

**Instituto Tecnológico de Costa Rica**

**Escuela de Ingeniería en Electrónica**

**Componentes Intel de Costa Rica  
Sistema de despliegue en red de los indicadores de desempeño del  
robot Delta Summit Handler**

**Informe de Proyecto de Graduación para optar por el Bachillerato  
en Ingeniería Electrónica**

**David Rodríguez Chaverri**

**Belén, Enero de 2002**

## Dedicatoria

A mis padres

A ella por esa devoción con que me cuida y que la hace rezar un rosario diario por mi endeble salud. A él por ser ese ejemplo de honestidad, lealtad y deseos de aprender.

## Agradecimiento

Al Ing. Alexander García S, quien dedicó varios domingos a construir conmigo la página web. Al experto en programación Ferdinando Obando, pieza fundamental del proyecto, diseñador del programa de refrescamiento de la base de datos (entre otros aportes) y cuyos acertados comentarios durante todo el camino condujeron a una solución más eficiente.

A mi supervisor Msc. Thomas Kvanvig, quien supo combinar exigencia con trato amable y en general a todo el personal técnico del departamento, principalmente a Ing. Roy Marín, MTE Carlos Salazar y FSE Gustavo Marín, partícipes del proyecto en varias de sus etapas.

Todos ellos demostraron gran aptitud para el trabajo en equipo y dieron su ayuda en forma desinteresada. Gracias de nuevo.

Finalmente, a los profesores Reinaldo Jiménez Santamaría y Miguel Arias Vílchez, quienes forjaron en mí el gusto por la matemática y a quienes les debo mucho más que este simple agradecimiento.

# Resumen

En Componentes Intel de Costa Rica se diseñó una herramienta para el control estadístico de la utilización del robot Delta Summit Handler, mismo que se emplea en prueba de las procesador y con un costo de varios cientos de miles de dólares, cuyo uso debe ser optimizado.

El proyecto aprovecha una base de datos almacenada en cada handler, la cual registra cada error, para desplegar la información en un sitio web mediante la conexión a red de cada handler. Procesar esa información en la condiciones actuales es engorroso y tomaría más de de 9 horas de trabajo para una persona. Por esta razón no existen indicadores confiables del uso del equipo.

Habilitar la conexión de red presenta dos riesgos: que los robots sean infectados con un virus y un decrecimiento en el rendimiento de la máquina. Por ello se instaló software antivirus en cada máquina, se limitó el uso de la red y se diseñaron experimentos para la medición de las unidades por hora, garantizando que no existe cambio debido al nuevo software o a la conexión. Por otro lado, la aplicación web fue construida usando la tecnología de Microsoft ActiveX dll así como una base de datos central que recopila los datos de todos los handlers.

La versión final del sitio web (disponible sólo en la intranet) incluye 3 gráficos y 7 indicadores para cada handler relacionados con los errores más frecuentes, los de más larga duración, tiempo perdido a causa de estas interrupciones así como valores medios entre fallas y de reparación de las mismas.

Palabras clave: handler, sitio web, indicadores, base de datos, interrupciones.

# Abstract

The Delta Summit Handler, a robot that costs some hundreds of thousand dollars, is used at Componentes Intel of Costa Rica for testing the microprocessors. For improving the machine utilization of the handler a statistic tool was designed and this tool is the scope of this document.

The project takes advantage of a database available in every handler to record every error. This data is displaying in a web site, providing an intranet connection plus a reporting tool. The same process without software automation is difficult: is enough for keeping busy 1 employee for more than 9 hours per day. For this reason at this moment there are no reliable indicators of the machine utilization.

Enabling the connection of the Handlers has two risks: the exposition to intranet viruses and decreasing the tool performance. We addressed both installing antivirus software, reducing the intranet/internet access to the minimal possible and designing experiments to measure the run rate with and without the antivirus shield. The reporting tool was implemented using a web application based in ActiveX technology that generates all graphs and indicators plus an Ms-Access database remote collecting tool.

The last version of the web site displays 7 indicators and 3 graphs for every handler, all of them related with the most frequent interrupts, the interrupts that take more time, mean times between failures and to repair failures. Because this is confidential information, the site is only accessible for Intel Intranet users.

Keywords: handler, web site, indicators, database, interrupts, machine utilization.

