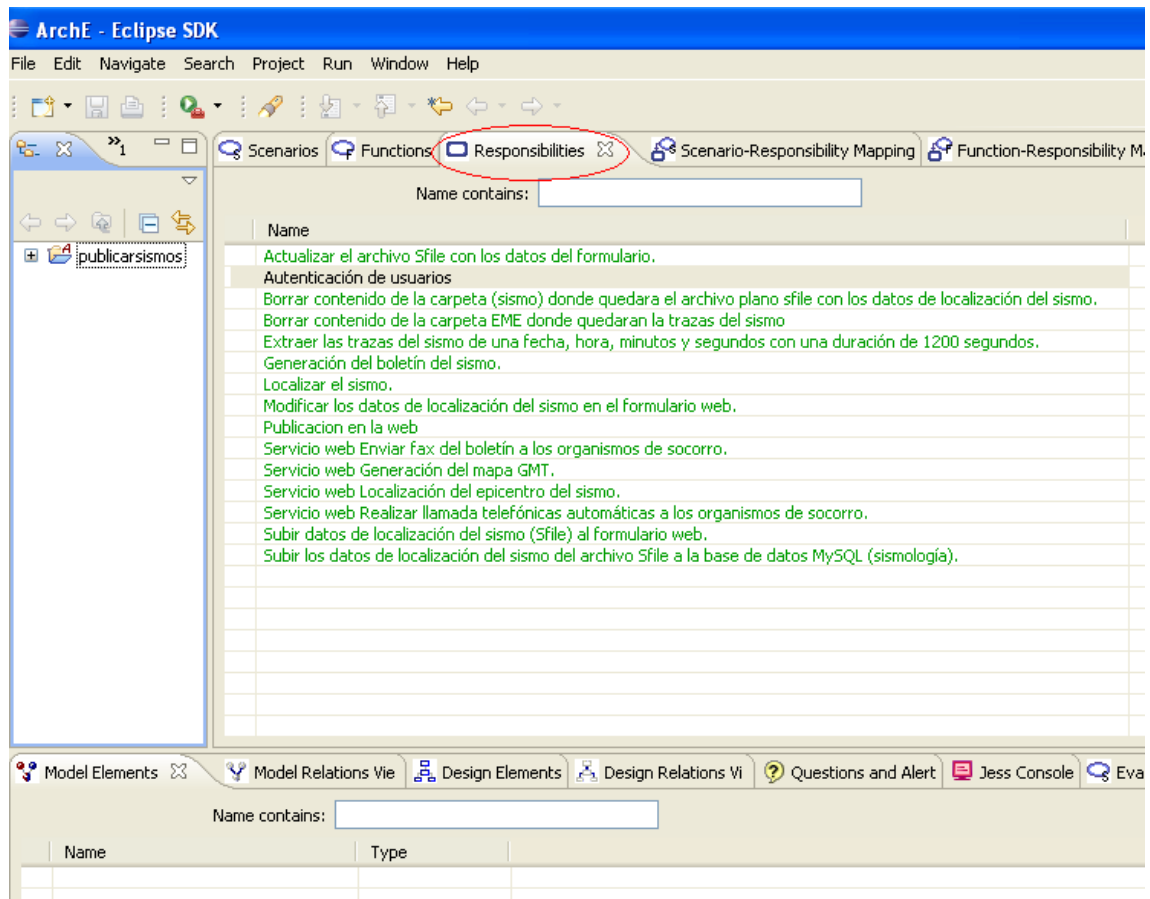


Ilustración 47. Responsabilidades para el proyecto publicar sismos.

Al seleccionar una responsabilidad con el mouse y haciendo click con el botón derecho se despliega un menú en donde aparecen las opciones de crear una nueva responsabilidad, editar o borrar la responsabilidad seleccionada. Ver ilustración No. 48.

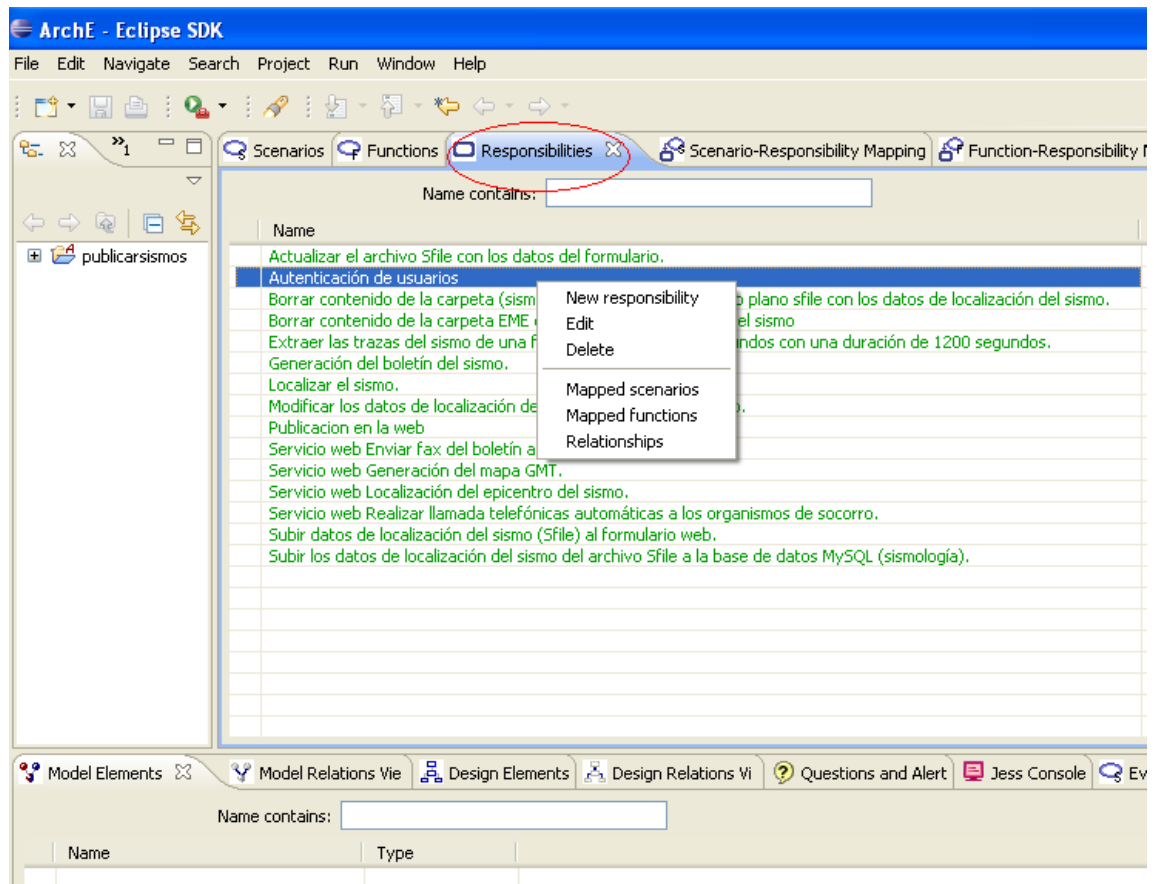


Ilustración 48. Menú emergente para el manejo de responsabilidades.

### 7.3.6 Manejo del mapeo entre funcionalidades y responsabilidades.

Para ver el mapeo se debe hacer click en la pestaña Function-Responsibility Mapping. Este mapeo se realiza automáticamente cuando se están creando las funcionalidades. Ver ilustración No. 49.

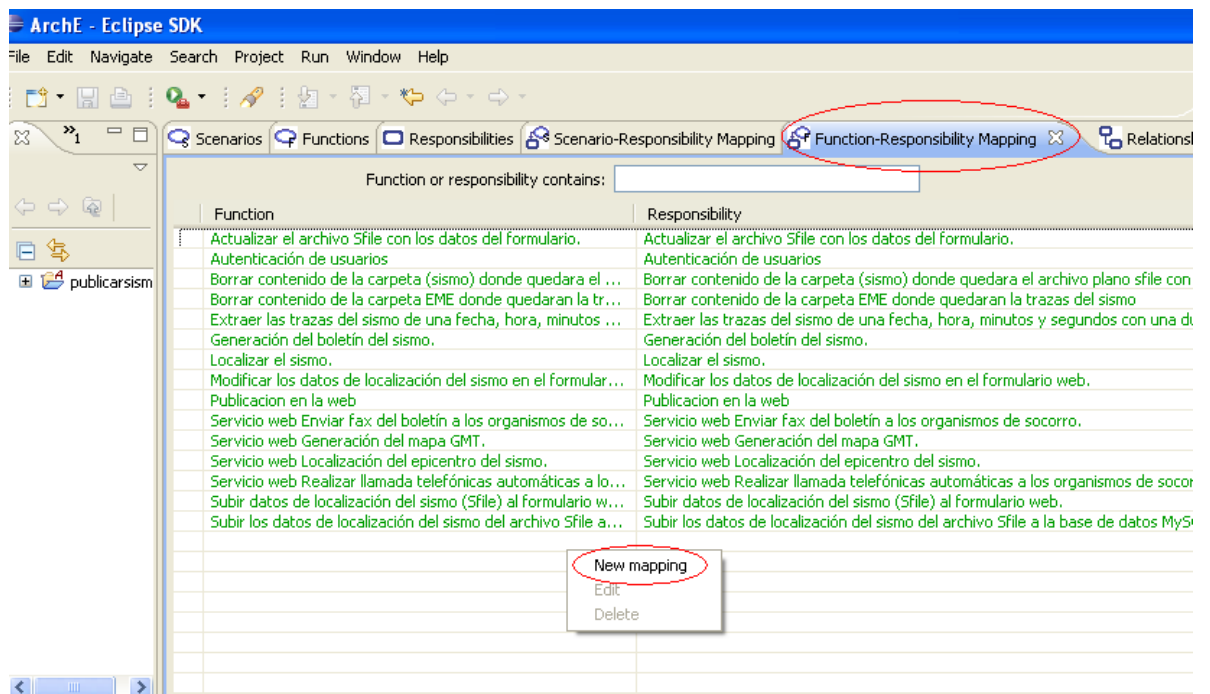


Ilustración 49. Mapeo entre funcionalidades y responsabilidades del proyecto publicar sismos.

Al hacer click con el botón derecho del mouse sobre la ventana del mapeo se despliega un menú. Seleccionando la opción New mapping se puede crear un nuevo mapeo, en el combo desplegable Function se selecciona la función y en el combo desplegable Responsibility se selecciona la responsabilidad y a continuación se oprime el botón ok. Ver ilustración No. 50.

