

Universidad Autónoma de Madrid



Escuela Politécnica Superior
Seminario en Ingeniería Informática

**SISTEMA EVOLUTIVO DE
GESTIÓN DE LA CONFIGURACIÓN
DEL SOFTWARE**

Jorge de las Peñas Plana

Directora: Miren Idoia Alarcón Rodríguez

ÍNDICE

- ⇒ Estado de la Cuestión
- ⇒ Planteamiento del Problema
- ⇒ Solución Propuesta
- ⇒ Análisis de Viabilidad
- ⇒ Conclusiones

ESTADO DE LA CUESTIÓN

⇒ Estado de la Cuestión

⇒ Sistemas de Gestión de la Configuración del Software

⇒ Redes Neuronales Artificiales

⇒ Espacio de Investigación

⇒ Planteamiento del Problema

⇒ Solución Propuesta

⇒ Análisis de Viabilidad

⇒ Conclusiones

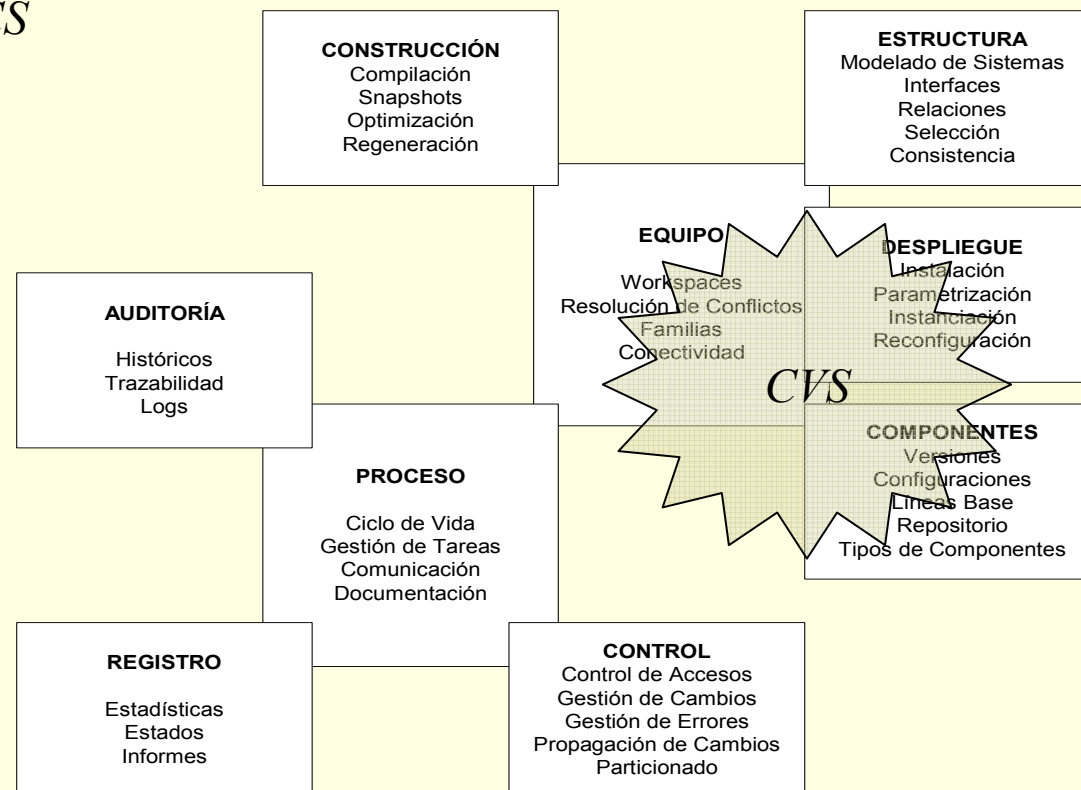
ESTADO DE LA CUESTIÓN

Sistemas de Gestión de la Configuración del Software

⇒ *Gestión de la Configuración del Software (GCS):*

⇒ Proceso de **identificar** y **definir** los objetos de un sistema, **controlando** sus cambios a lo largo de su ciclo de vida, **registrando** e **informando** de los estados de dichos objetos y de los cambios y verificando su corrección y su completitud.

⇒ *Sistema de GCS*



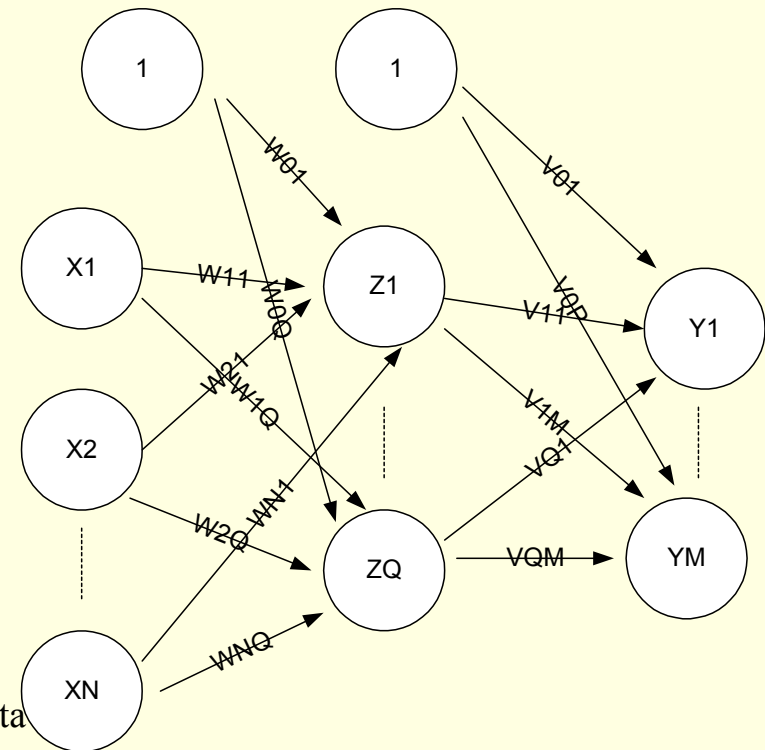
ESTADO DE LA CUESTIÓN

Redes Neuronales Artificiales

- ⇒ Dentro de la *Minería de Datos*, y en particular de las técnicas de *Aprendizaje Automático*.
- ⇒ Modelos computacionales compuestos de unidades de proceso paralelo (“neuronas”) densamente interconectadas. Inspiración modelo biológico
 - ⇒ *Perceptrón Multicapa (MLP)*

Aspectos de Diseño

- ⇒ Algoritmo (Backpropagation) y Regla de Aprendizaje (Delta)
- ⇒ Arquitectura de la Red
 - ⇒ Capas Ocultas. Número de Nodos de la Capa Oculta
- ⇒ Elección de Patrones ejemplo (Sobreaprendizaje)
- ⇒ Condición de Finalización y Factor de Aprendizaje
- ⇒ *Librería de Uso*: TORCH (API C++ Aprendizaje Automático)



ESTADO DE LA CUESTIÓN

Espacio de Investigación

- ⇒ *El cruce entre la GCS y el Aprendizaje Automático es el espacio (“gap”) que se va a investigar con este trabajo para tratar de elaborar un modelo de GCS y de Redes Neuronales, con el fin de aportar a una organización **conocimiento de utilidad** para su futuro.*
 - ⇒ No se trata por tanto de una innovación en el campo de las Redes Neuronales y el Aprendizaje Automático, si no de la aplicación de estas estructuras para añadir valor a procesos.
 - ⇒ Tampoco es una extensión a un sistema de Gestión de la Configuración del Software. Se trata de dar un nuevo modelo de procesos de GCS.

PLANTEAMIENTO DEL PROBLEMA

⇒ Estado de la Cuestión

⇒ Planteamiento del Problema

⇒ Hipótesis

⇒ Planteamiento y Orientación a la Solución

⇒ Solución Propuesta

⇒ Análisis de Viabilidad

⇒ Conclusiones

PLANTEAMIENTO DEL PROBLEMA

Hipótesis

⇒ Hipótesis Principal:

H: La Gestión de la Configuración del Software, aplicando técnicas de Aprendizaje Automático, puede ayudar, no sólo a controlar el presente y el pasado de la evolución de los sistemas software, sino también a *detectar tendencias y patrones que impliquen conocimiento de utilidad para ayudar a una organización en la toma de decisiones futuras.*

⇒ Subhipótesis:

H₁: La Gestión de la Configuración del Software, aplicando técnicas de Aprendizaje Automático, *puede ayudar en los procesos de evaluación de los diferentes usuarios de los diferentes productos.* Al mantener la información histórica de todas las acciones que han realizado los diferentes usuarios de los diferentes sistemas de la compañía y la información de las evaluaciones que han obtenido, se puede tratar de inferir cuál será la evaluación en un periodo.

H₂: La Gestión de la Configuración del Software, aplicando técnicas de Aprendizaje Automático, *puede ayudar en la detección de acciones correctoras a nivel general,* con el fin de trazar acciones de mejora general de los sistemas a partir de la detección de tendencias y patrones en las causas que provocan los errores de una compañía.

PLANTEAMIENTO DEL PROBLEMA

Planteamiento y Orientación a la Solución

Modelización y Análisis

- Organización
- Procesos y Entidades GCS

Arquitectura de la Solución

- Funcional
- Tecnológica

Descubrimiento de Datos

Generación de Herramienta

Análisis de Viabilidad

SOLUCIÓN PROPUESTA

- ⇒ Estado de la Cuestión
- ⇒ Planteamiento del Problema

⇒ Solución Propuesta

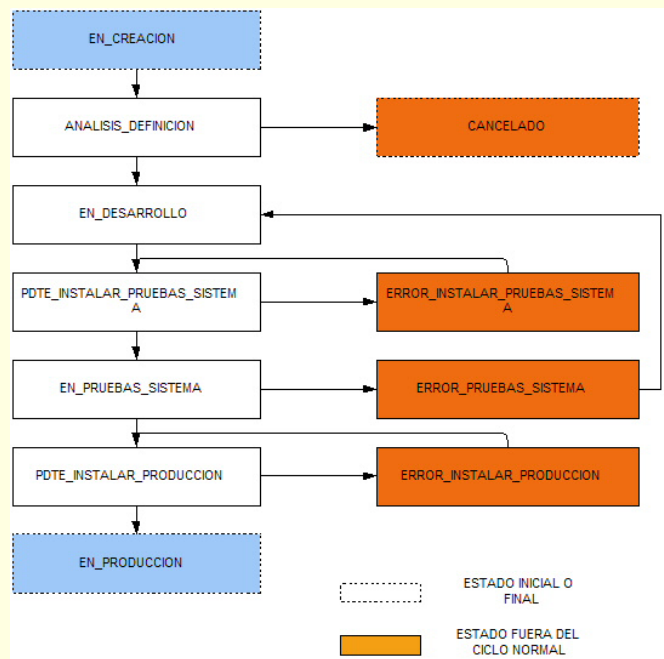
- ⇒ Procesos de Gestión de Cambios
- ⇒ Arquitectura Funcional. Arquitectura Tecnológica.
- ⇒ Componentes: Cliente, Servidor, Base de Datos, Repositorio. Integración
- ⇒ Descubrimiento de Datos
 - ⇒ Red Neuronal de Clasificación de Usuarios según sus Errores
 - ⇒ Minería de Datos sobre las Causas que provocan Errores
- ⇒ Funcionalidad