# ÍNDICE GENERAL

1. INTRODUCCIÓN .....	7
1.1 Descripción de la empresa .....	7
1.2 Definición del problema y su importancia .....	8
1.3 Objetivos .....	9
1.3.1 Objetivo General .....	9
1.3.2 Objetivos específicos: .....	10
2. ANTECEDENTES .....	11
2.1 Estudio del problema a resolver .....	11
2.1.1 ¿Cómo extraer los datos? .....	11
2.1.2 ¿Cómo realizar la conexión a red? .....	12
2.2 Requerimientos de la empresa .....	14
2.3 Solución propuesta .....	15
Entrenamiento: .....	15
Conexión de los Handlers: .....	16
Sistema para capturar y desplegar los datos: .....	16
3. Procedimiento metodológico .....	18
4. Descripción del hardware utilizado .....	22
4.1 Summit Handler .....	22
4.2 Componentes de red .....	23
Inventario en piso de producción: .....	23
Configuración del servidor: .....	23
5. Descripción del software utilizado .....	25
MS Access: .....	25
MS Visual Studio: .....	26
MSDN: .....	26
Visual InterDev: .....	27
MS Front Page: .....	27
6. Análisis de resultados .....	28
6.1 Descripción del diseño .....	28
6.1.1 Elección del tipo de subred a utilizar: .....	28
6.1.2 Descripción del software desarrollado: .....	32
6.3 <i>Características del proceso de solución:</i> .....	43
6.2 Alcances y limitaciones .....	46
6.2.2 Limitaciones: .....	47
7. Conclusiones .....	49
7.2 Recomendaciones .....	50
Bibliografía .....	51

Apéndices .....	52
A1 Glosario .....	53
A2 Guía de instalación del Software antivirus para el Summit Handler.....	56

## ÍNDICE DE TABLAS Y FIGURAS

Figura 1 Diagrama de bloques de la solución.....	15
Tabla 1 Distribución de tareas en el proyecto.....	21
Figura 2 Delta Summit Handler (fotografía tomada de la página web del fabricante) .....	22
Figura 3 Subred cerrada para conexión de los handlers. ....	29
Figura 4 Conexión abierta usando software adicional de protección antivirus.....	30
Figura 5 Tabla Tools de la base de datos SummitDB. ....	33
Figura 6 Plantilla del gráfico de las 10 interrupciones de más larga duración. ....	36
Figura 7 Página de entrada del sitio web. ....	41
Figura 8 Despliegue de indicadores en la página web.....	42

# 1. INTRODUCCIÓN

## 1.1 Descripción de la empresa

El proyecto se desarrolló en Componentes Intel de Costa Rica, que pertenece a la transnacional Intel Corporation. La empresa normalmente emplea a más de 85 000 personas alrededor del mundo. Fue fundada el 18 de julio de 1968.

La actividad principal de Intel es la fabricación de microprocesadores. Actualmente, los productos más conocidos en el mercado son los procesadores Celeron, Pentium III y IV así como procesadores para servidores. En Costa Rica, una primer planta inició operaciones en el inicio de 1998. Una segunda planta comenzó a operar en 1999. La presencia de Intel ha tenido un fuerte impacto en la economía costarricense, aumentando significativamente las exportaciones. La planta que se encuentra en Costa Rica es de Assembly/Test, es decir, de ensamblaje y prueba.

El proyecto pertenece al departamento de Test Process Engineering, que pertenece a su vez a Assembly Test Engineering.

La sección Assembly Test Engineering tiene 95 empleados, entre los cuales se cuentan profesionales en electrónica, mecánica y computación. Por razones de confidencialidad se omite la cantidad de ingenieros en electrónica que laboran en el departamento.

La función principal del departamento es dar soporte al sistema de test en el piso de producción. Para garantizar la calidad de sus productos, en Intel se emplea un sofisticado mecanismo de prueba de los chips. Manejar ese mecanismo, así como darle mantenimiento y buscar su optimización, es la tarea primordial de la sección en que se desarrolló el proyecto.

## **1.2 Definición del problema y su importancia**

La producción de microprocesadores es un proceso delicado y cuidadoso en cada una de sus etapas. Componentes Intel de Costa Rica debe garantizar a sus clientes que sus productos han sido probados y se ha verificado su correcto funcionamiento.