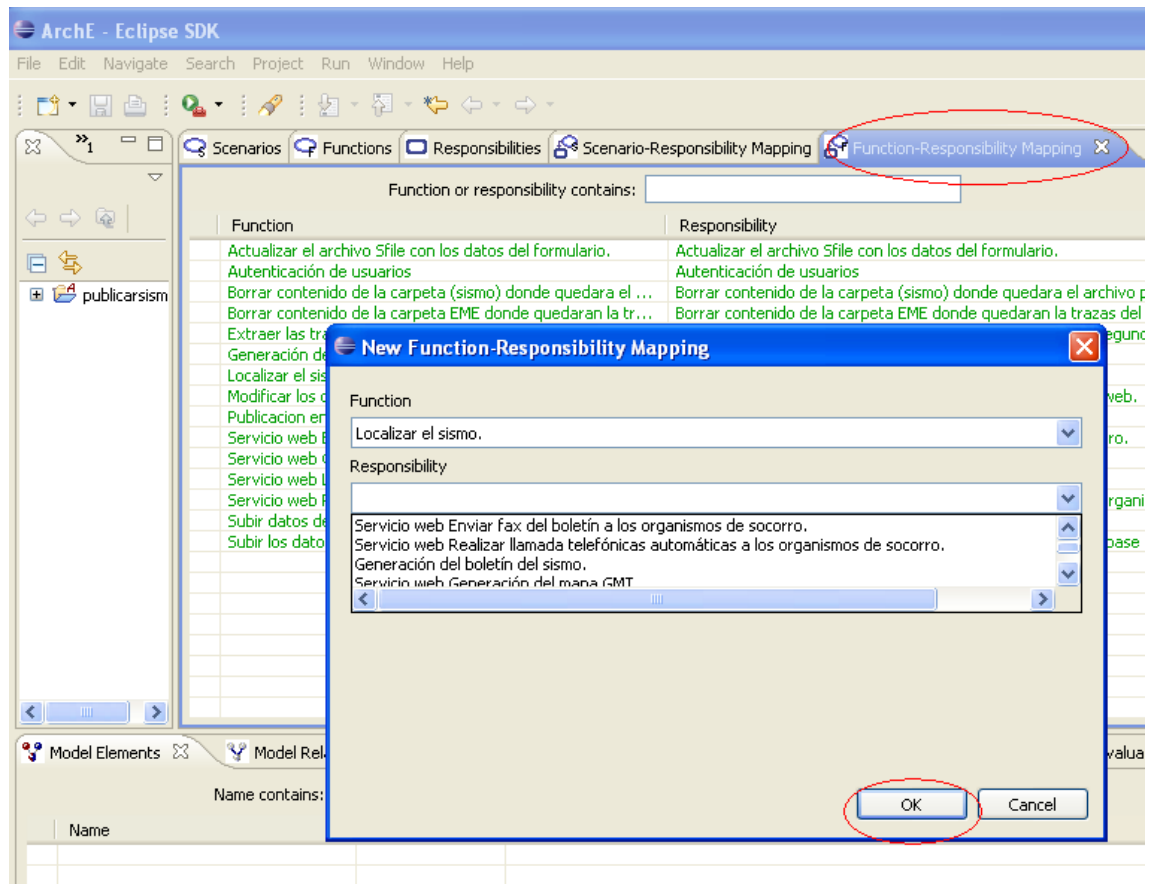



Ilustración 50. Creación de un nuevo mapeo.

### 7.3.7 Creación de los escenarios de calidad de modificabilidad

Para poder aplicar en ArchE los escenarios para atributos de calidad de modificabilidad al proyecto de publicación de eventos sísmicos y a cualquier otro proyecto ArchE, lo primero que se debe hacer es cargar el marco de razonamiento de modificabilidad. Para esto se debe hacer click en el icono  "Show View as a fast view". Ver ilustración No. 51.

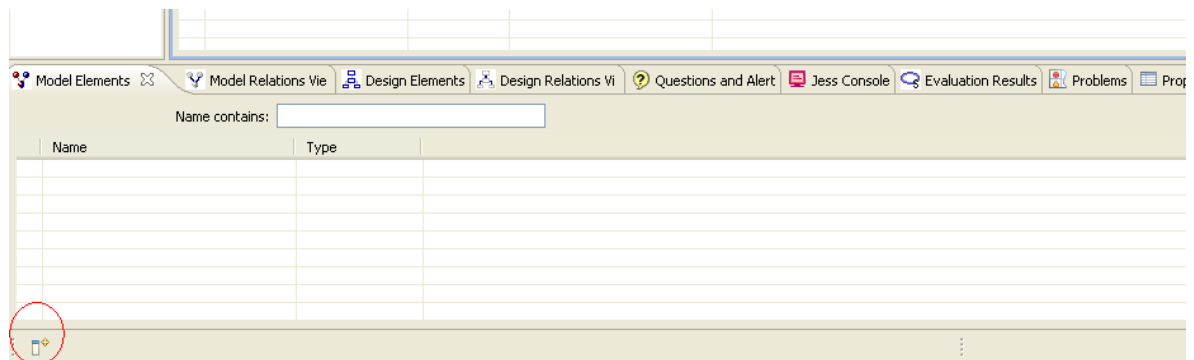


Illustration 51. Icono Show View as a fast view.

Al hacer click sobre este icono se despliega un menú, de donde se selecciona la opción Other. Ver ilustración No 52. En seguida se muestra el cuadro de dialogo “Show View”, después se hace click sobre la carpeta “ArchE External RF Samples” para mostrar los marcos de razonamiento, en seguida se selecciona el marco de modificabilidad “ChangelImpact Modifiability RF View” y se oprime el botón OK. Ver ilustración No .53.

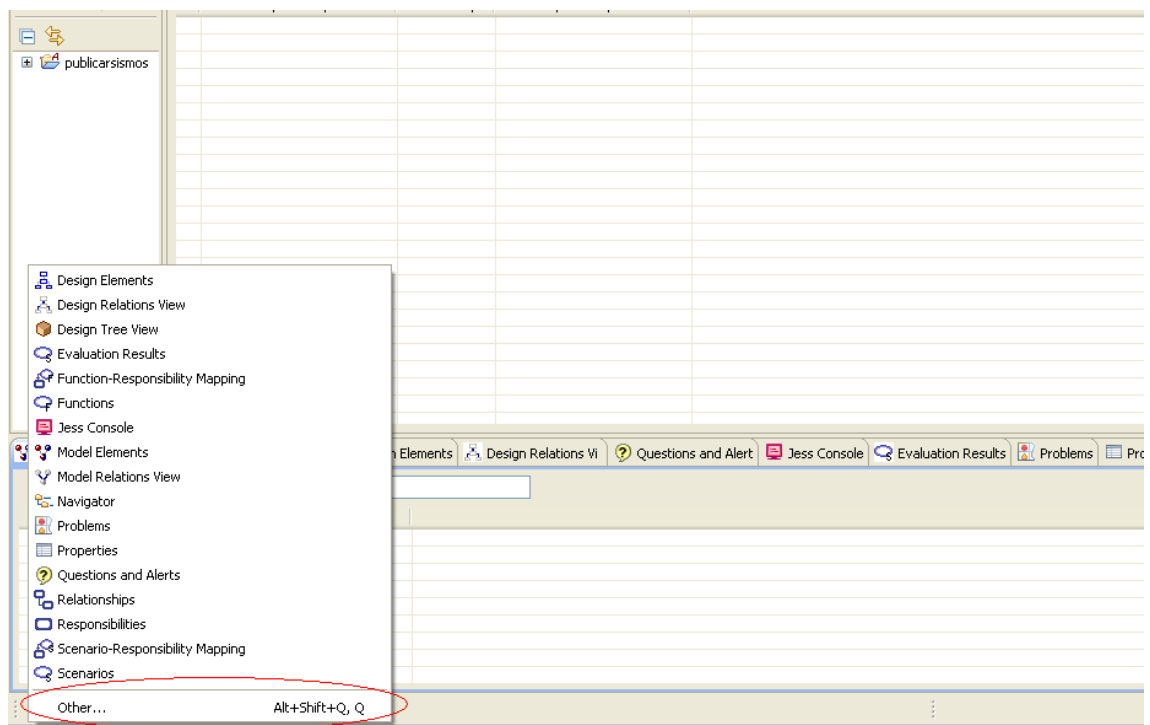


Ilustración 52. Menú emergente.

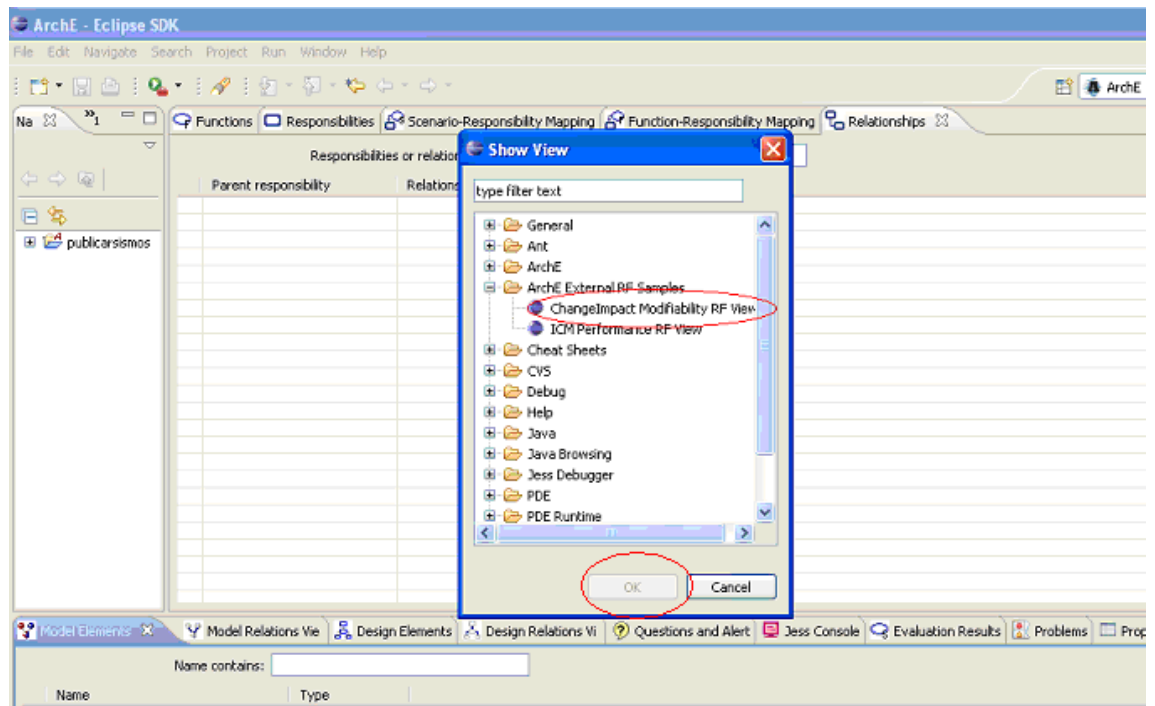


Ilustración 53. Seleccionar marco de razonamiento de modificabilidad.

Para ejecutar el marco de razonamiento se oprime el botón Start. Ver Ilustración No. 54.