- ⇒ Análisis de Viabilidad
- ⇒ Conclusiones

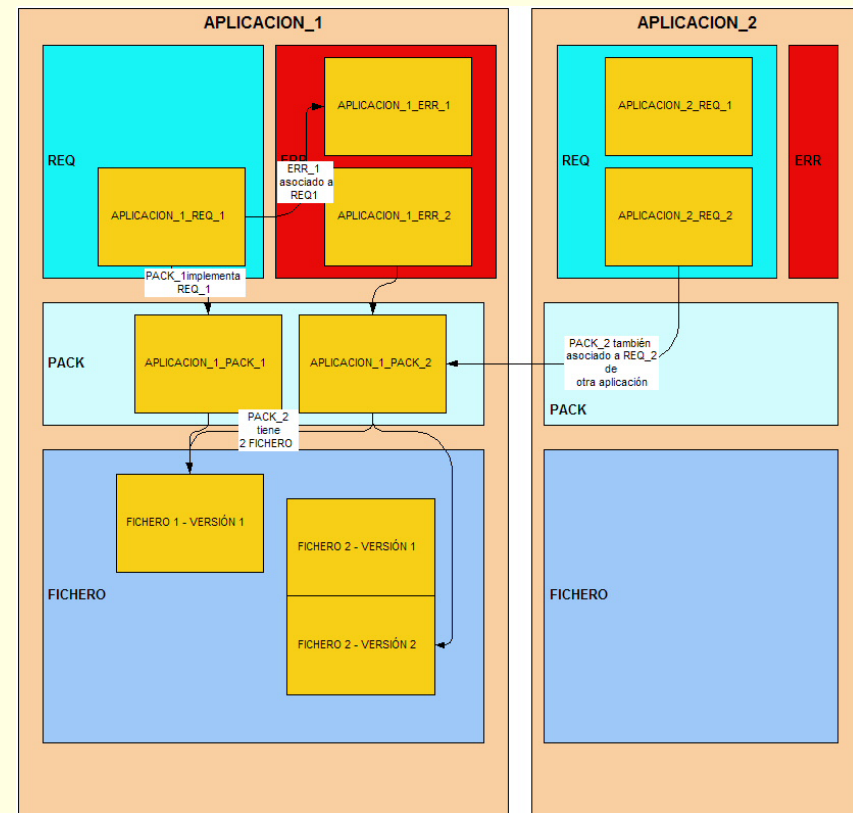
SOLUCIÓN PROPUESTA

Procesos de Gestión de Cambios

- ⇒ *Gestión de Versiones: FICHERO*
- ⇒ *Gestión de Releases: PACK*
- ⇒ *Gestión de Requerimientos: REQ*
- ⇒ *Gestión de Errores: ERR*