Para lograr los altos estándares de calidad, Intel somete a cada chip a un proceso de test, para ello cuenta con equipo de alta tecnología y de un elevado valor económico. Los procesadores Pentium, como parte de ese proceso, pasan por prueba funcionales y estructurales que determinan si el chip funciona y la velocidad del procesador. Para cada prueba existe un tester especial. Existe un robot o servomecanismo que sirve como interfase entre el chip y el tester, que permite colocar un grupo de chips y ser probados en forma automática. Este aparato tiene un costo bastante elevado pero significativamente menor al de cada tester.

Cada tester y cada robot (llamado Summit Handler, en Costa Rica existen 43) tienen fallas con cierta frecuencia que obligan a detener el proceso para hacer reparaciones. También es necesario realizar mantenimiento preventivo. Dado el alto costo económico del equipo, cada minuto que se pierde es bastante significativo. Por ello se diseñó una página web con información de los testers que incluye si está detenido o en funcionamiento, el tiempo que tiene en el estado correspondiente y el tipo de falla producida, si es el caso. La página sirve para hacer un análisis estadístico de cada tester y puede orientar el mantenimiento.

El problema es que no se cuenta con un sistema de monitoreo para los handlers, y estos tienen un tiempo entre fallas menor al de los testers. Por ello, la información que se recibe de la página web no es exacta, ya que un tester puede estar detenido a causa de una falla en el handler. Los handlers no cuentan con conexión a la red, y si se quiere acceder a la información del aparato es necesario exportar la información a un disquete.

Es indispensable reducir el tiempo en que el handler se encuentra detenido para acelerar el proceso de test de los componentes.

## **1.3 Objetivos**

### **1.3.1 Objetivo General**

Conectar cada Delta Summit Handler a la intranet de Intel y diseñar una página web que muestre periódicamente los indicadores de desempeño de cada robot.

### 1.3.2 Objetivos específicos:

1. Recibir entrenamiento en el manejo del sistema Summit Handler-Tester.
2. Determinar los componentes de red necesarios para realizar la conexión física de los Handlers a la intranet.
3. Realizar la conexión de un handler a la red y realizar pruebas utilizando una PC adicional.
4. Documentar la conexión a red de los handlers para ser usada en otros sitios de Intel.
5. Determinar los datos que deben ser extraídos del Delta Summit Handler.
6. Elegir un lenguaje de programación de alto nivel para extraer la información.
7. Investigar posibles métodos para generar las bases de datos necesarias, en cada handler, para generar la página web. Es posible que se requiera contactar al fabricante del software.
8. Implementar un programa que genere la información de la página web, para ser corrido en un servidor.
9. Probar dicho programa utilizando una PC y uno de los handlers.
10. Diseñar la página web.
11. Probar y depurar el sistema.
12. Garantizar la seguridad del sistema mediante la instalación de software antivirus en cada PC.
13. Realizar la conexión del resto de los Handlers incluyendo la configuración de cada PC.
14. Documentar el sistema total para su evaluación en el I.T.C.R y en otros sitios de Intel.

## 2. ANTECEDENTES

### 2.1 Estudio del problema a resolver

La idea básica del proyecto es proveer una herramienta de despliegue de los indicadores de desempeño del robot Summit Handler. Esto quiere decir contar con un sistema capaz de obtener valores promedios de importancia como el tiempo medio entre fallas, tiempo promedio de reparación de esas fallas, errores más comunes, cantidad de tiempo perdido debido a fallas, etc. Es deseable que parte de los datos se presenten en un formato gráfico de fácil manejo.

#### 2.1.1 ¿Cómo extraer los datos?

El Summit Handler se comunica con el tester utilizando un puerto GPIB que es gobernado por una estación de trabajo SUN. Existe una herramienta que emplea ese puerto para extraer cierta información del Summit. Una de las tareas consistía precisamente en determinar si una nueva conexión era necesaria y hasta que punto el proyecto iba a suministrar información realmente nueva. Debe decirse aquí que el manejo de los datos vía GPIB representa un riesgo importante: afectar el rendimiento del tester. De hecho, existe una experiencia negativa dentro de la empresa con una página de reporte de indicadores del tester que causaba un decremento en el rendimiento de éste.