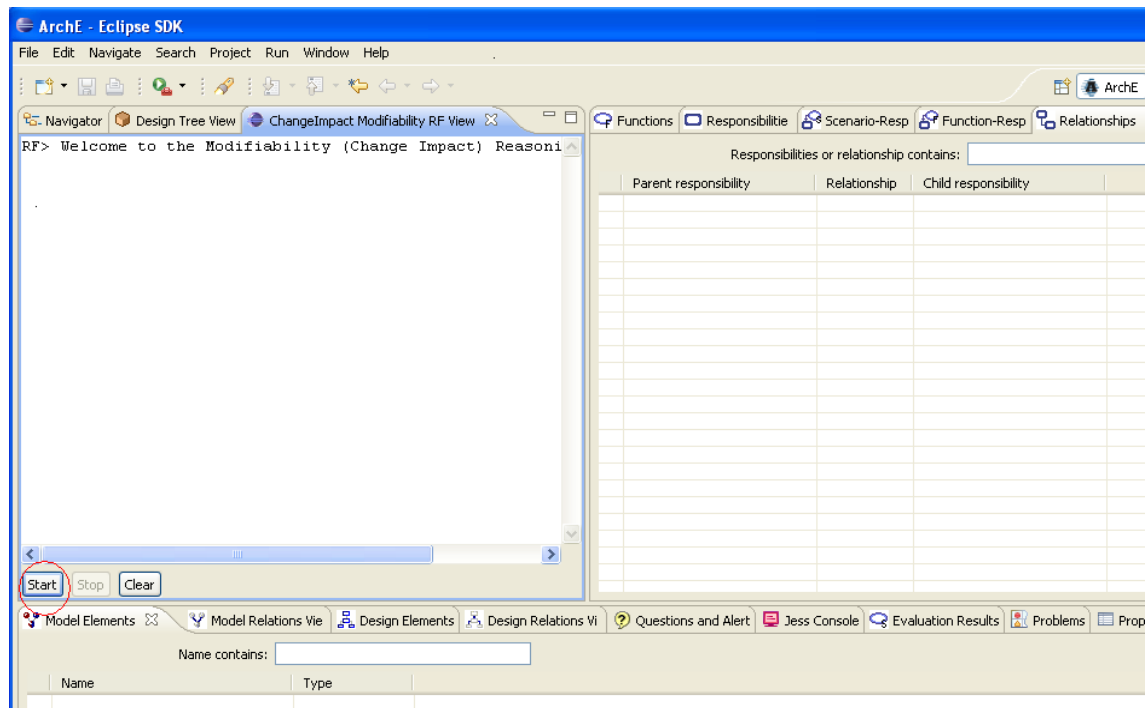


Ilustración 54. Ejecución del marco de razonamiento de modificabilidad.

Una vez cargado el marco de razonamiento de modificabilidad se pueden crear las relaciones entre responsabilidades y los escenarios de los atributos de calidad de modificabilidad.

En segundo lugar se debe crear las Relationships. Para esto se hizo una tabla donde se muestra las responsabilidades y sus dependencias. Ver ilustración No. 55.

Responsabilidad	Depende de la responsabilidad
Autenticación de usuarios	Ninguna
Borrar contenido de la carpeta EME donde quedarán las trazas del sismo	Autenticación de usuarios.
Extraer las trazas del sismo de una fecha, hora, minutos y segundos con una duración de 1200 segundos.	Autenticación de usuarios.
Borrar contenido de la carpeta "sismo" donde quedará el archivo plano sfile	Autenticación de usuarios.

Localizar el sismo.	Extraer las trazas del sismo de una fecha, hora, minutos y segundos con una duración de 1200 segundos.
Subir datos de localización del sismo (Sfile) al formulario web.	Localizar el sismo.
Modificar los datos de localización del sismo en el formulario web.	Subir datos de localización del sismo (Sfile) al formulario web.
Actualizar el archivo Sfile con los datos del formulario.	Modificar los datos de localización del sismo en el formulario web.
Subir los datos de localización del sismo del archivo Sfile a la base de datos MySQL (sismología).	<ul style="list-style-type: none"> <li>• Localizar el sismo.</li> <li>• Modificar los datos de localización del sismo en el formulario web.</li> </ul>
Localización del epicentro (lugar) del sismo.	Subir los datos de localización del sismo del archivo Sfile a la base de datos MySQL (sismología).
Generación del mapa GMT.	Localización del epicentro (lugar) del sismo.
Generación del boletín del sismo.	<ul style="list-style-type: none"> <li>• Subir los datos de localización del sismo del archivo Sfile a la base de datos MySQL (sismología).</li> <li>• Localización del epicentro (lugar) del sismo.</li> <li>• Generación del mapa GMT.</li> </ul>
Realizar llamada telefónicas automáticas a los organismos de socorro.	Subir los datos de localización del sismo del archivo Sfile a la base de datos MySQL (sismología).
Enviar fax del boletín a los organismos de socorro.	Generación del boletín del sismo.
Publicación en la web	<ul style="list-style-type: none"> <li>• Subir los datos de localización del sismo del archivo Sfile a la base de datos MySQL (sismología).</li> <li>• Localización del epicentro</li> </ul>



	(lugar) del sismo. <ul style="list-style-type: none"> <li>• Generación del mapa GMT.</li> </ul>
--	---

Ilustración 55. Dependencia entre responsabilidades.

Para crear las relaciones entre responsabilidades en ArchE, se selecciona la pestaña Relationships y haciendo click con el botón derecho del mouse sobre la ventana, emerge un menú de donde se selecciona la opción New relationship para crear una nueva relación. Ver ilustración No 56. En seguida se despliega un cuadro de dialogo con tres combos. El primer combo "Responsibility" es para seleccionar la responsabilidad que se le va a crear la relación. El segundo combo "Relationship" es para seleccionar la clase de dependencia, en este caso se selección la única que había "Dependency". El tercer combo "Responsibility" es para seleccionar la responsabilidad que se va a relacionar y finalmente se oprime el botón OK. Ver ilustraciones No. 57 y 58

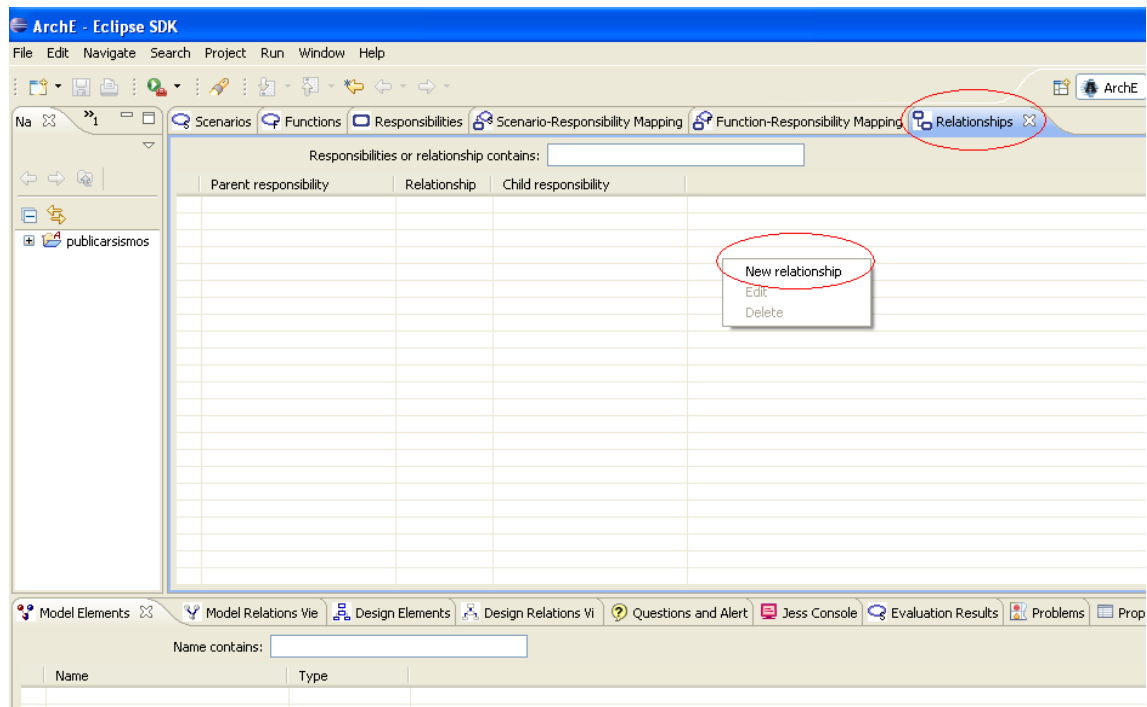


Ilustración 56. Menú emergente para el manejo de relaciones entre responsabilidades.

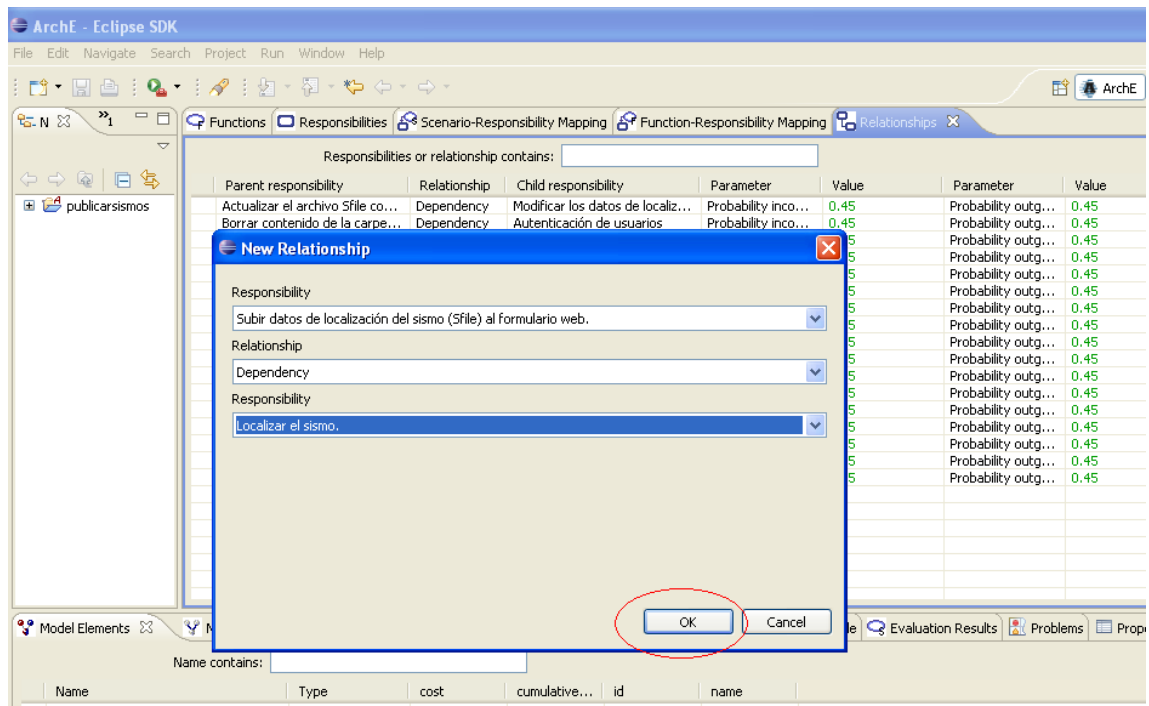


Ilustración 57. Creación de relaciones.