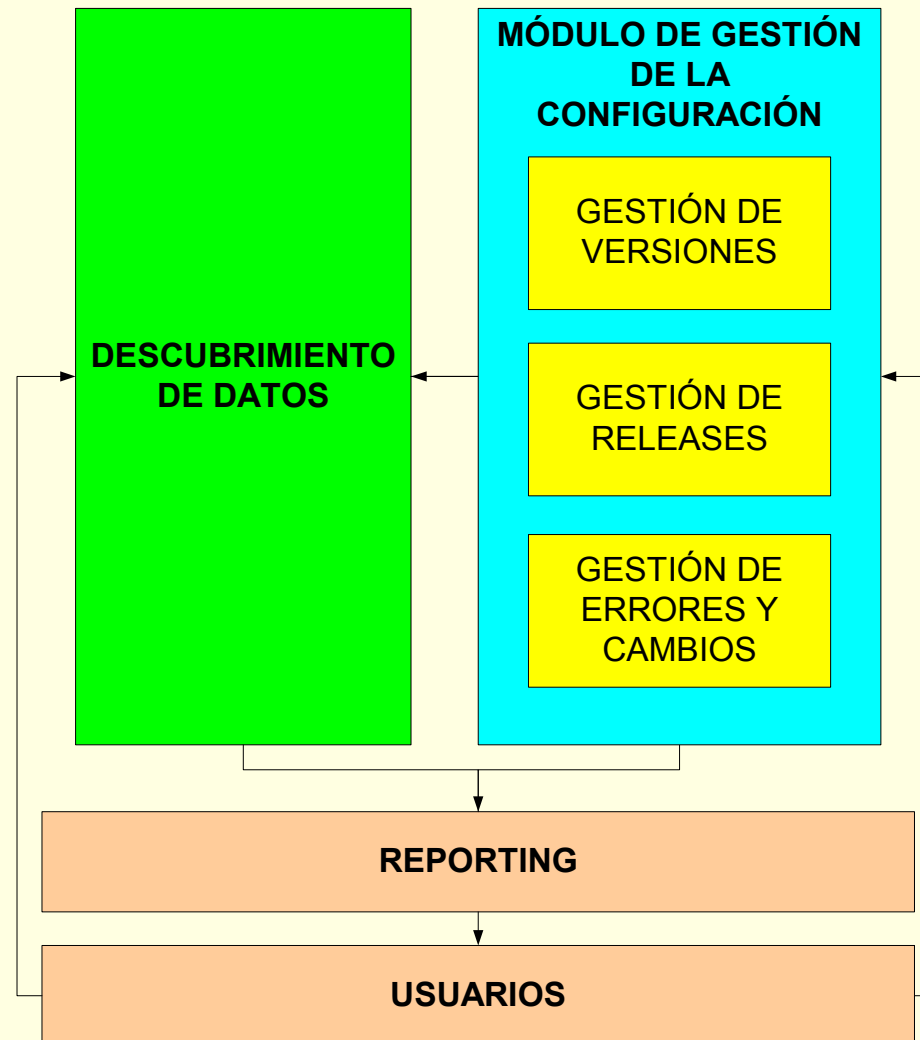
Ciclo de Vida



Jerarquía de Unidades de Control

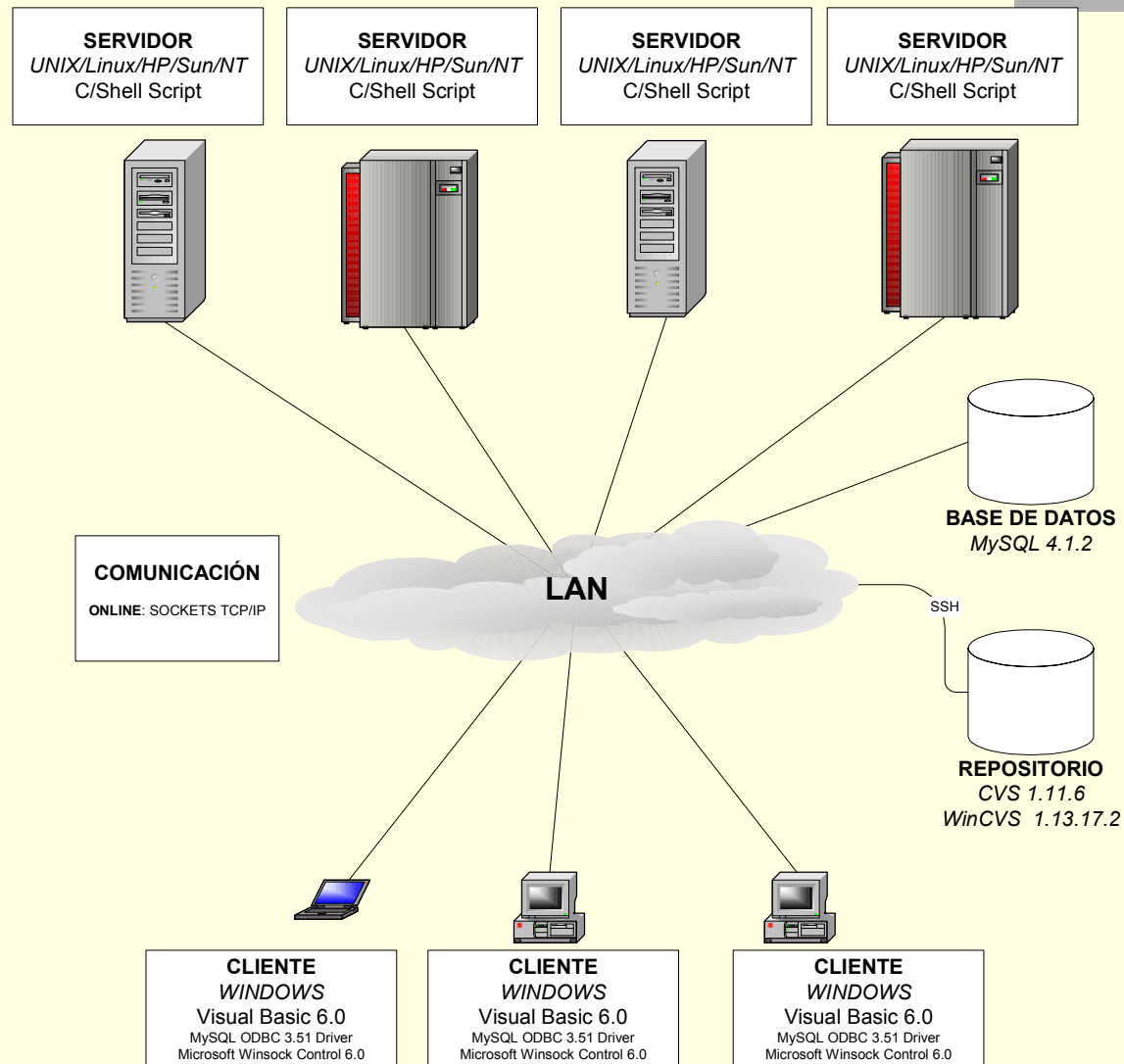
SOLUCIÓN PROPUESTA

Arquitectura Funcional



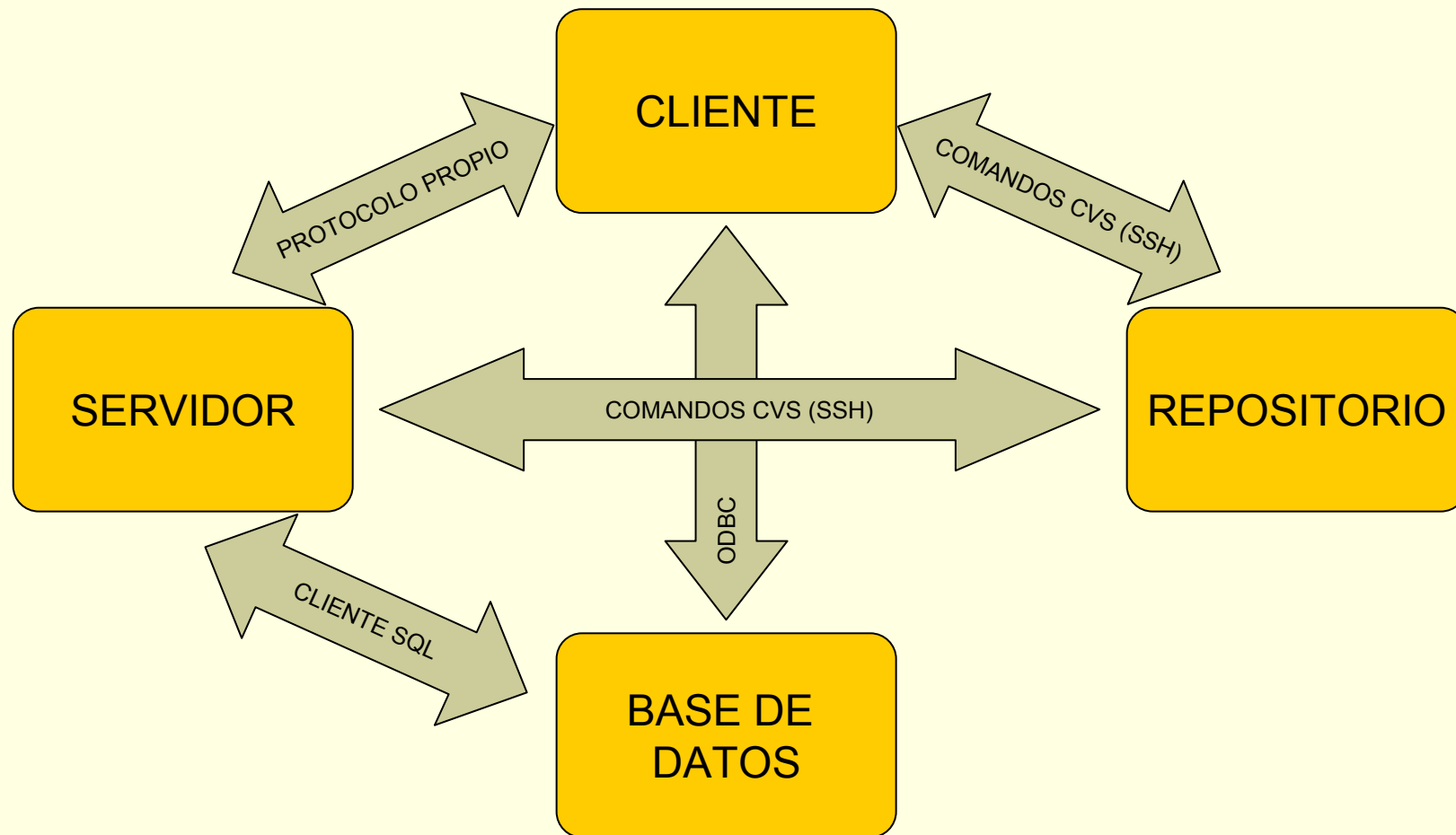
SOLUCIÓN PROPUESTA

Arquitectura Tecnológica



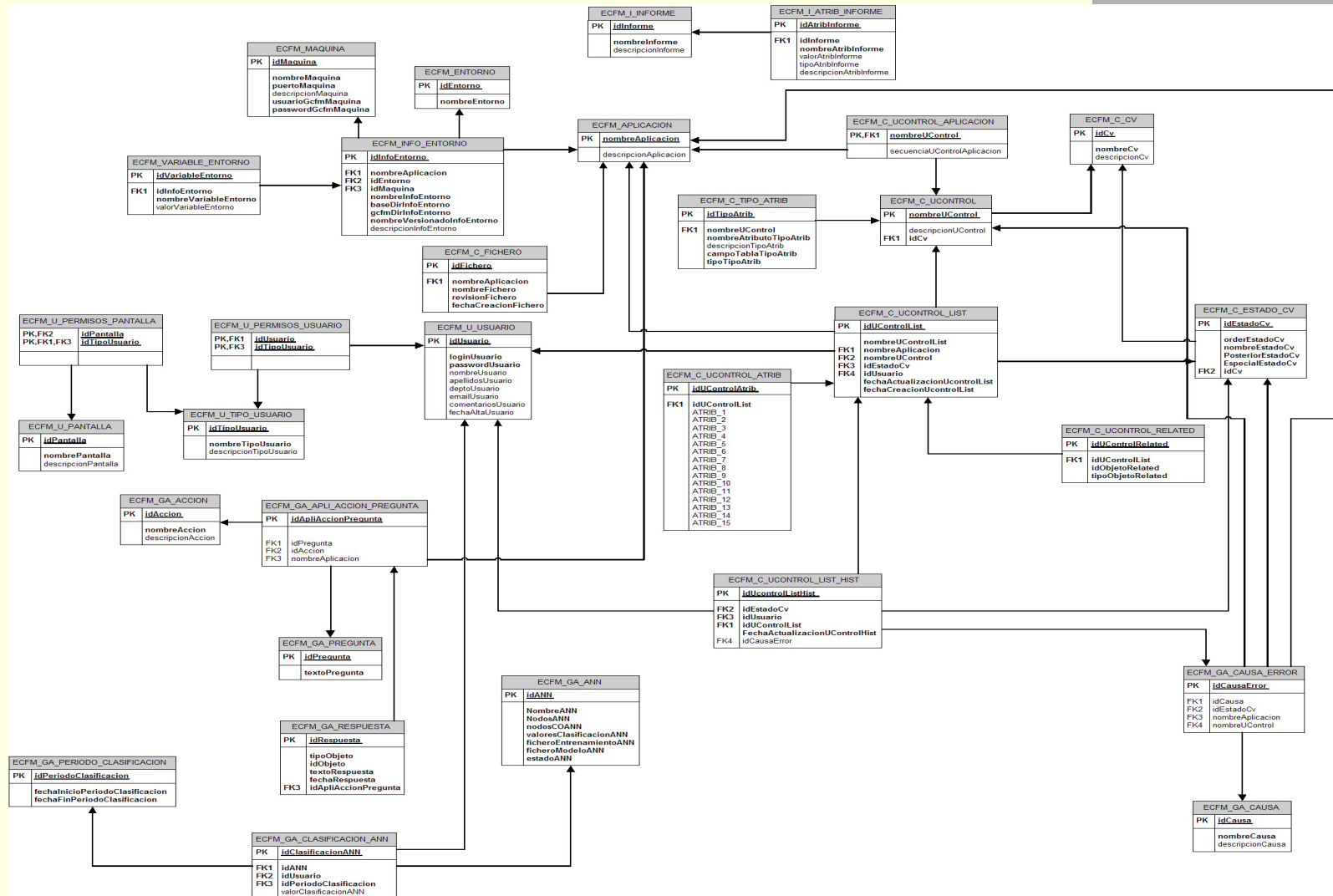
SOLUCIÓN PROPUESTA

Componentes



SOLUCIÓN PROPUESTA

Componentes: Modelo de Datos



SOLUCIÓN PROPUESTA

Descubrimiento de Datos

⇒ La parte de *Descubrimiento de Datos* se ocupa de aportar al sistema la funcionalidad necesaria para inferir conocimiento útil para el futuro (cubrir la Hipótesis de Trabajo H). Consistirá en:

⇒ *Red Neuronal de Clasificación de Usuarios* (Subhipótesis H_1)

⇒ Detectar patrones en información histórica de GCS que nos permitan predecir las evaluaciones de usuarios en un determinado periodo de tiempo.

⇒ *Minería de Datos sobre las Causas de los Errores*
(Subhipótesis H_2)

⇒ Identificar cuáles son las causas en un determinado periodo de tiempo que pueden ocasionar a futuro evaluaciones no adecuadas, de manera que se puedan anticipar acciones correctoras.

SOLUCIÓN PROPUESTA

Red Neuronal de Clasificación de Usuarios según sus Errores

⇒ Perceptrón Multicapa

⇒ Nodos de la Capa de Entrada:

2*Número de Estados de Error de cada Unidad de Control y Aplicación + 2 Totales (Por Usuario/Periodo de tiempo)

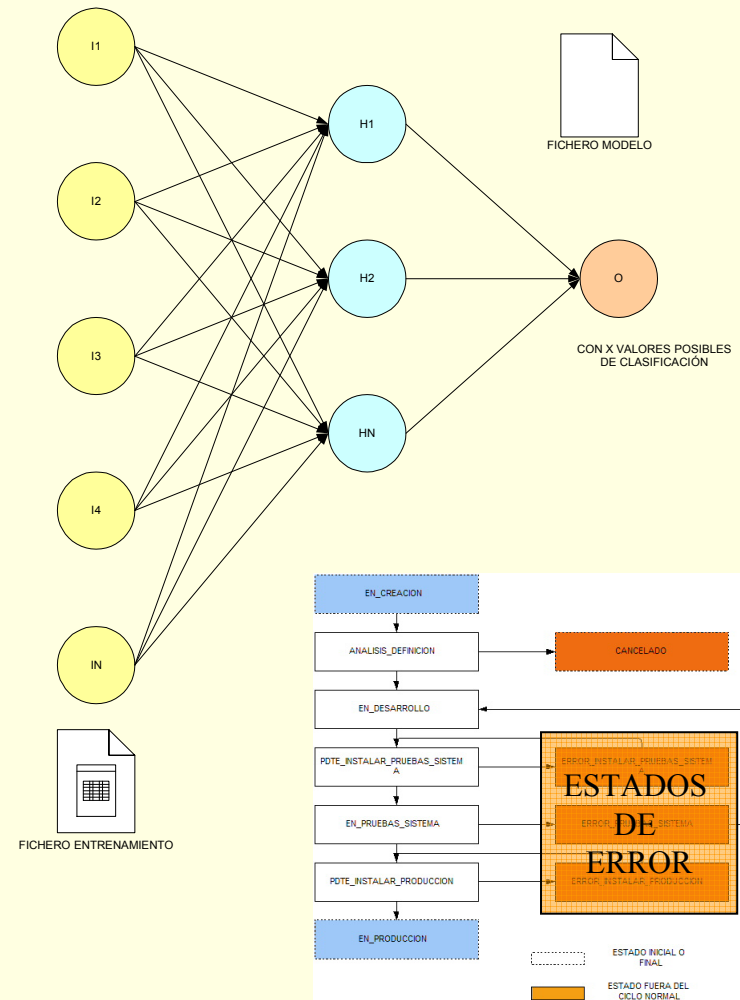
⇒ Número total de Errores

⇒ Número total de Iteraciones

⇒ Nodos en Unidad de Salida: 1 con tantos valores de clasificación como posibles Evaluaciones.