La propuesta de la empresa fue usar una conexión de red independiente del funcionamiento del tester. El mismo robot tiene una opción de exportar los datos a un archivo Excel, con información de los indicadores más importantes, pero obtener los datos de todos los handlers se convierte en una tarea bastante tediosa.

De hecho, calculando 10 minutos por cada máquina y 1 hora 30 minutos para recopilar toda la información y desplegarla en gráficos, este proceso tomaría más de un día de trabajo para un empleado normal. Con los handlers en red se la descarga puede hacerse en forma automática.

El otro problema consistía en utilizar los comandos del software de manejo del Summit (interfaz entre el operario y el robot) de forma remota. Específicamente, como emplear el comando export desde una PC distinta a la PC interna del Summit. Indudablemente era necesario consultar al fabricante del Summit. No obstante, se encontró un camino más corto y sencillo: cada vez que se usaba la orden export los archivos se guardaban en la misma carpeta denominada export files. Investigando un poco se encontró que esta orden empleaba una base de datos local, por lo que la clave era usar estas bases de datos. La existencia de estas bases de datos no era conocida por el estudiante al inicio del proyecto, aunque sí era usada en otros sitios de Intel.

Entonces, el problema se reduce a procesar en forma remota los datos de dichas bases.

### 2.1.2 ¿Cómo realizar la conexión a red?

El Handler cuenta con una PC interna con un puerto de red disponible, pues posee una tarjeta dual Ethernet. A primera vista, el asunto es simplemente habilitar un puerto de red cercano a cada handler, sin embargo existen varios aspectos a considerar.

- La seguridad de la conexión. La información del Summit Handler es confidencial. El sitio web no debe ser accesible desde cualquier PC sino sólo a través de la intranet.

- El riesgo de exposición a virus: Actualmente el principal medio de propagación de virus informáticos es la red de computadoras, con muchos casos recientes de virus que utilizan el correo electrónico o que se auto envían desde servidores. Se debe garantizar que el sistema cumple con los requerimientos de Intel en protección antivirus.
- El Summit Handler utiliza una red interna para la comunicación entre diversas partes del robot y el computador central. Esta red se basa en el protocolo TCP/IP. La nueva conexión debe funcionar en forma separada de esta red.
- En total se requieren 40 conexiones, cuyo número puede aumentar con el tiempo. El número de puertos de red es limitado y esto puede requerir crear nuevos segmentos de red.
- El tráfico de red se incrementa.
- El tiempo de configuración de cada máquina (cambio de dirección IP por ejemplo) debe mantenerse tan corto como sea posible para no impactar la productividad del área de test.

Debe tomarse en cuenta que el Summit Handler se encuentra en producción casi todo el tiempo, por lo que no es sencillo realizar experimentos. Esto dificulta algunos pasos, como la instalación de software antivirus. Además es necesario seguir el estándar de Intel, que tiene reglas particulares para conectar una PC a la intranet. Efectuar experimentos en el Summit exige un entrenamiento previo sobre el funcionamiento del mismo.

## 2.2 Requerimientos de la empresa.

La empresa definió el proyecto con base en 4 objetivos básicos:

- Conectar 1 Handler.
- Conectar el resto de Handlers
- Diseñar una página web que despliegue los principales indicadores del Summit Handler.
- Proliferar el sistema a otros sitios de Intel.

Además de esto, la empresa mostró interés en que el practicante tomara cursos en diversas áreas, no todos relacionados directamente con el proyecto aunque sí útiles. El entrenamiento en el Summit fue parte de este plan, así como cursos de manejo de proyectos y solución estructurada de problemas entre otros.

Prácticamente cada avance del proyecto fue compartido con otros sitios de Intel a través de diversas presentaciones, esto con los siguientes propósitos

1. Informar del estado del proyecto.
2. Recibir realimentación de posibles usuarios
3. Desarrollar, en la medida de lo posible, tareas en paralelo con otros sitios

De hecho algunas etapas se sometieron a la evaluación de un foro con representantes de cada planta de Intel. Estas presentaciones no se incluyen dentro de este reporte por tratarse de información confidencial.

El proyecto fue supervisado por el Msc. Ing. Thomas Kvanvig, quién definió junto con el estudiante un detallado calendario del proyecto, el cual se evaluaba semanalmente. Posteriormente se estableció una reunión (efectuado cada lunes) con cada usuario, posible beneficiario o colaborador del proyecto, a fin de coordinar todo el trabajo y dar una idea clara de lo que se pretendía hacer.