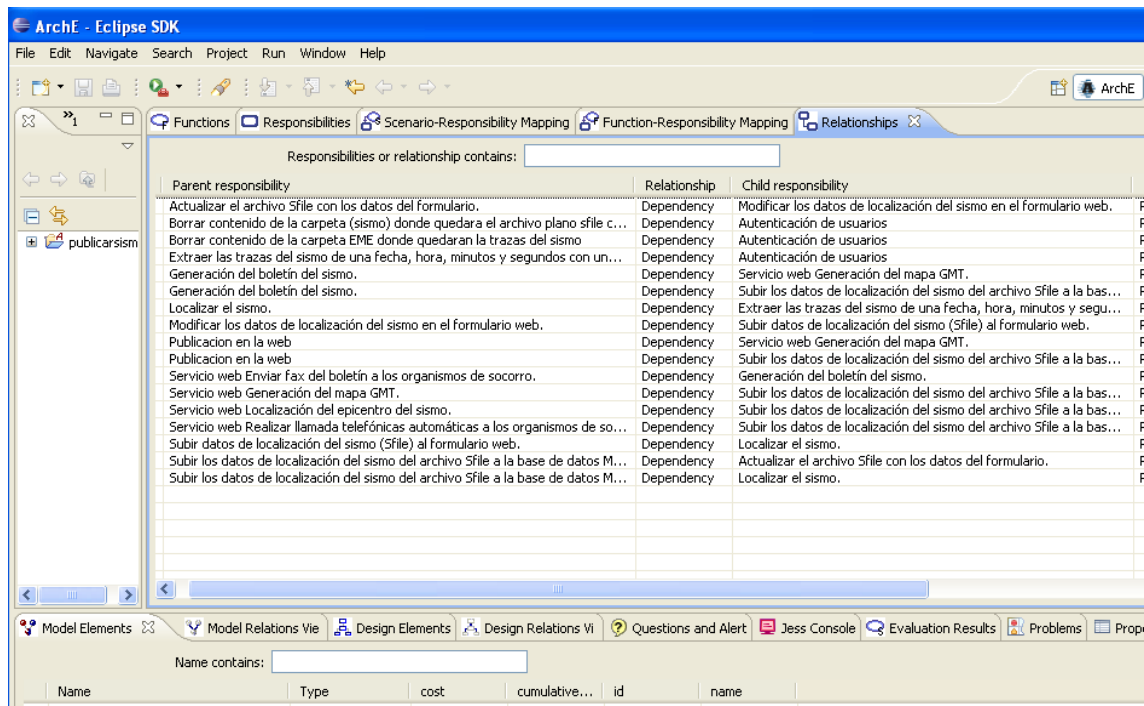


Ilustración 58. Relaciones para el S.I. publicar eventos sísmicos.

En tercer lugar se crean los escenarios de los atributos de calidad de modificabilidad.

### 7.3.7.1 Creación del Escenario 1: configuración de magnitudes.

Para crear el escenario1 se selecciona la pestaña Scenarios y se hace click con el botón derecho del mouse y se selecciona la opción New Scenario del menú emergente. Ver ilustración No. 59. En seguida se despliega el cuadro de dialogo Scenario; en la caja de texto Scenario Text se escribe el nombre del escenario, En el combo type se selecciona “ChangelImpact Modifiability” y se digitan los demás datos correspondientes al escenario1. Ver Ilustración No 60.

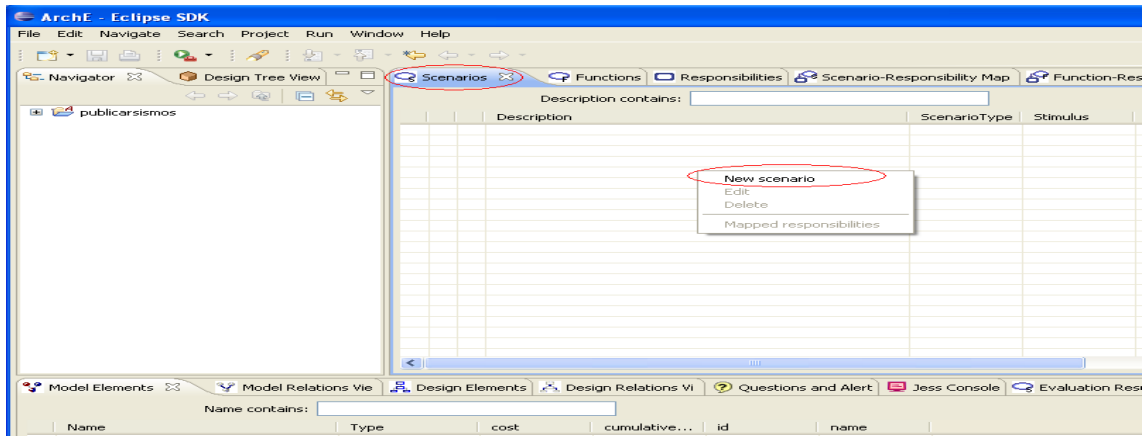


Ilustración 59. Menú emergente para el manejo de escenarios.

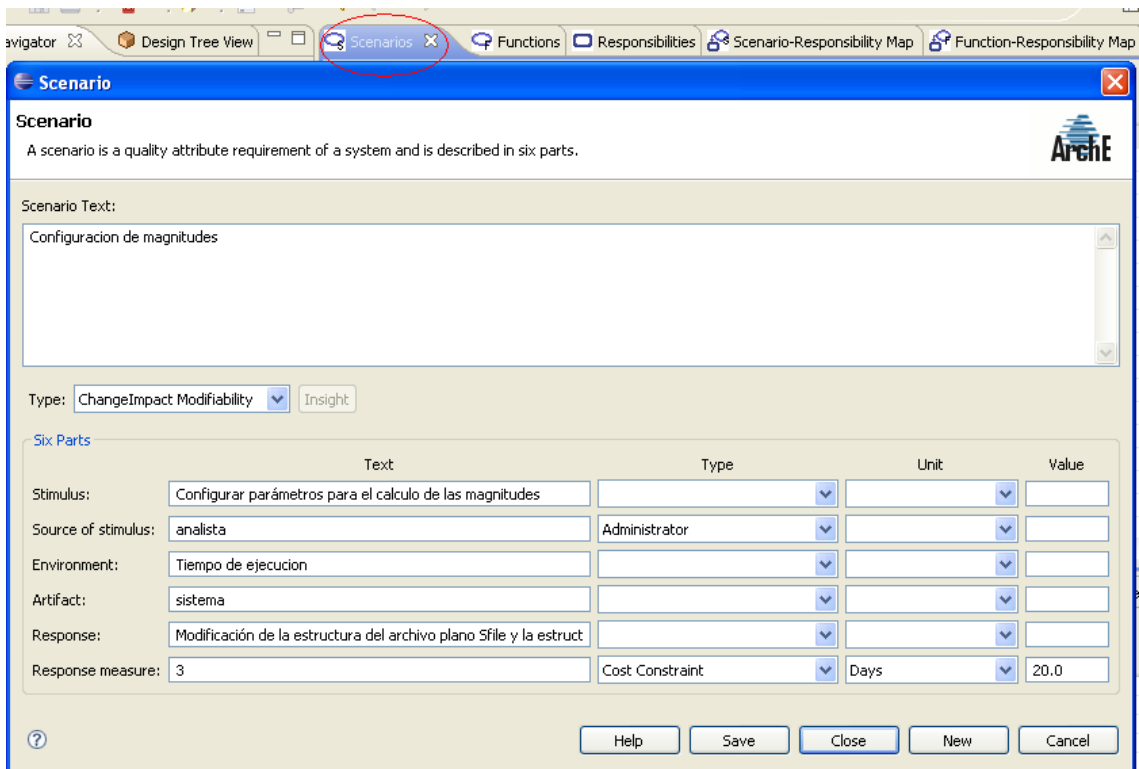


Ilustración 60. Creación del escenario configuración de magnitudes.

Hasta este punto ArchE sugiere una arquitectura inicial en el momento en que se vuelve a ejecutar el marco de razonamiento de modificabilidad. Ver ilustración No. 61.

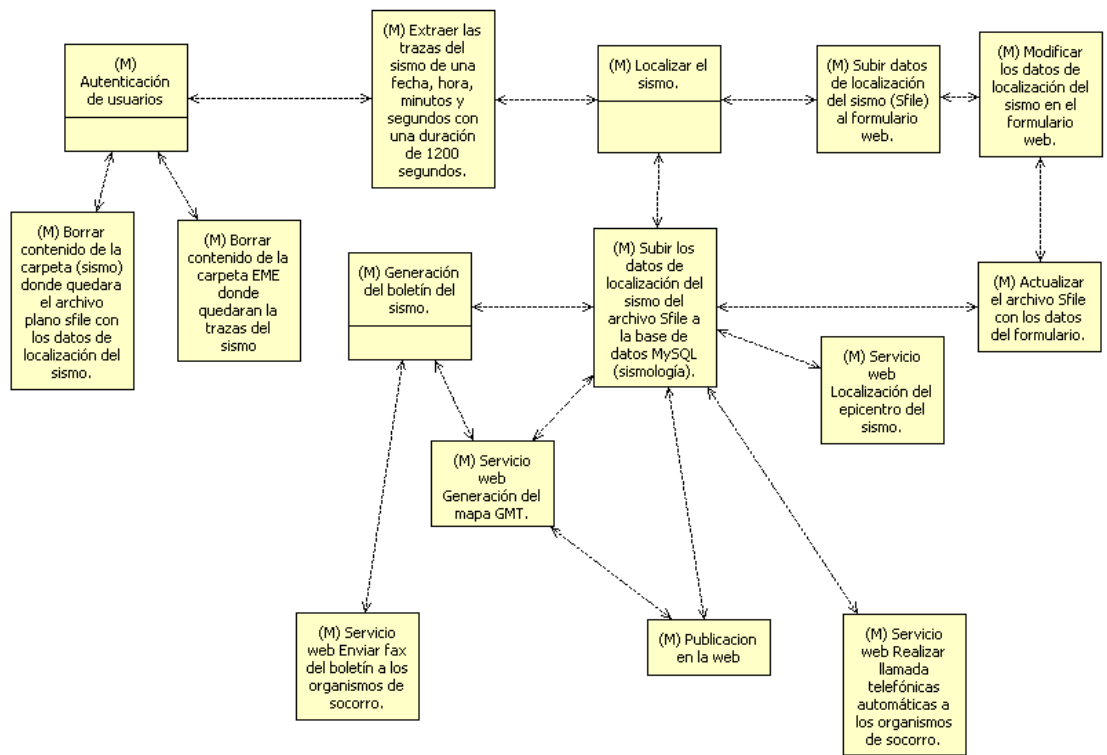


Ilustración 61. Arquitectura inicial propuesta por ArchE.

Se crea la funcionalidad “Parametrizar magnitudes” que es la funcionalidad para el escenario 1. Ver ilustración No 62.

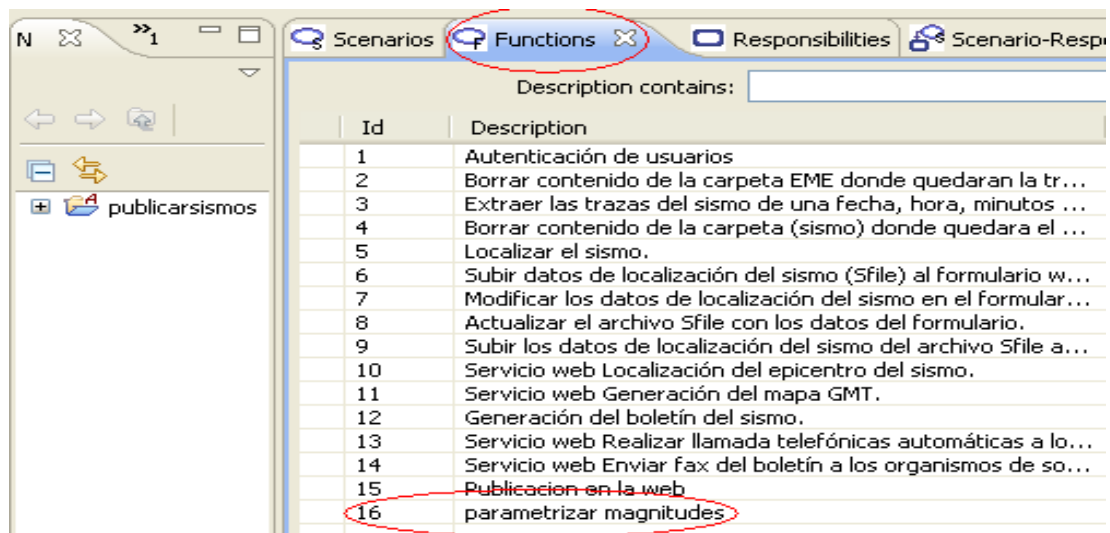


Ilustración 62. Creación de la funcionalidad parametrizar funcionalidades.