⇒ Nodos de la Capa Oculta a Determinar

⇒ Iteraciones y tasa de Aprendizaje a determinar



SOLUCIÓN PROPUESTA

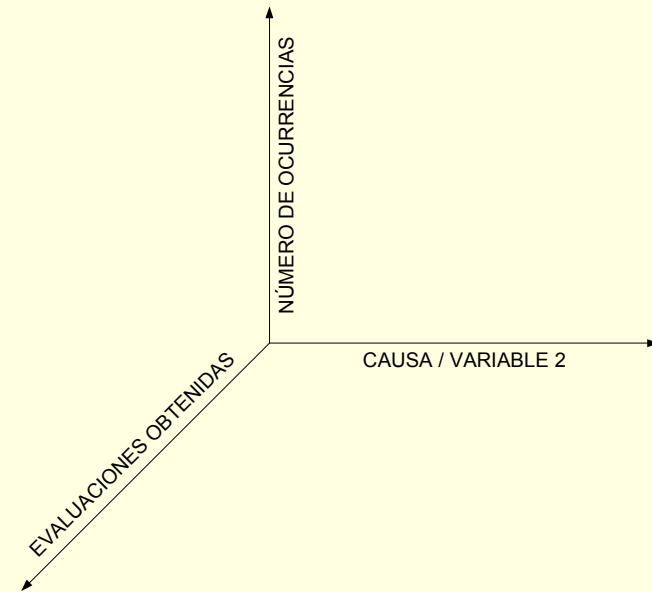
Minería de Datos sobre las Causas que provocan Errores

⇒ En los *cambios de Estado a ERROR*, se incluye la causa que ha provocado el error.

⇒ La *Tipificación de Causas* dependerá de:

- ⇒ APLICACIÓN
- ⇒ UNIDAD DE CONTROL
- ⇒ ESTADO DE ERROR

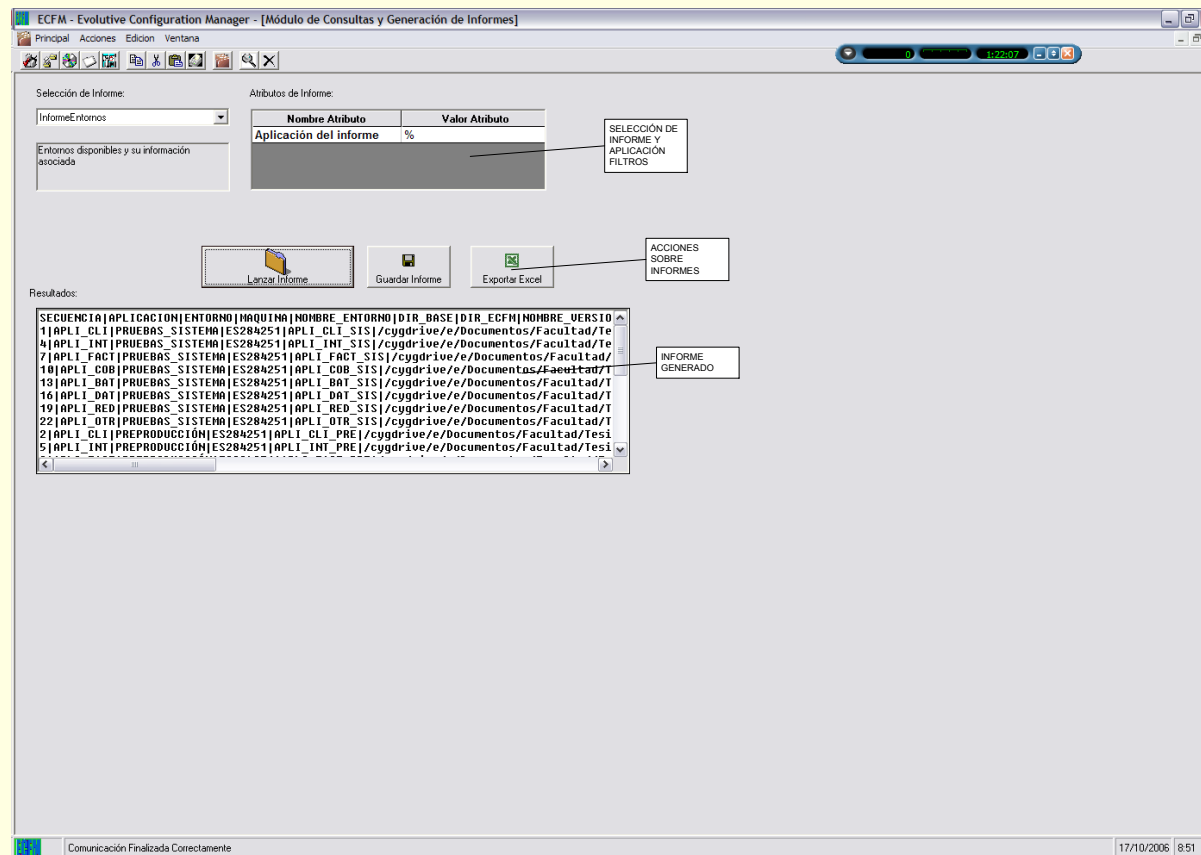
⇒ Se cruzarán las Causas con los datos de Evaluaciones generados por la Red de Evaluaciones para analizar las que provocan evaluaciones más altas y bajas combinando con: Usuarios, Aplicaciones y Estados de Error



SOLUCIÓN PROPUESTA

Funcionalidad

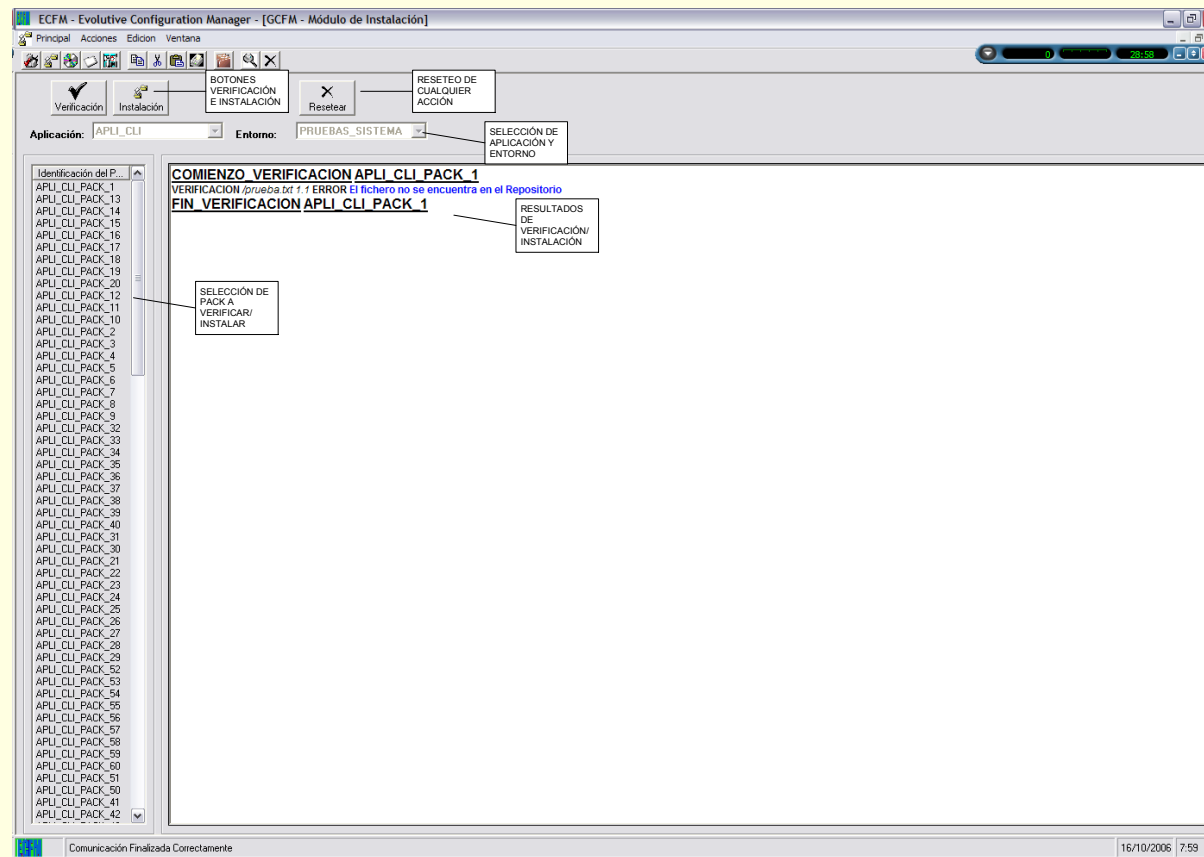
⇒ Módulo de Generación de Informes



SOLUCIÓN PROPUESTA

Funcionalidad

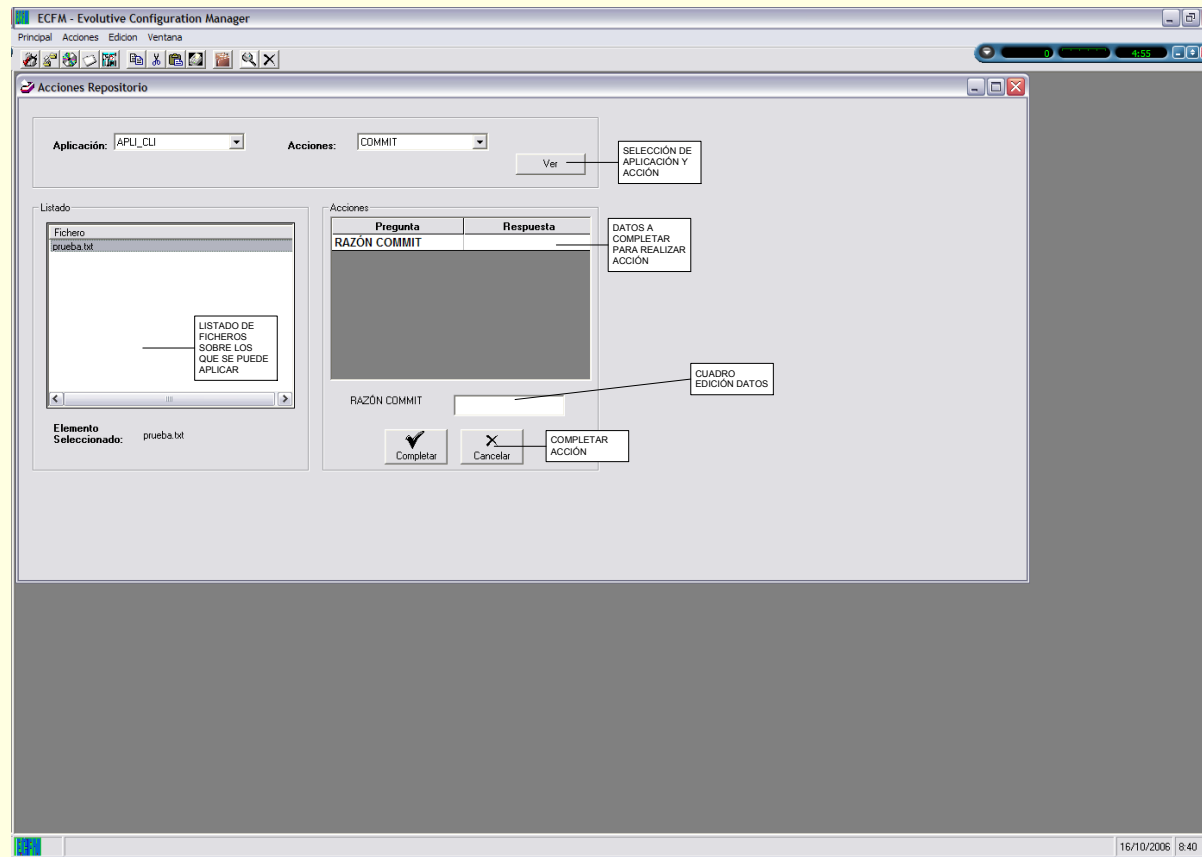
⇒ Módulo de Instalación



SOLUCIÓN PROPUESTA

Funcionalidad

⇒ Módulo de Acciones sobre el Repositorio



SOLUCIÓN PROPUESTA

Funcionalidad

⇒ Módulo de Control del Ciclo de Vida

The screenshot displays the 'Operaciones con Unidades de Control' dialog box within the ECFM - Evolutive Configuration Manager application. The dialog is used for managing control units and their attributes.