## 2.3 Solución propuesta.

La solución al problema es diseñar una página web en la intranet para los handlers, similar a la que ya existe para los testers. En esta página aparecerían los indicadores de desempeño principales (tiempo medio entre asistencias, jams y fallas, tiempo medio de reparación de los mismos) así como distintos gráficos con los errores más significativos (los más frecuentes, los de más larga duración, etc.) en diferentes intervalos (diario, semanal, mensual, etc.).

A continuación se muestra un diagrama de bloques de la solución:

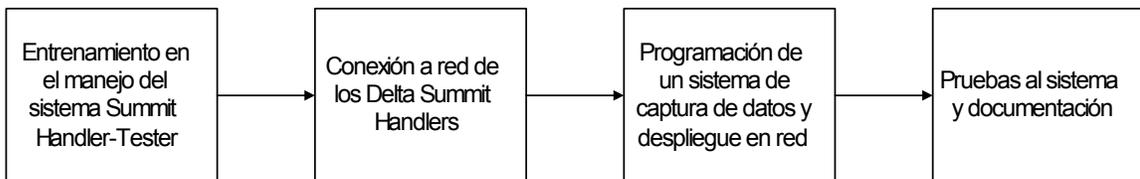


Figura 1 Diagrama de bloques de la solución.

### Entrenamiento:

La primer etapa es recibir entrenamiento en el manejo del robot y de los testers, para obtener conocimientos básicos sobre la operación del sistema, el tipo de datos que es necesario colocar en la red, la forma en que se extraen esos

datos en este momento, etc. Esta etapa es de suma importancia, puesto que se debe garantizar que la nueva aplicación no afecta el rendimiento del robot y ello supone la realización de experimentos.

El entrenamiento consiste en la certificación como operario de nivel 1 del Summit Handler. Esto permite utilizar el Handler en su modo de funcionamiento normal.

#### Conexión de los Handlers:

La segunda fase de la solución es conectar los handlers a la red de computadoras internas. Cada handler cuenta con una PC interna, desde la cual se corre el software que controla al mismo. Se debe evaluar el sistema con que se cuenta, a fin de aclarar cuestiones como los recursos disponibles, si existen puertos de red libres cerca del handler, que componentes de red se deben agregar (hubs, un servidor, etc.), garantizar la seguridad de la conexión y configurar finalmente cada máquina para el acceso a la red. El punto más relevante de esta etapa es la confiabilidad del sistema, es preciso proteger a los Handlers de posibles virus en la red, ya sea mediante instalación de software adicional o bien manteniendo a los Handlers en un ambiente seguro (deshabilitando el acceso a internet, manteniendo contacto sólo con la red local). Este aspecto se retoma en posteriores secciones del reporte.

#### Sistema para capturar y desplegar los datos:

Aquí se propone tomar un lenguaje de programación de alto nivel (Visual Basic) para la recopilación y posterior procesamiento de la información de las bases de datos.

El sistema consta de dos partes: el programa de carga de la información en una base de datos central (Summit Loader) y el sistema de procesamiento de la misma propiamente. Se puede encontrar una descripción amplia de este software

en la sección de análisis de resultados.

Para finalizar se debe diseñar la página web antes referida y someter el sistema a pruebas.

La solución fue propuesta por la empresa. Muchos aspectos quedan a la elección de practicante, como escoger los componentes de red y la forma en que la red va a administrar la información o el lenguaje de programación a utilizar. La idea principal para resolver el problema (conectar los robots a la red) fue sugerida antes de iniciar el proyecto.

### **3. Procedimiento metodológico.**

1. Entrenamiento (2 semanas):

- 1.1 Contactar al encargado de entrenamiento de primer nivel y definir un horario de entrenamiento (1 día).
- 1.2 Recibir el entrenamiento de primer nivel de manejo del Summit Handler y Tester (10 días).