Al crear la funcionalidad “Parametrizar magnitudes” automáticamente se crea la responsabilidad “Parametrizar magnitudes” y el mapeo respectivo.

Se crea el mapeo del escenario No 1 con las responsabilidades. Para ejecutar la responsabilidad “localizar el sismo” las magnitudes deben estar configuradas, y para configurar las magnitudes estas se deben parametrizar. Ver ilustración No. 63

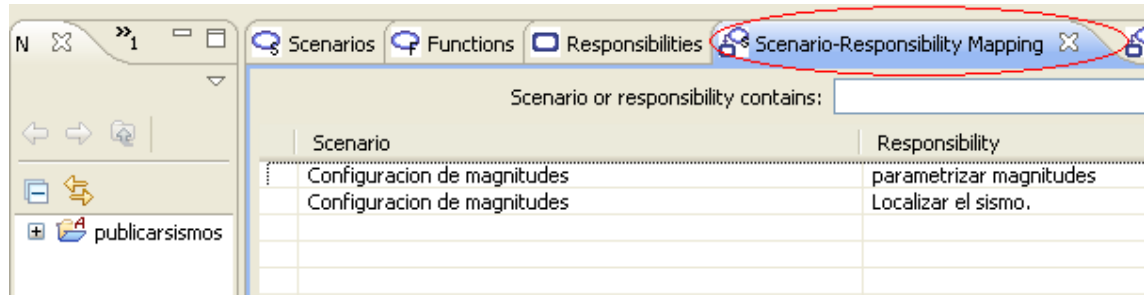


Ilustración 63. Creación del mapeo entre el escenario "Configuración magnitudes" y responsabilidades respectivas.

El marco de razonamiento de modificabilidad evalúa el escenario No 1. Ver ilustración No. 64.

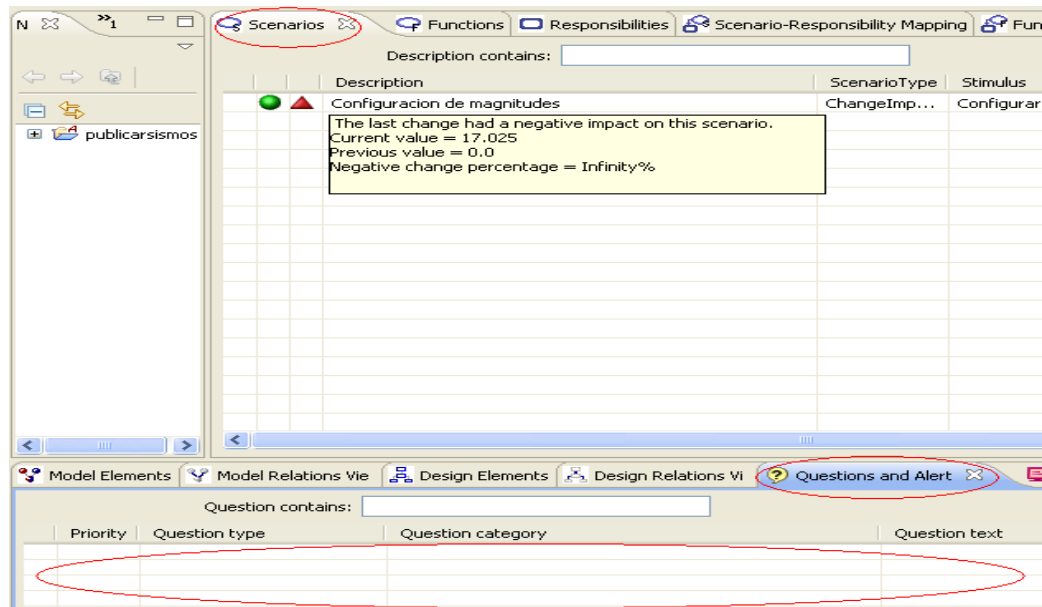


Ilustración 64. Evaluación del escenario "configuración de magnitudes."

El círculo verde indica que el escenario se ha evaluado y se satisface. Esto me hace concluir que la arquitectura SOA propuesta es viable, debido a que ArchE no sugirió ninguna táctica para satisfacer el escenario No 1.

El triángulo rojo significa que el diseño de la arquitectura desmejoro en un valor de 17.025 con respecto al inicial.

ArchE sugiere un nuevo diseño arquitectónico, donde se puede apreciar la adición del módulo “parametrizar magnitudes”. Ver ilustración No. 65.

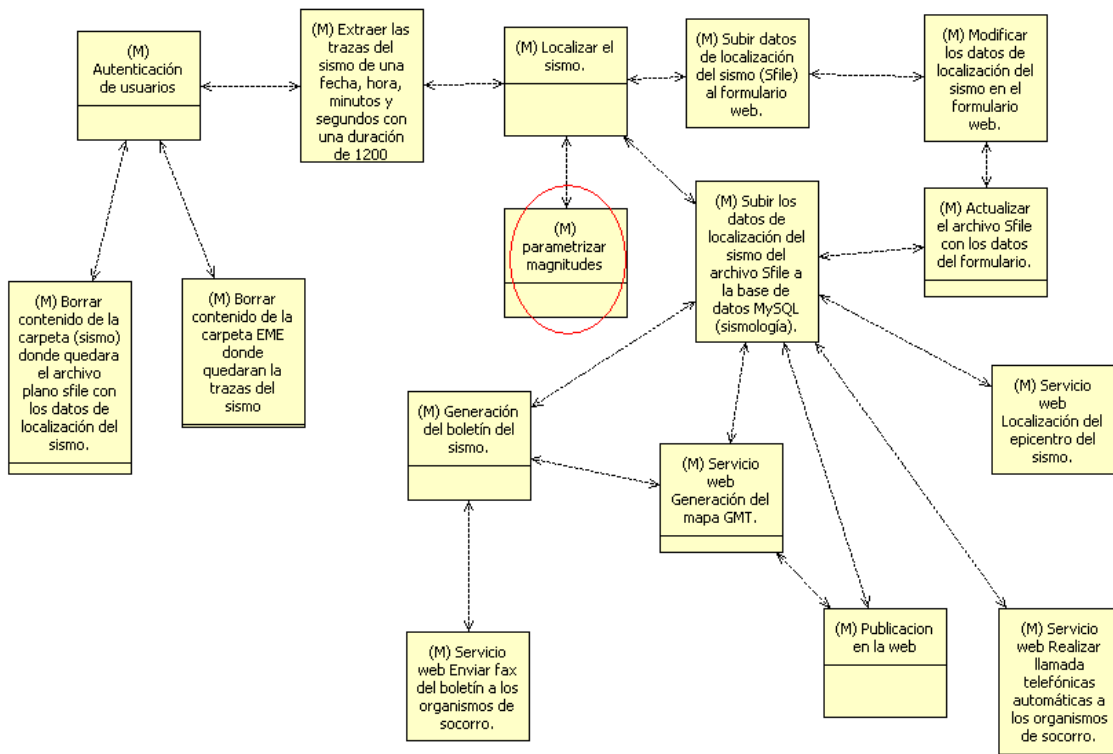


Ilustración 65. Segundo diseño arquitectónico propuesto por ArchE

### 7.3.7.2 Creación del escenario 2: Publicación de eventos sísmicos volcánicos.

En este escenario no hay necesidad de localizar el sismo. Los datos de la localización del sismo llegan por medio de un correo electrónico desde los observatorios. El sistema debe crear un archivo Sfile como evidencia, subir estos datos a la base de datos MySQL (sismología) y publicar el sismo en la web.

En la ilustración No 66 se muestra la creación del escenario 2 en ArchE.

Scenario

A scenario is a quality attribute requirement of a system and is described in six parts.

Scenario Text:  
Publicar eventos sismicos volcanicos

Type: ChangeImpact Modifiability Insight

	Text	Type	Unit	Value
Stimulus:	Se recibe la información del evento sismico volcanico desde un			
Source of stimulus:	Analista	End user		
Environment:	Tiempo de ejecución			
Artifact:	sistema			
Response:	Se debe digitar en el sistema la magnitud, profundidad,dia,hori:			
Response measure:	5 dias	Cost Constraint	Days	20.0

Buttons: Help, Save, Close, New, Cancel

Ilustración 66. Creación del escenario "publicar eventos sísmicos volcánicos".

Después de haber creado el escenario se crea el mapeo del escenario con las siguientes responsabilidades: Autenticación de usuarios, Actualizar el archivo Sfile con los datos del formulario (Se digitan los datos en el formulario), subir los datos de localización del sismo del Archivo Sfile a la base de datos Mysql (sismología) y publicación en la web. Ver ilustración No 67.



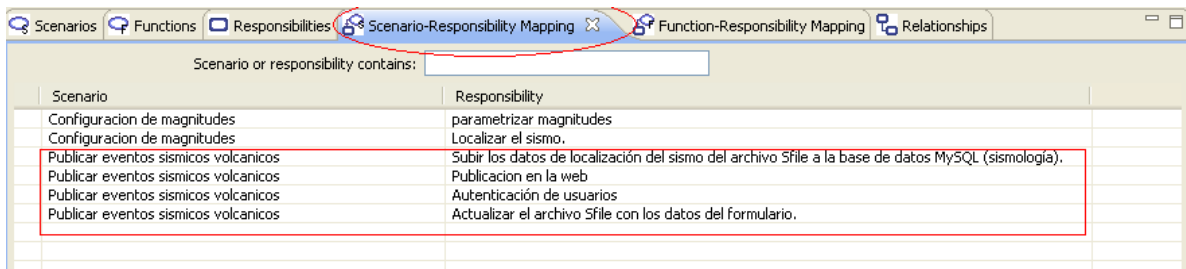


Ilustración 67. Creación del mapeo entre el escenario "publicar eventos sísmicos volcánicos" con sus respectivas responsabilidades.

Una vez creado el mapeo, el marco de razonamiento de modificabilidad evalúa ambos escenarios. Ver ilustración No 68.