Unidad de Control:

- Nombre:
- Estado: Ver Todos
- Título:

Atributos:

Nombre Atributo	Valor Atributo
Determina la Grav	-
Release en la que	-
Release en la que	-
Entorno en el que	-
Descripción de la	-
Estimación de esf	-
Si se cancela, indi	-

Relaciones:

- U.Control/Aplicacion:
- Fichero:

Buttons:

Bottom buttons:

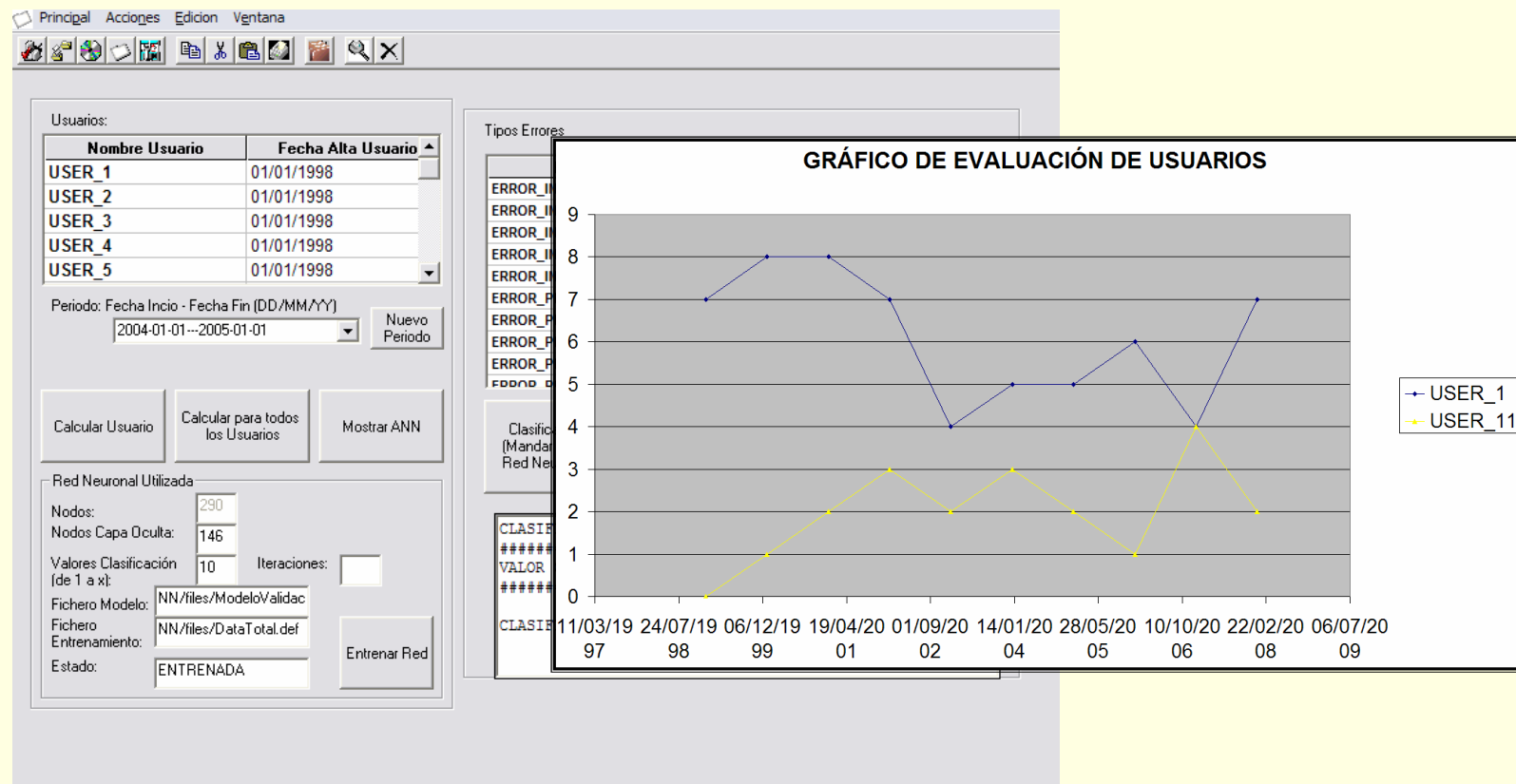
Ciclo Vida Table (Background):

Código	Estado	Título
APLI_CLI_ERR_1	CERRADO	APLI_CLI_ERR_1
APLI_CLI_ERR_10	CERRADO	APLI_CLI_ERR_10
APLI_CLI_ERR_11	CERRADO	APLI_CLI_ERR_11
APLI_CLI_ERR_12	CERRADO	APLI_CLI_ERR_12
APLI_CLI_ERR_13	CERRADO	APLI_CLI_ERR_13
APLI_CLI_ERR_14	CERRADO	APLI_CLI_ERR_14
APLI_CLI_ERR_15	CERRADO	APLI_CLI_ERR_15
APLI_CLI_ERR_16	CERRADO	APLI_CLI_ERR_16
APLI_CLI_ERR_17	CERRADO	APLI_CLI_ERR_17
APLI_CLI_ERR_18	CERRADO	APLI_CLI_ERR_18
APLI_CLI_ERR_19	CERRADO	APLI_CLI_ERR_19
APLI_CLI_ERR_2	CERRADO	APLI_CLI_ERR_2
APLI_CLI_ERR_20	CERRADO	APLI_CLI_ERR_20

SOLUCIÓN PROPUESTA

Funcionalidad

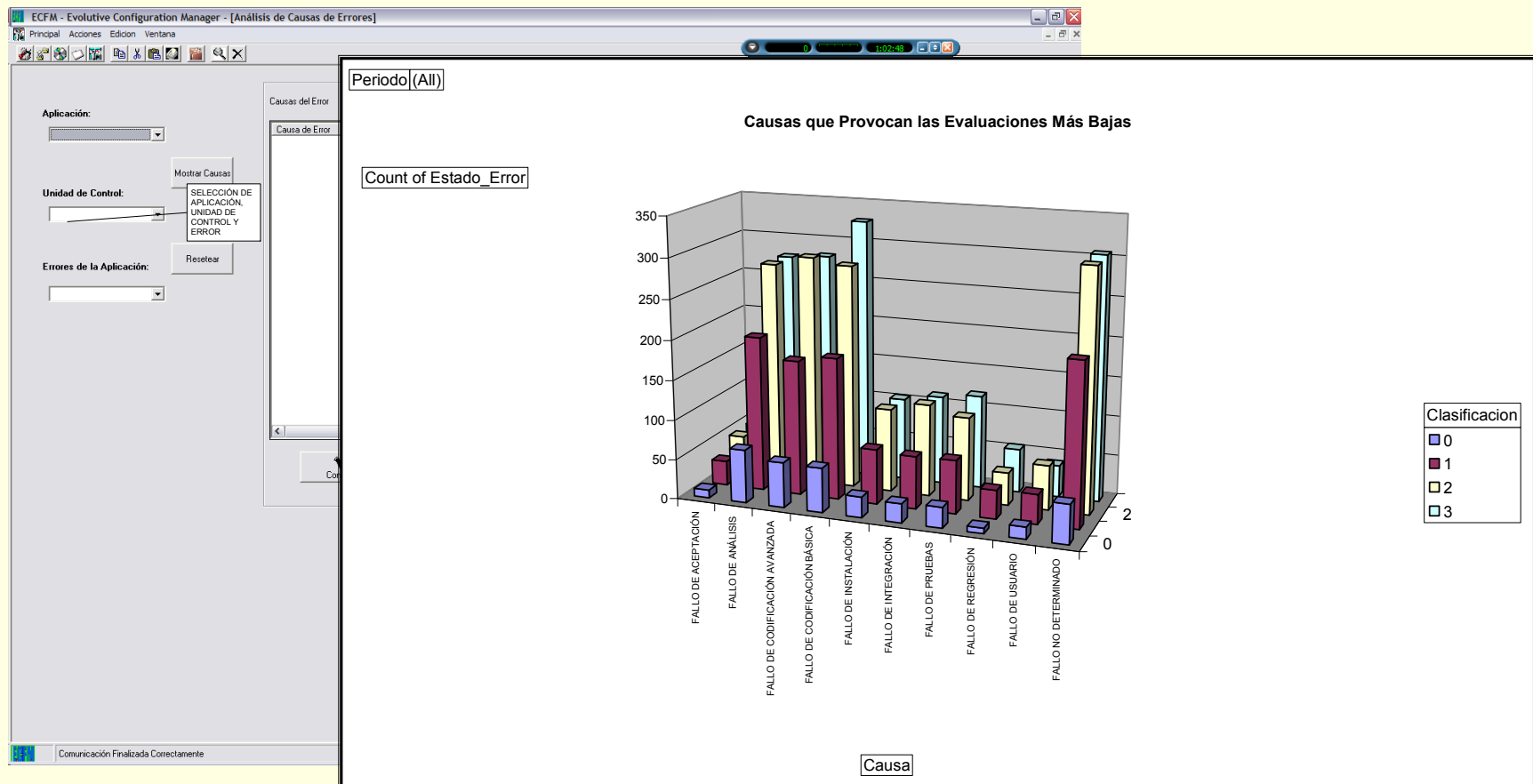
⇒ Módulo de Gestión de Red Neuronal de Evaluación de Usuarios