2. Determinar los componentes de red necesarios (2 semanas):

- 2.1 Concertar una reunión con el administrador de la intranet (1 día).
- 2.2 Realizar visitas al piso de producción a fin de determinar puertos de red libres cerca de cada handler (2 días).
- 2.3 Elegir los componentes de red necesarios (hubs, servidor, etc) (10 días)

3. Conectar un handler a la red (2 semanas):

- 3.1 Realizar la conexión física y lógica (configuración) (1 semana).
- 3.2 Establecer comunicación con una PC de la intranet (1 día)

4. Documentar la conexión a la red (1/2 semana):

- 4.1 Elaborar un informe de los componentes empleados y la forma en que se configuró la conexión del handler a la red.

5. Determinar los datos que deben ser extraídos del Delta Summit Handler (1 semana):

- 5.1 Consultar al ingeniero de proceso sobre los datos que son exportados frecuentemente desde el handler. (1 día).

5.2 Consultar la página web existente para la información de los testers (1 día)

5.3 Estudiar la forma en que se generan esos datos en el software del handler (3 días).

6. Elección del lenguaje de programación a emplear (2 semanas):

6.1 Determinar un lenguaje de programación que facilite la interacción con el software del Handler (2 días)

6.2 Estudiar las características principales del lenguaje escogido (13 días)

7. Implementación de la generación de las bases de datos desde el Handler (2 semanas):

7.1 Consultar al fabricante del software sobre la utilización del programa mediante líneas de comando y el uso de la información de las bases de datos (2 días).

7.3 Verificación y análisis de las bases de datos existentes en el handler (12 días).

8. Implementación del programa a correr en el servidor (6 semanas):

8.1 Entrenamiento en el lenguaje escogido (1 semana)

8.2 Programación (5 semanas)

9. Prueba del programa usando una PC y un handler (1 semana):

9.1 Instalar el programa en el handler que se ha conectado a red y en una PC que se utilice como servidor (1 día).

9.2 Realizar pruebas al sistema (4 días).

10. Diseño de la página web (1 semana):

10.1 Diseñar una página web que presente el estado de los handlers y refresque la información en intervalos cortos de tiempo (1 semana)

11. Pruebas al sistema (1 semana):

11.1 Verificar el funcionamiento del sistema total (2 días)

11.2 Depurar el sistema. (3 días)

12. Instalación de software antivirus en cada Handler (3 semanas):

12.1 Planeación y realización de experimentos en una máquina para asegurar que el rendimiento no se va a afectar (1 semana)

12.2 Instalación en cada máquina (2 semanas)

13. Conexión del resto de los handlers (1 semana):

13.1 Efectuar la conexión física y lógica de los demás handlers siguiendo la documentación de la conexión del primer handler.

13.2 Instalar el software de generación de la información para la página web.

13. Documentación del sistema (2 semanas):

14.1 Elaboración del informe de práctica y presentación final.

14.2 Entrega de la documentación a la Biblioteca del I.T.C.R.

14.3 Elaboración de un informe a Componentes Intel de Costa Rica para su evaluación.

14.4 Presentación del proyecto en el I.T.C.R.

Para las tareas de programación y prueba, así como en el diseño de la página web, se cuenta con la asesoría de Ferdinando Obando, programador de la empresa. En la elección de los componentes de red y asignación de direcciones IP, el Ing. José Fung está asignado. El Ing. Robert Walker es el Ingeniero asesor en la empresa. En la siguiente tabla se muestra la distribución de responsabilidades del proyecto:

Encargado	Función
David Rodríguez Ch. (practicante)	Coordinar el proyecto, realizar la programación y las pruebas al sistema.
Ferdinando Obando	Brindar asesoría en el lenguaje utilizado, diseño del programa de recolección de datos y colaboración en la etapa de pruebas.
Ing. Alexander García	Diseño de la página web
Ing. José Fung	Asignar las direcciones IP, garantizar el espacio de red para la conexión del robot.
Ing. Thomas Kvanvig	Supervisar el proyecto.
Ing. Roy Marín.	Colaboración en la etapa de calificación del proyecto y proliferación.
Ing. Robert Walker	Brindar asesoría en la realización del proyecto en general.

Tabla 1 Distribución de tareas en el proyecto

## 4. Descripción del hardware utilizado

En cuanto al hardware sólo puede hacerse referencia al Delta Summit Handler, un robot, y a los componentes de red necesarios para conectarlo a la red.