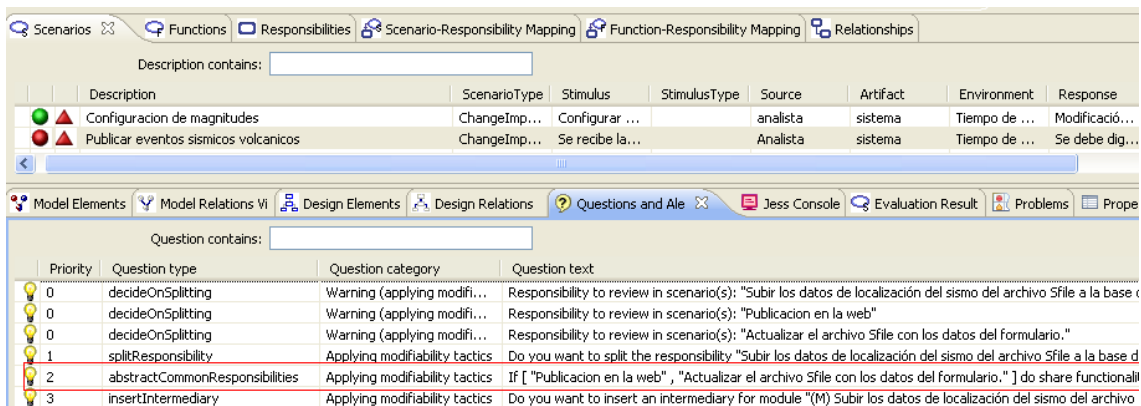


Ilustración 68. Primera iteración: Evaluación de los escenarios "configuración de magnitudes " y "publicar eventos sísmicos volcánicos"

El círculo verde del escenario 1 "configuración de magnitudes" indica que este escenario satisface.

El círculo rojo del escenario 2 "Publicar eventos sísmicos volcánicos" indica que este escenario no satisface. ArchE, Sugiere 3 tácticas a aplicar para que el escenario satisfaga. Tome la decisión de aplicar la segunda "abstractCommonResponsibilities" táctica que dice así:

"Las responsabilidades "Publicación en la web" y "Actualizar el archivo sFile Con Los Datos del Formulario". Pueden compartir varias funcionalidades, y también

pueden estar en diferentes módulos. Por lo tanto, podría ser una buena idea mover las funcionalidades a un solo módulo en común con el fin de mejorar la cohesión semántica. Una estimación sugiere que podría reducir los costos de "40.565" a "38.87226470588235" días-persona para este escenario."

Para aplicar la segunda táctica se hace doble click sobre ella, entonces se despliega el cuadro de dialogo de la ilustración No.

A la pregunta: {Si "Publicación en la web" y "Actualizar el archivo sFile Con Los Datos del Formulario" comparten funcionalidades, ¿usted quiere dividir y poner las partes comunes en un módulo independiente?} Seleccione la respuesta "SI" y oprímí el botón Finish. Ver ilustración No. 69.

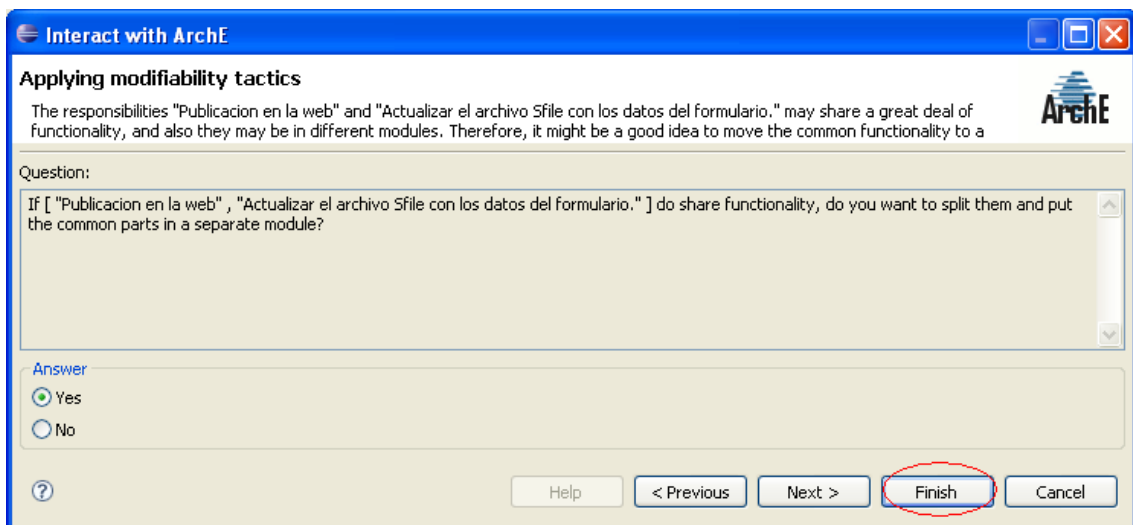


Ilustración 69. Primera iteración: aplicación de la segunda táctica "abstractCommonResponsibilities"

Al terminar ArchE de aplicar la táctica, vuelve y evalúa los dos escenarios. En el escenario 1 el círculo está en verde indicando que se satisface. En el escenario 2 el círculo es de color rojo indicando que no satisface, pero el triángulo verde del escenario 2 indica que el diseño de la arquitectura mejoró en 12.54%. Ver ilustración No. 70.

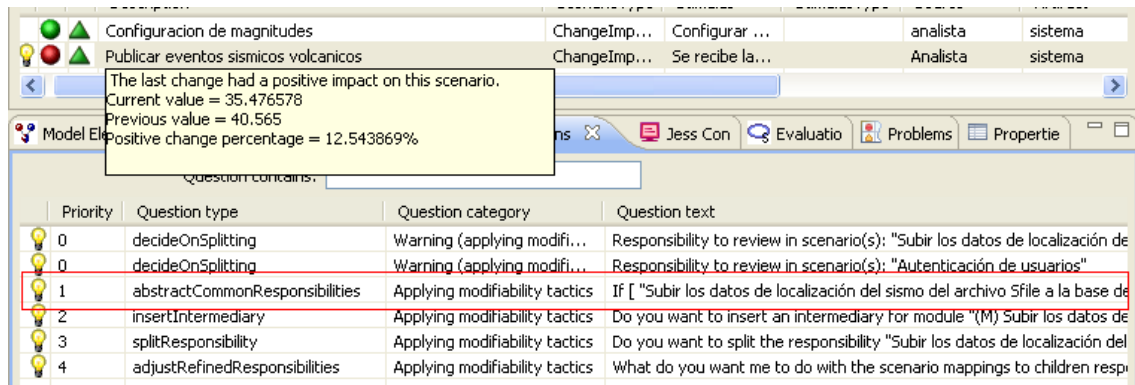


Ilustración 70. Segunda iteración: Evaluación de escenarios.

ArcHE vuelve a sugerir 4 tácticas más. En este caso se aplica la táctica 1 "abstractCommonResponsibilities" que dice así:

Las responsabilidades: "Subir los Datos de Localización del sismo del archivo sFile a la base de MySQL Datos de (sismología)." y "Autenticación de Usuarios" pueden compartir una gran cantidad de funcionalidades, y también pueden estar en diferentes módulos. Por lo tanto, podría ser una buena idea para mover la funcionalidad común a un solo módulo con el fin de mejorar la cohesión semántica. Una estimación sugiere que podría reducir los costos de "35.476578947368424" a "34.058515625" días-persona para este escenario de cambio.

Se aplica la táctica y se responde "SI" a la pregunta: {Si "Subir los Datos de Localización del sismo del archivo sFile a la base de MySQL (sismología)." y "Autenticación de Usuarios" comparten funcionalidades, ¿usted quiere dividir y poner las partes comunes en un módulo independiente? Ver ilustración No 71.

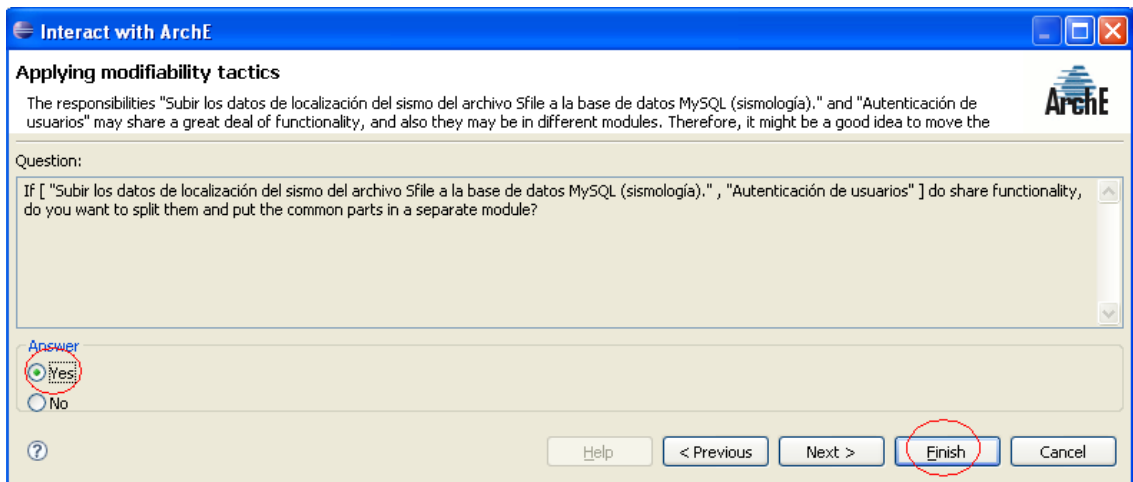


Ilustración 71. Segunda iteración: aplicación de la táctica 1 “abstractCommonResponsibilities”

ArchE vuelve a evaluar los escenarios y el escenario 2 vuelve a no satisfacer. Ver ilustración No 72.

Description	ScenarioType	Stimulus	StimulusType	Source	Artifact	Environment	Response
Configuración de magnitudes	ChangeImp...	Configurar ...		analista	sistema	Tiempo de ...	Modificació.
Publicar eventos sísmicos volcánicos	ChangeImp...	Se recibe la...		Analista	sistema	Tiempo de ...	Se debe dig

Priority	Question type	Question category	Question text
0	decideOnSplitting	Warning (applying modifiability tactics)	Responsibility to review in escenario(s): "Subir los datos de localización del sismo"
0	decideOnSplitting	Warning (applying modifiability tactics)	Responsibility to review in escenario(s): "Publicacion en la web_shared_6"
1	insertIntermediary	Applying modifiability tactics	Do you want to insert an intermediary for module "(M) Subir los datos de localiz
2	abstractCommonResponsi...	Applying modifiability tactics	If [ "Publicacion en la web_shared_6" , "Subir los datos de localización del sismo
3	splitResponsibility	Applying modifiability tactics	Do you want to split the responsibility "Publicacion en la web_shared_6"?
4	adjustRefinedResponsibilit...	Applying modifiability tactics	What do you want me to do with the scenario mappings to children responsibi

Ilustración 72. Tercera iteración: Evaluación de los escenarios.

En esta oportunidad se aplica la cuarta táctica “adjustRefineResponsibilities” que dice así:

Después de aplicar “abstractCommonServices”, la funcionalidad "Autenticación de Usuarios" y la funcionalidad "Subir los Datos de Localización del sismo del archivo Sfile a la base de MySQL (sismología)” quedaron refinadas en "Autenticación de usuarios\_child\_16 " y en "Subir los Datos de Localización del sismo del archivo

sFile a la base de datos MySQL (sismología)\_shared\_14". Esto podría tener una influencia en el cálculo del coste del escenario. Sin embargo, el escenario no puede afectar realmente las responsabilidades de los hijos, y si es así, el costo podría reducirse más. Ver ilustración No. 73.