SOLUCIÓN PROPUESTA

Funcionalidad

⇒ Módulo de Análisis de Causas de Error



ANÁLISIS DE VIABILIDAD

- ⇒ Estado de la Cuestión
- ⇒ Planteamiento del Problema
- ⇒ Solución Propuesta

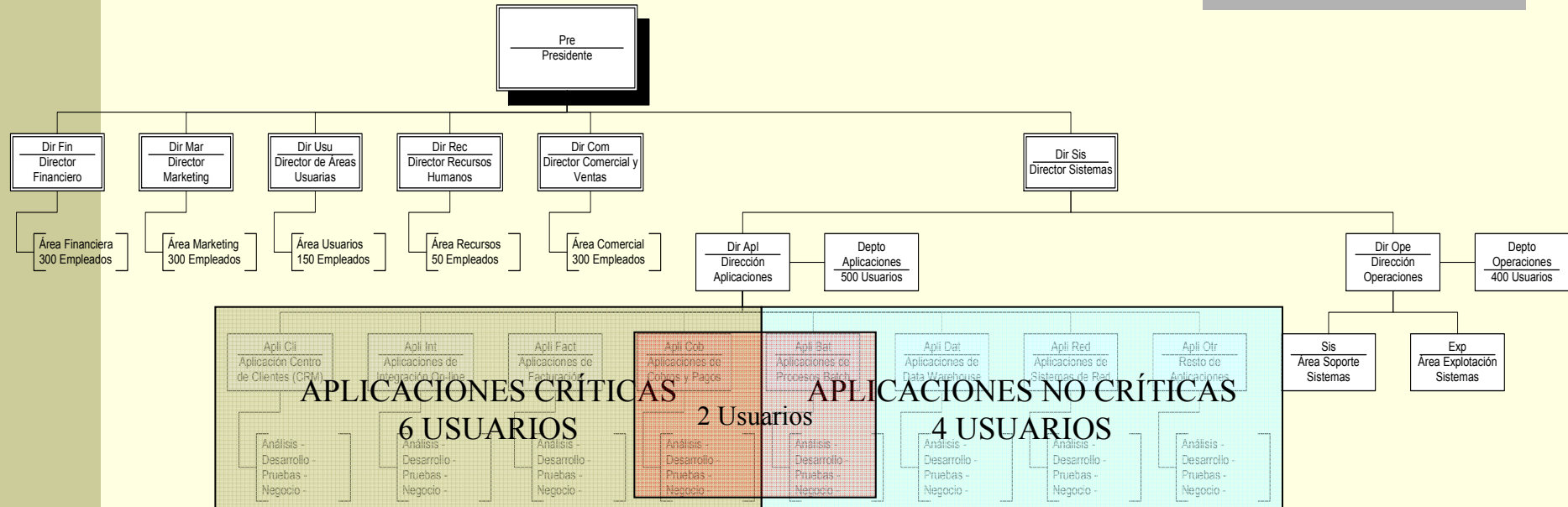
⇒ Análisis de Viabilidad

- ⇒ Modelo de Organización
- ⇒ Procesos de Gestión de Cambios
- ⇒ Tipificación de Causas
- ⇒ Topología de la Red Neuronal de Clasificación de Usuarios
- ⇒ Información Histórica
- ⇒ Experimentos: Ciclo Aleatorio y Ciclo Similar a otro
- ⇒ Distribución y Análisis de Resultados

- ⇒ Conclusiones

ANÁLISIS DE VIABILIDAD

Modelo de Organización



GRUPOS
ADMIN
ANA
DES
PRU
NEG
INST

ENTORNOS
PRUEBAS_SISTEMA
PREPRODUCCIÓN
PRODUCCIÓN

MÁQUINAS
WINDOWS-SERVER
LINUX-SERVER
LINUX-SERVER_2

ANÁLISIS DE VIABILIDAD

Procesos de Gestión de Cambios

⇒ Unidades de Control

⇒ ERR

⇒ REQ

⇒ PACK

⇒ Grupos de Errores

⇒ Errores en etapa de Desarrollo y Pruebas de Sistema:

⇒ ERROR_INSTALAR_PRUEBAS_SISTEMA

⇒ ERROR_PRUEBAS_SISTEMA

⇒ Errores en etapa de Preproducción:

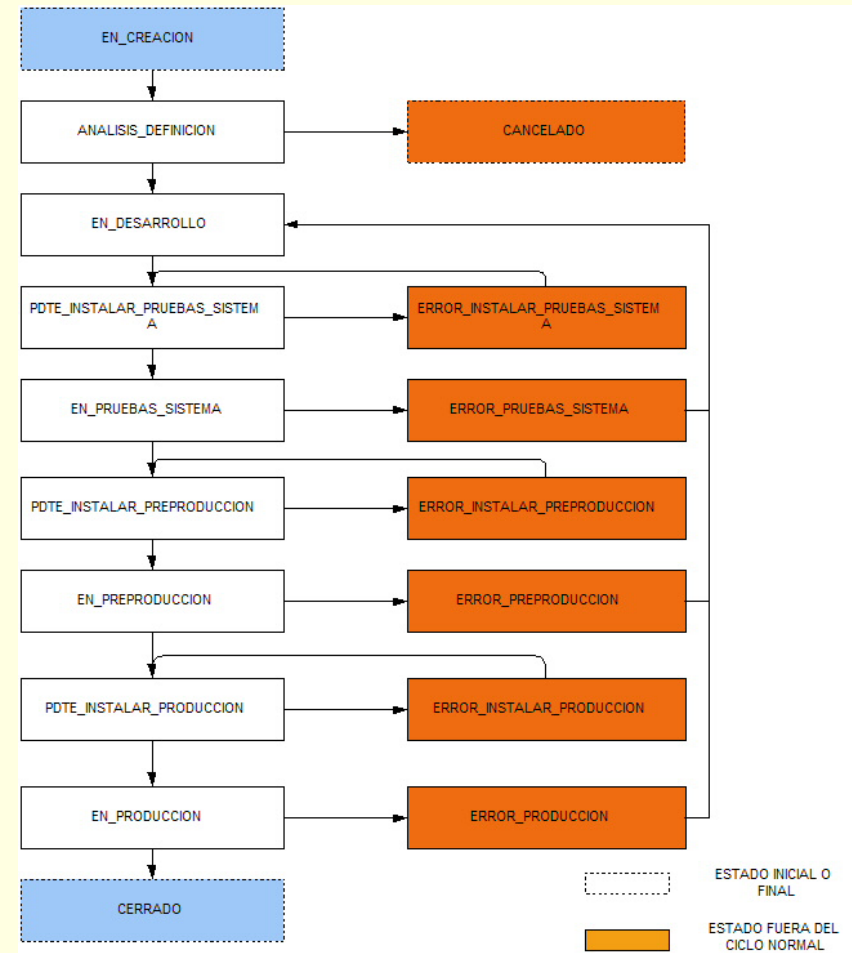
⇒ ERROR_INSTALAR_PREPRODUCCION

⇒ ERROR_PREPRODUCCION

⇒ Error en etapa de Producción:

⇒ ERROR_INSTALAR_PRODUCCION

⇒ ERROR_PRODUCCION



ANÁLISIS DE VIABILIDAD

Tipificación de Causas

⇒ *Causas de Desarrollo*

- ⇒ FALLO NO DETERMINADO
- ⇒ FALLO DE CODIFICACIÓN BÁSICA
- ⇒ FALLO DE CODIFICACIÓN AVANZADA
- ⇒ FALLO DE ANÁLISIS

⇒ *Causas de Pruebas*

- ⇒ FALLO DE INTEGRACIÓN
- ⇒ FALLO DE INSTALACIÓN
- ⇒ FALLO DE PRUEBAS

⇒ *Causas de Usuario*

- ⇒ FALLO DE USUARIO
- ⇒ FALLO DE REGRESIÓN
- ⇒ FALLO DE ACEPTACIÓN

10 causas
* 6 estados posibles de error
* 3 unidades de control
* 8 aplicaciones
= **1440** combinaciones

ANÁLISIS DE VIABILIDAD

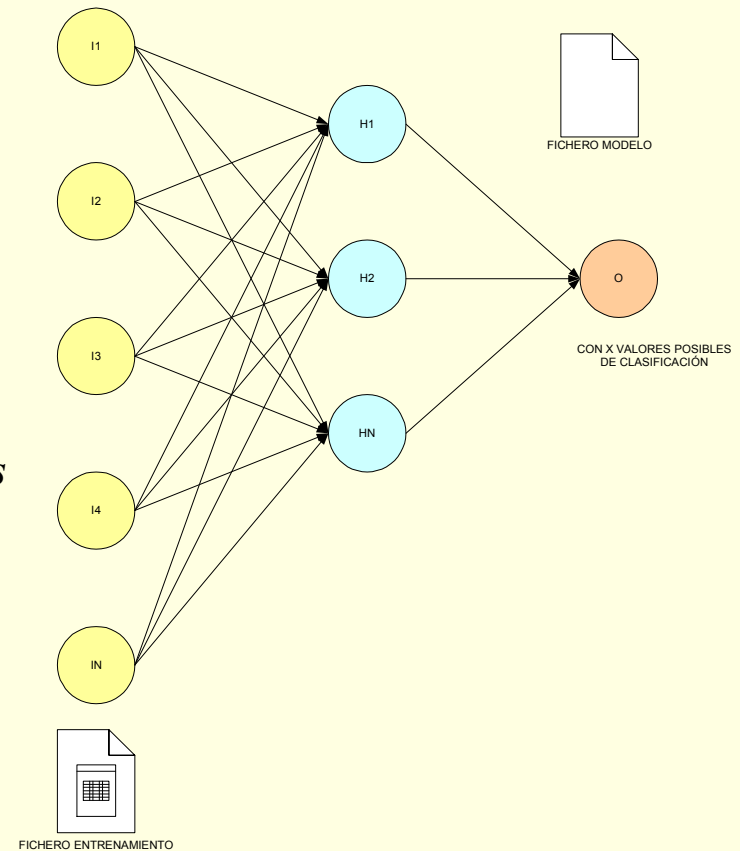
Topología de la Red Neuronal de Clasificación de Usuarios

⇒ *Nodos de la Capa de Entrada*

- ⇒ 8 Aplicaciones
- ⇒ 3 Unidades de Control/Aplicación
- ⇒ 6 Estados Error/Unidad de Control
- ⇒ 2 Unidades de Entrada/Estado Error
 - ⇒ Total de Errores
 - ⇒ Total de Transiciones
- ⇒ **TOTAL: $8*3*6*2+2$ (Totales)= 290 Unidades**

⇒ *Valores de Unidad de Salida*

- ⇒ Evaluaciones de **0 a 9** (10 en total)
- ⇒ **Ejemplos: 38 Usuarios * 8 Periodos = 304**
- ⇒ *Nodos de la Capa Oculta: 146*
- ⇒ *Iteraciones: 1000*
- ⇒ *Tasa de Aprendizaje: 0,0001*



ANÁLISIS DE VIABILIDAD

Información Histórica

DISTRIBUCIÓN UNIDADES DE CONTROL:
2448 Unidades en el listado

Más Críticas: 8 REQ, 20 ERR, 20 PACK Periodo
Menos Críticas: 4 REQ, 10 ERR, 10 PACK Periodo

DISTRIBUCIÓN HISTÓRICOS:
60862 Registros

48307 registros en 8 periodos evaluados
12555 registros en 2 periodos evaluables

Grupos de Valores de Clasificacõ n:

- Evaluaciones Bajas: 0-3 (25%) – Probabilidad Error 1
- Evaluaciones Medias: 4-6 (50%) – Probabilidad Error 0,5
- Evaluaciones Altas: 7-9 (25%) – Probabilidad Error 0

Por Grupo de Error:

- Errores de Desarrollo: 0,7
- Errores de Preproducción: 0,2
- Errores de Producción: 0,1

Causas Origen:

- Causas de Desarrollo: 0,7
- Causas de Pruebas: 0,2
- Causas de Usuario: 0,1

ANÁLISIS DE VIABILIDAD

Experimentos: Ciclo Aleatorio y Ciclo Similar a Otro

⇒ *Predicción de un Ciclo Aleatorio:*

⇒ Para un periodo de tiempo se generan datos históricos **aleatorios** sin seguir patrón de Evaluaciones.

⇒ OBJETIVO: Identificar sentido de las tendencias dados los datos aleatorios.

⇒ *Predicción de un Ciclo Similar a Otro:*

⇒ Los datos históricos de un periodo son **similares** a los de otro anteriormente clasificado.

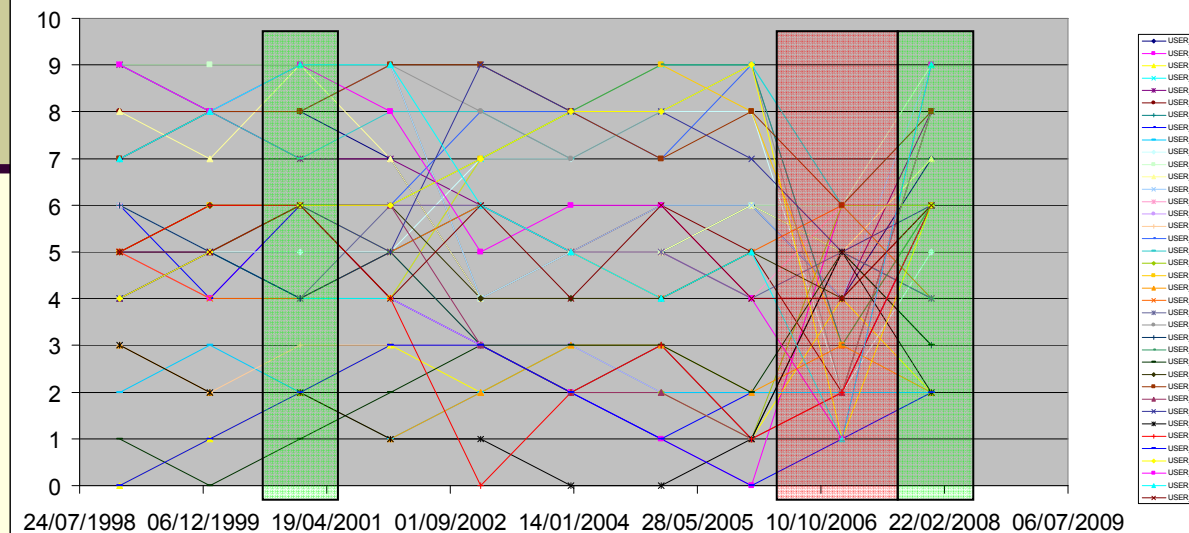
⇒ OBJETIVO: Detectar tendencias similares a periodo clasificado previamente.

ANÁLISIS DE VIABILIDAD

Distribución de Resultados

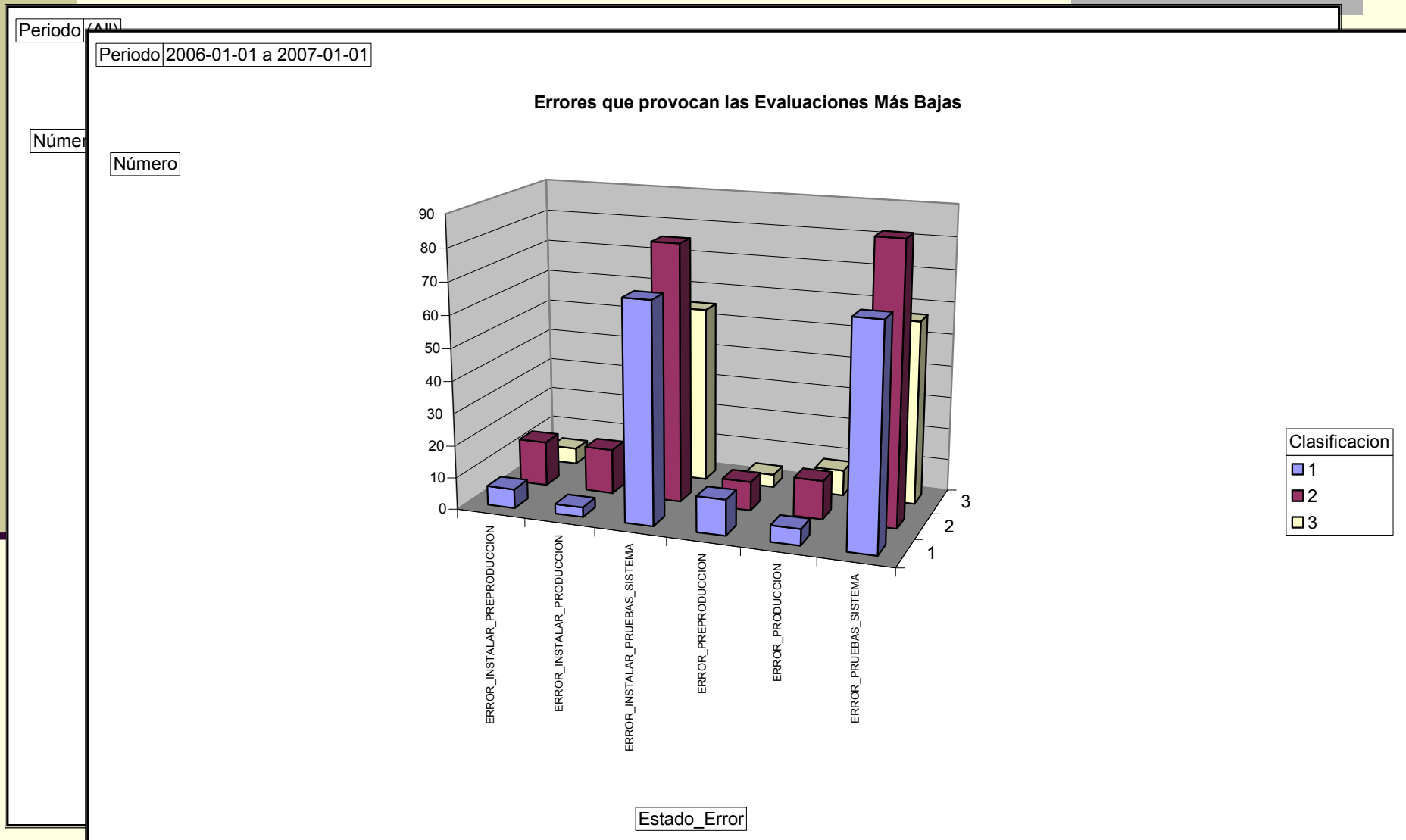
Periodo	TIPO		
	ALTA	BAJA	MEDIA
1998-01-01 a 1999-01-01	28,95%	18,42%	52,63%
1999-01-01 a 2000-01-01	28,95%	18,42%	52,63%
2000-01-01 a 2001-01-01	28,95%	18,42%	52,63%
2001-01-01 a 2002-01-01	28,95%	18,42%	52,63%
2002-01-01 a 2003-01-01	26,32%	36,84%	36,84%
2003-01-01 a 2004-01-01	26,32%	36,84%	36,84%
2004-01-01 a 2005-01-01	26,32%	36,84%	36,84%
2005-01-01 a 2006-01-01	26,32%	36,84%	36,84%
2006-01-01 a 2007-01-01	0,00%	39,47%	60,53%
2007-01-01 a 2008-01-01	28,95%	21,05%	50,00%
Total	25,00%	28,16%	46,84%