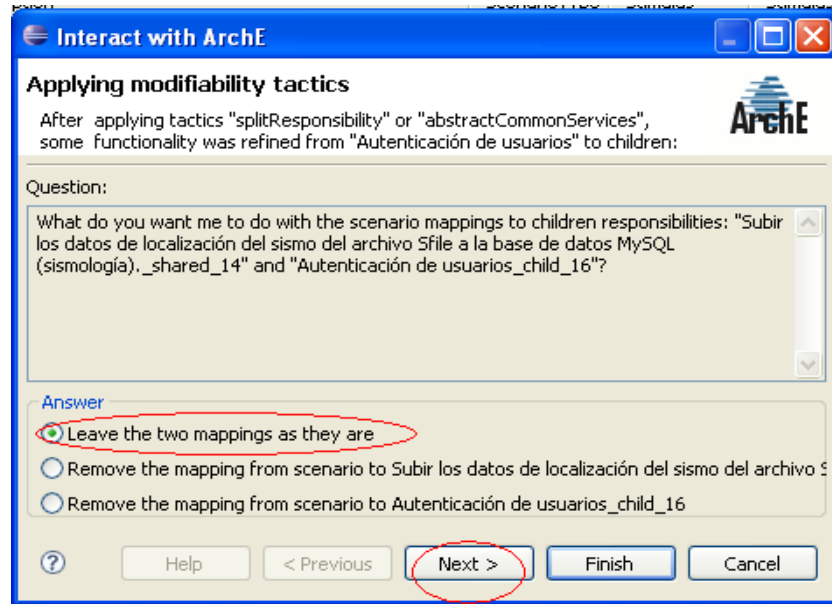


Ilustración 73. Tercera iteración: Aplicación de cuarta táctica “adjustRefineResponsabilities”

Al aplicar esta táctica se asigna un costo de 30.0 para realizar el cambio. Ver ilustración No. 74.

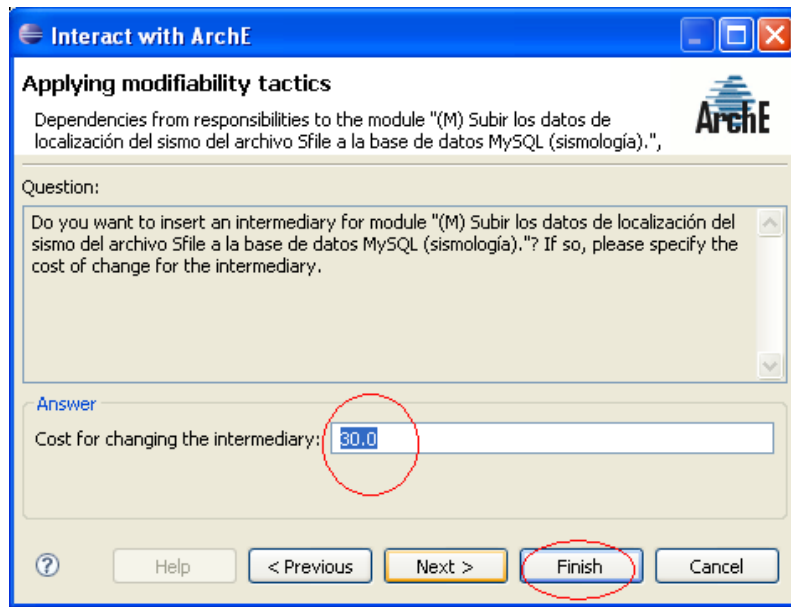


Ilustración 74. Tercera iteración: costo de la cuarta táctica “adjustRefineResponsibilities”

Al evaluar esta táctica se cumple que los escenarios no se afectaron, según el triángulo amarillo que aparecen en cada escenario. Ver ilustración No 75.

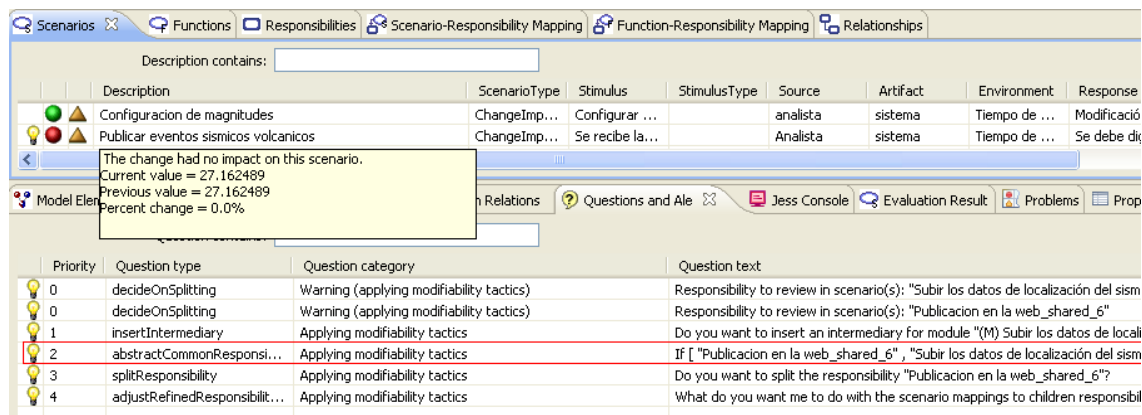


Ilustración 75. Cuarta iteración: Evaluación de los escenarios.

Aquí se aplica la táctica 2 “abstractCommonResponsibilities” que es para dejar las responsabilidades "Publicación en la web\_shared\_6" y "Subir los Datos de Localización del sismo del archivo sFile a la base de MySQL de Datos (sismología). \_shared\_14" en un solo modulo. Se estima que este cambio podría

reducir los costos de "27.162488636363634" a "26.571244565217395" días-persona.

Efectivamente este cambio se redujo de 27.162489 a 24.956938 como se puede apreciar en la ilustración No. 76, Pero el escenario 2 sigue sin satisfacer.

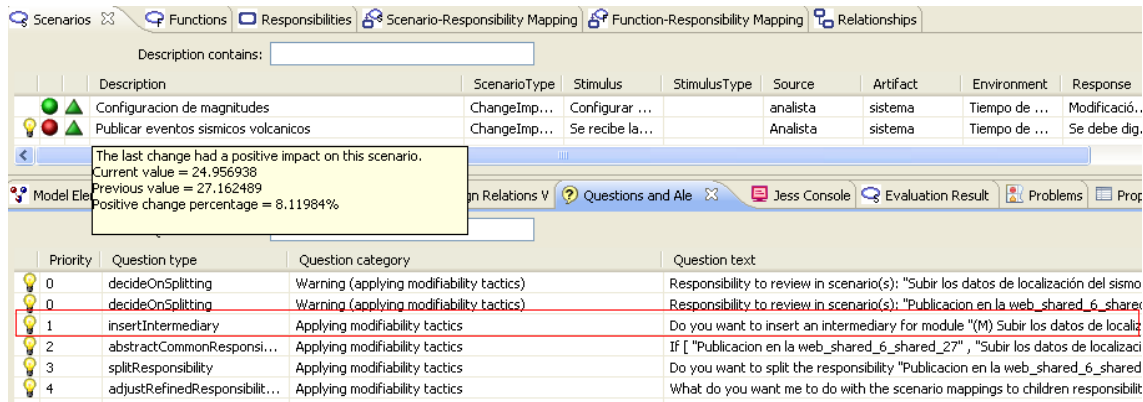


Ilustración 76. Quinta iteración: Evaluación de los escenarios.

Se debe seguir reduciendo el costo. Para esto aplicamos la táctica 1 "InsertIntermediary" para insertar un intermediario en el módulo "Subir los Datos de Localización del sismo del archivo Sfile a la base de datos MySQL (sismología). Con un costo de 30. Con esto se estima reducir el costo del escenario de "24.956937499999995" persona-día a "19.057045673076928" persona-día.

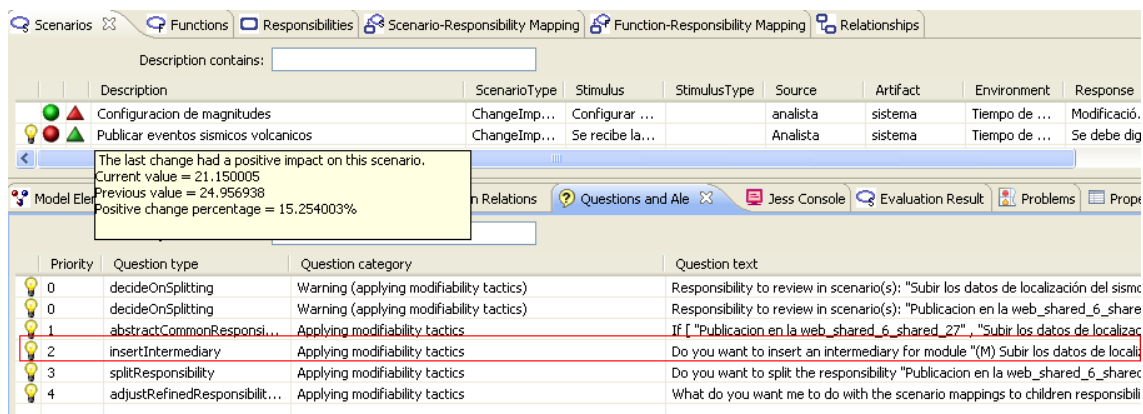


Ilustración 77. Sexta iteración: Evaluación de los escenarios.

En la ilustración No. 77 se ve que el costo se redujo de 24.956938 a 21.150005 persona-día.

Y por último aplicamos la táctica 2 “insertIntermediary” para insertar un intermediario del módulo “Subir los Datos de Localización del sismo del archivo sFile a la base de MySQL Datos de (sismología). \_Intermediary\_14” con un costo de cambio de "9.7000000000000005" día-persona. Ver Ilustración No. 78.

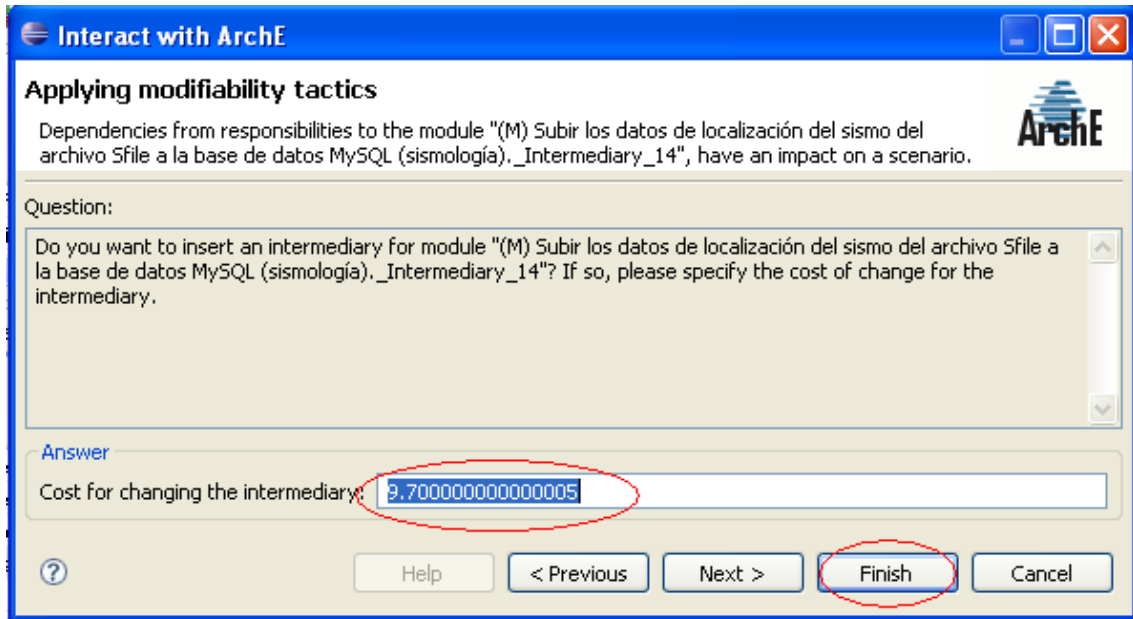


Ilustración 78. Sexta iteración: Aplicación de la táctica 2 “insertIntermediary”.

Evaluando esta última táctica los dos escenarios se satisfacen. Se puede Ver en Ilustración No 79 que aparecen los dos círculos y los dos triángulos en color verde.







del boletín, envió del fax, localización del sismo y llamadas telefónicas automáticas.

Cuando se va a publicar un evento sísmico volcánico el sistema deberá llamar al servicio web compuesto para realizar el siguiente flujo: utilizara el servicio web1 para que el usuario se autentique y el servicio web2 para la creación del archivo Sfile con los datos del formulario. Utilizará el servicio web1 para subir los datos del archivo Sfile a la base de datos Mysql y el servicio web2 para publicar el evento sísmico en la página web. De esta manera se satisface el escenario 2.

## 8 CONCLUSIONES Y TRABAJOS FUTUROS.

### 8.1 CONCLUSIONES.

El procedimiento que se haga ya sea automático o manual para obtener el diseño de una arquitectura, teniendo en cuenta los atributos de calidad es muy importante, ya que con un buen diseño arquitectónico se puede desarrollar un software de calidad. Esto también conlleva a lograr una certificación de calidad del procedimiento para la creación del software.

Debido a que es muy difícil que un arquitecto tenga todo el conocimiento y experiencia sobre los atributos de calidad y sus tácticas para realizar un buen diseño arquitectónico, se han desarrollado herramientas como ArchE que ayudan al arquitecto a tomar decisiones y a seleccionar la mejor arquitectura.

El diseño arquitectónico SOA inicial que sugerí, era solo para la publicación de eventos sísmicos tectónicos. Al querer incluir otra funcionalidad como la de publicar eventos sísmicos volcánicos, la herramienta ArchE me ayudo a la modificación de la arquitectura aplicando el atributo de calidad de modificabilidad. En mi concepto personal pensaba que ArchE solo servía para ayudar a validar arquitecturas clásicas, como cliente servidor, de dos capas, etc. Pero con este trabajo concluyo que ArchE sirve para evaluar cualquier arquitectura.

### 8.2 TRABAJOS FUTUROS.

Para la ampliación de esta investigación se haría el desarrollo e implementación de este sistema con software libre.

En mi concepto personal y que tendría que evaluar con los sismólogos, es la adición de un nuevo módulo que automáticamente validaría el tipo de sismo ya sea tectónico, volcánico o tsunami.

Ofrecer este sistema a las redes sismológicas de Panamá, México, Managua, Guatemala, Portugal, Andalucía y Noruega. Países que utilizan la herramienta SEISAN para el análisis de eventos sísmicos.

## 9 LISTADO DE SIGLAS, ABREVIATURAS Y ACRÓNIMOS.

### **Arquitectónico.**

Adj. De la arquitectura o relativo a este arte.

### **Arquitectura.**

Diseño e implementación de estructuras de software de alto nivel.

### **BPM (Business Process Management).**

Una nueva categoría de software empresarial que permite a las empresas modelar, implementar y ejecutar conjuntos de actividades interrelacionadas.

### **BRE (Business Rule Engine).**

Es un software que permite a los no programadores agregar o cambiar la lógica de negocio en un sistema de gestión de procesos empresariales (BPM).

### **BUS.**

Conjunto de líneas conductoras de hardware utilizadas para la transmisión de datos entre los componentes de un sistema informático.

### **CASOS DE USO.**

Documentan el comportamiento de un sistema desde el punto de vista del usuario.

### **CEP (Complex Event Processing).**

Permite a los responsables de la toma de decisiones definir, detectar y analizar rápidamente la información que necesitan para garantizar una acción pertinente y oportuna.

### **Colapso.**

Caída total o parcial de una vivienda, edificio o elemento de los mismos, como una viga o una columna.

### **EDM (Enterprise decision management).**

Es un enfoque para automatizar y mejorar las decisiones de negocios, provee de un manejo centralizado de la lógica de negocios manejados por las aplicaciones (JBoss Rules).

**Epicentro.**

Punto de la superficie de la Tierra bajo el cual tiene origen un terremoto.

**ESB (Enterprise Service Bus).**

Plataforma de integración que enruta y transforma mensajes en diversos protocolos de comunicación enviados entre las aplicaciones.

**Escenario.**

Es una breve descripción de la interacción de los usuarios con el desarrollo del sistema.

**ESB (Enterprise Service Bus).**

Ofrece conexión entre aplicaciones dentro de una empresa, permiten la administración y el monitoreo de interacciones entre aplicaciones.

**Extraew.**

Es el nombre de la rutina que extrae las trazas del evento sísmico del sistema de adquisición de datos.

**Fax (facsimil).**

Es la transmisión telefónica de material escaneado impreso (tanto texto como imágenes), normalmente a un número de teléfono conectado a una impresora o a otro dispositivo de salida.

**FIFO (First In, First Out)**

Primero en entrar, primero en salir, es un conjunto de actividades o acciones que contienen un determinado orden y que son atendidos conforme van llegando a la cola de procesos.

**GMT (Generic Mapping Tools).**

Es una colección de programas de código abierto para la confección de diagramas y mapas geológicos o geográficos.

**Hipocentro.**

Es el punto interior de la Tierra donde se inicia un movimiento sísmico o terremoto.

**Interplaca.**

Es el roce entre dos placas tectónicas, por ejemplo entre la placa de Nazca y la Sudamericana.

**Isotrópico.**

Son materiales que tienen las mismas propiedades físicas en todas las direcciones.

**Middleware.**

Es un software que conecta o aplicaciones para que puedan intercambiar datos.

**Mulplt.**

Se utiliza para el trazado y análisis de la señal perteneciente al evento sísmico, Se utiliza con HYP para realizar la localización del hipocentro.

**Onda P.**

Las ondas P (primarias o primae del verbo griego) son ondas longitudinales o compresionales, lo cual significa que el suelo es alternadamente comprimido y dilatado en la dirección de la propagación.

**Onda S.**

Las ondas S (SECUNDARIAS) son ondas transversales o de corte, lo cual significa que el suelo es desplazado perpendicularmente a la dirección de propagación, alternadamente hacia un lado y hacia el otro.

**Password.**

Clave o contraseña que identifica a un usuario en un sistema.

**PostCondición.**

Describe lo que debe hacerse cuando el proceso a terminado.

**Protocolo.**

Son estándares o reglas que controlan la conexión, comunicación y transferencia de datos entre dos puntos.

**SCA (Service Component Architecture).**

Es un conjunto de especificaciones que describen el modelo de creación de aplicaciones y sistemas mediante una arquitectura orientada a servicios.

### **SEISAN.**

Es un software para análisis de terremotos. Fue desarrollado en 1992 por los Dres.: Jens Havskov y Terje UTHEIM, de la universidad de Bergen, Noruega.

### **Sfile.**

Es un archivo que contiene todos los datos de localización de un evento sísmico. Este archivo es generado después que se analiza el sismograma con la rutina mulplt de seisan.

### **Sismógrafos.**

Consiste de una pequeña 'masa de prueba', confinada por fuerzas eléctricas, manejada por electrónica sofisticada. Cuando la Tierra se mueve, electrónicamente se trata de mantener la masa fija a través de la retroalimentación del circuito. La cantidad de fuerza necesaria para conseguir esto es entonces registrada.

### **Sismograma.**

Un sismograma es un registro del movimiento del suelo llevado a cabo por un sismógrafo. La energía medida en un sismograma resulta de fuentes naturales como son los sismos (o terremotos), o de fuentes artificiales como son los explosivos (sismos inducidos).

### **SOA (Service Oriented Architecture).**

Es una arquitectura basada en servicios web reutilizables con interfaces públicas, donde los proveedores de servicios y consumidores interactúan en forma desacoplada para realizar los procesos de negocio.

### **Trazas sísmicas.**

Una traza sísmica representa la respuesta del campo de ondas elásticas a la velocidad y los contrastes de densidad a lo largo de las interfaces de las capas de rocas o los sedimentos a medida que la energía viaja a través de las capas del subsuelo, desde una fuente hasta un receptor o un arreglo de receptores.

### **Vista arquitectónica.**

Es la representación de un sistema completo.



## 10 BIBLIOGRAFIA.

Jeff Davis. 2009. SOA: Open Source: Anaya Multimedia.

jBoss project. Obtenido de <http://jbpm.jboss.org/> .

Synapse. 2012. Apache Synapse. Obtenido de <http://synapse.apache.org/> .

EsperTech. Obtenido de <http://esper.codehaus.org/>

Ws02 Governance Registry. Obtenido de <http://wso2.com/products/governance-registry/>

Apache Tuscany. Obtenido de <http://tuscany.apache.org/>

Juan Garbajosa Sopeña, Francisco Javier Soriano y otros. Tecnologías software orientada a servicios: Fundación Madrid para el Conocimiento Velázquez, 76. E-28001 Madrid.

ProgrammableWeb. Obtenido de [http://www.programmableweb.com/category/all/apis?order=field\\_popularity](http://www.programmableweb.com/category/all/apis?order=field_popularity)

Microsoft Developer Network. Obtenido de <http://msdn.microsoft.com/en-us/library/ff701710.aspx>

Buffa, W. (2007). Física. México: Pearson Prentice Hall.

Sarachaga, M. H. (2008). Conceptos básicos de sismología para Ingenieros. Perú: Centro Peruano Japonés de Investigaciones Sísmicas y Mitigación de Desastres.

Castro, J. M., & Jiménez, Z. (1986). Terremotos y ondas sísmicas: una breve introducción. México: Universidad Nacional Autónoma de México.

Valdivia, L. A., Plaza, & Valdés. (1996). Propiedades Geofísicas de la Tierra. México: Plaza y Valdés Editores.

Bass, Len; Clements, Paul; Kazman, Rick. (2003). Software Architecture in Practice (Second Edition ed.). Boston - MA, USA: Addison Wesley.

Muhammad Ali Babar, Alan W. Brown, Ivan Mistrik. (2013). Agile Software Architecture. Morgan Kaufmann.

Robert Hanmer. (2013). Pattern-Oriented Software Architecture For Dummies. For Dummies.

Richard N. Taylor; Nenad Medvidovic; Eric M. Dashofy (2009). SOFTWARE ARCHITECTURE: Foundations, Theory, and Practice. John Wiley & Sons.

Sommerville, Ian. (2011). Ingenieria de software. Pearson Educación.