GRÁFICO DE EVALUACIÓN DE USUARIOS



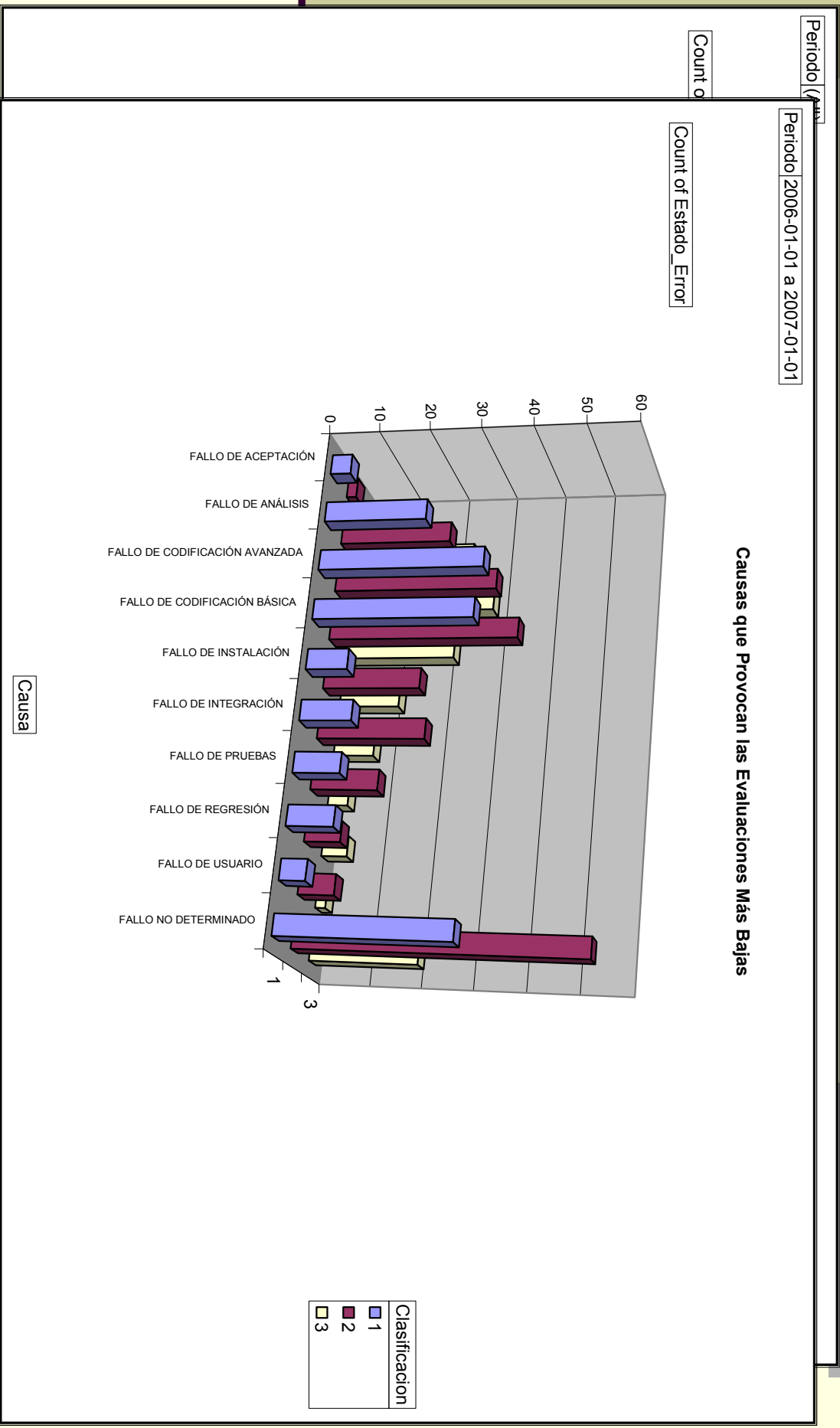
ANÁLISIS DE VIABILIDAD

Análisis de Resultados. Ciclo Aleatorio



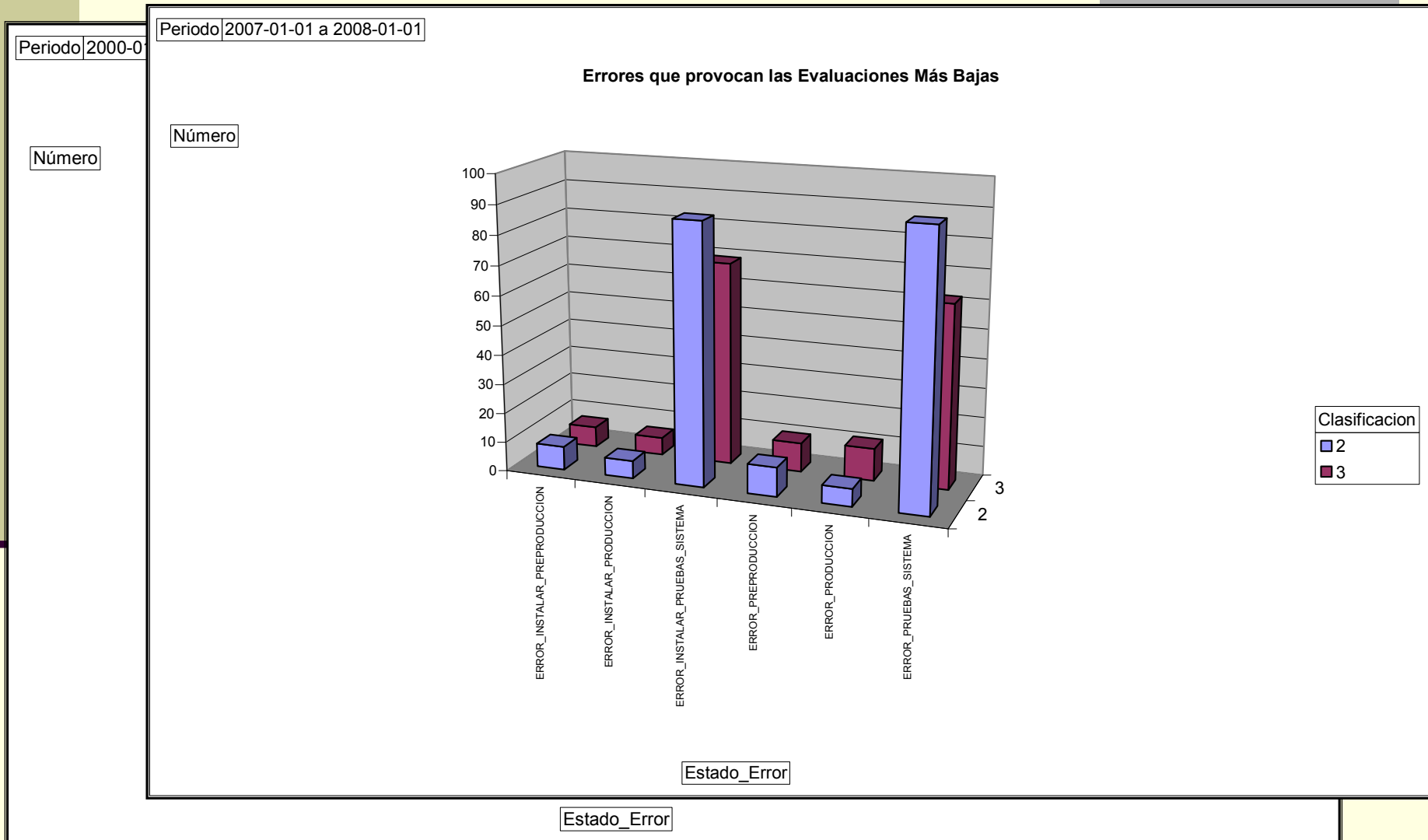
ANÁLISIS DE VIABILIDAD

Análisis de Resultados. Ciclo Aleatorio



ANÁLISIS DE VIABILIDAD

Análisis de Resultados. Ciclo Similar a Otro



ANÁLISIS DE VIABILIDAD

Análisis de Resultados. Ciclo Similar a Otro

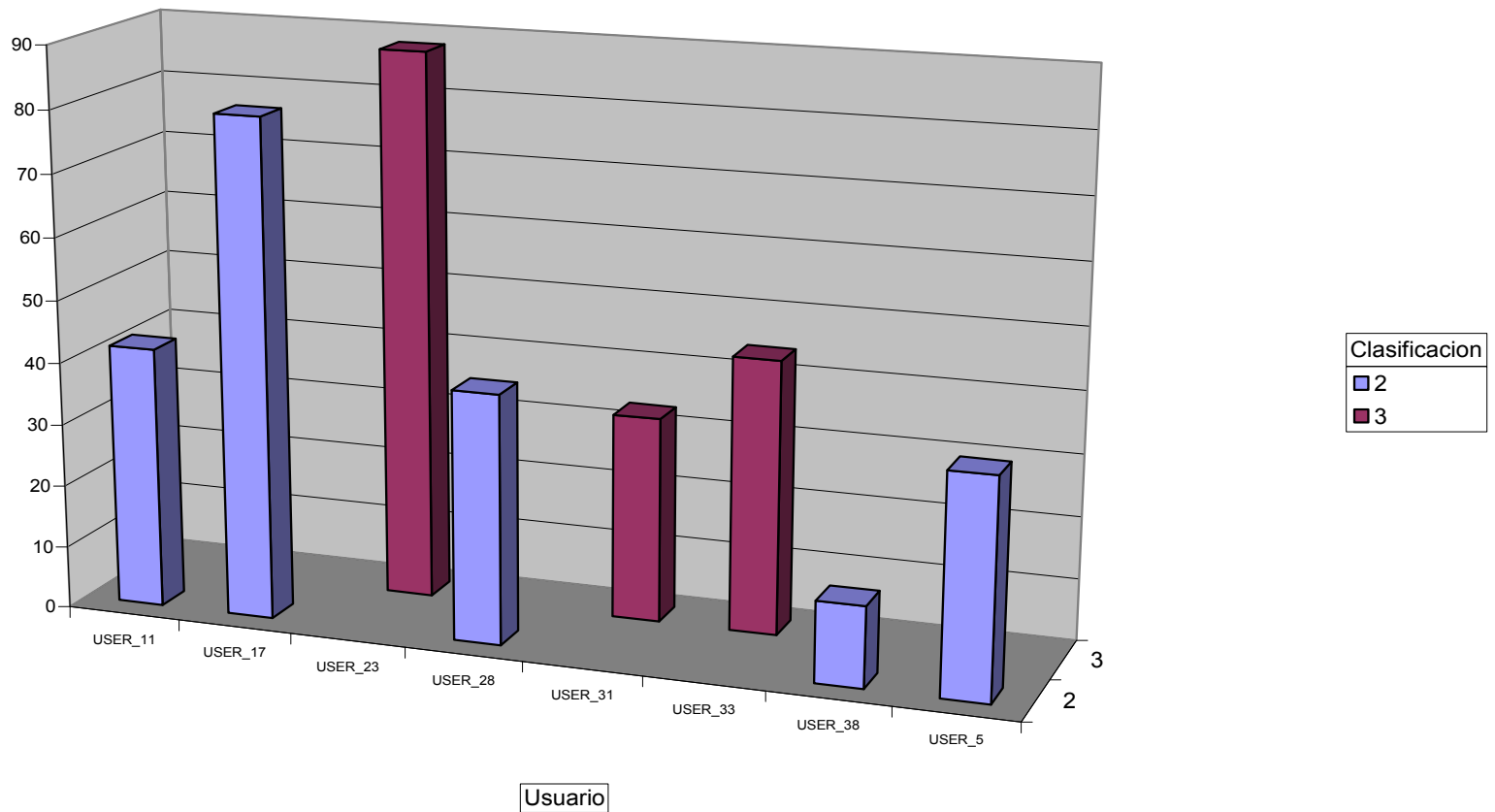
Periodo 2000-01-01 a 2001-01-01

Periodo 2007-01-01 a 2008-01-01

Número

Número

Usuarios con las Evaluaciones Más Bajas



CONCLUSIONES

- ⇒ Estado de la Cuestión
- ⇒ Planteamiento del Problema
- ⇒ Solución Propuesta
- ⇒ Análisis de Viabilidad

⇒ Conclusiones

- ⇒ Conclusiones de la Investigación Realizada
- ⇒ Líneas Futuras de Investigación. Mejoras del Sistema

CONCLUSIONES

Conclusiones de la Investigación Realizada

- ⇒ Solución que *Cumple las Hipótesis de Trabajo*
- ⇒ Solución de *Utilidad* para Toma de Decisiones y Detección de Tendencias, así como para anticipación y mitigación de Errores.
- ⇒ *Generalizable* y *Adaptable*.
- ⇒ Sistema *Integrado* Cliente/Servidor
 - ⇒ Gestión de la Configuración del Software (Control de Software, Control del Ciclo de Vida)
 - ⇒ Aprendizaje Automático (Redes Neuronales, Minería de Datos)

CONCLUSIONES

Líneas Futuras de Investigación. Mejoras del Sistema

⇒ Líneas Futuras de Investigación

- ⇒ *Descubrimiento del conocimiento en Razones “Check In” Implementado*
- ⇒ *Generalización Redes Neuronales*
- ⇒ *Evolución dinámica de Topologías de Redes Neuronales*

⇒ Mejoras del Sistema

- ⇒ *Generalización de Cliente*
- ⇒ *Administración de Datos*
- ⇒ *Extensión de la Seguridad*
- ⇒ *Otras herramientas de GCS*
- ⇒ *Gestión Desatendida de Trabajos*

RUEGOS Y PREGUNTAS

MUCHAS GRACIAS