### 4.1 Summit Handler



Figura 2 Delta Summit Handler (fotografía tomada de la página web del fabricante)

El Delta Summit Handler es un sistema electromecánico que se encarga de tomar cada procesador y prepararlo para su proceso de prueba. El tester necesita ciertas condiciones en el chip, como una temperatura específica, las cuales se obtienen dentro del Handler. El Handler somete a cada chip a la prueba durante

varios segundos (el tiempo depende del tipo de procesador y de la prueba), la cual se realiza utilizando una tarjeta de interfaz tester-chip denominada TIU. Todas las condiciones de prueba se especifican en un archivo de tipo texto.

El costo del sistema total (tester, handler y estaciones de trabajo) es bastante elevado, superando los \$2 millones. El handler, por su parte, tiene un precio de varios cientos de miles de dólares. Aumentar su eficiencia, por lo tanto, es de vital importancia.

## **4.2 Componentes de red**

Este hardware no fue manejado directamente por el estudiante pues existe un departamento dentro de la empresa que se encarga de la administración de la intranet local, aunque sí hubo participación del practicante en algunas tareas, como se procede a explicar.

### Inventario en piso de producción:

Dado que cada handler requiere un puerto de datos libres se debió identificar cuál era el TO (caja de conectadores) más cercano en cada caso. Después de una visita a piso se envió el inventario, que incluye los componentes conectados y los puertos libres, escogiéndose el puerto libre más cómodo, así como la cantidad de cable necesaria para cada caso.

### Configuración del servidor:

Fue necesario destinar una computadora para mantener la página web siempre disponible, además de almacenar la base de datos que recopila la información de todos los Handlers del sitio. Como la aplicación a diseñar no es excesivamente grande y no se tienen grandes

requerimientos en cuanto a la velocidad del sistema (el usuario promedio accedería la página una o dos veces por semana), se puede utilizar una PC normal (es decir, no optimizada para su uso como servidor, que implica diseño orientado a sistemas multitarea y multiusuario). Para reducir costos se empleó una PC HP Kayak Pentium II de 500MHz que estaba disponible a la que se le agregó un disco duro de 30 GB (en lugar del de 6GB con que contaba inicialmente), 64MB de RAM (194MB RAM en total), una tarjeta de red adicional (requerimiento para su inserción en el Computer Room, lugar especialmente diseñado para mantenimiento de servidores) y una tarjeta controladora SCSI para el manejo de una cinta magnética de respaldo de la información.

## 5. Descripción del software utilizado.

La mayor parte del software utilizado es relativo a la programación. Una característica en común de todos los programas es que se trata de software fabricado por Microsoft. A continuación una breve reseña:

### **MS Access:**

Se empleó para el manejo de base de datos. El Access es un programa de muy fácil utilización que permite manipular bases de datos pequeñas (en teoría permite hasta 3GB de información). La facilidad de aprendizaje implica un sacrificio en la eficiencia: el Access no trabaja muy bien en ambientes multiusuario si el número de usuarios es grande, además es mucho más lento que otros sistemas de almacenamiento de la información como Sql Server.

La elección de Access para el manejo de las bases de datos se dio basado en las siguientes consideraciones:

- El software del Summit Handler almacena los datos usando el formato de Access.
- El programa Access permite un manejo mucho más simple de la información, lo cual es deseable dada la poca experiencia del practicante en bases de datos.
- La aplicación es pequeña y se supone que el número de usuarios es pequeño. La posibilidad de tener muchos usuarios al mismo tiempo es en realidad remota.
- La base de datos no es muy grande. Si en algún momento se acercara al límite de 3GHz puede crearse un respaldo de la información y eliminar la información de cierta antigüedad.

De todas formas, sería deseable migrar a Sql Server la base de datos del servidor en el futuro, sin afectar el formato de las bases en los Handlers.

## **MS Visual Studio:**

El Visual Studio es un conjunto de herramientas para la programación en Windows, que incluye Visual Basic, C++, J++, Interdev y la biblioteca de ayuda MSDN. De ellos fueron utilizados Visual Basic, Visual Interdev y el MSDN.

### Visual Basic:

Lenguaje de programación orientado a objetos, con un conjunto muy poderoso de herramientas y un entorno de programación muy amigable. La desventaja principal es la baja velocidad en tiempo de ejecución, en comparación con otros lenguajes; sin embargo la rapidez de respuesta del sistema no es primordial en este proyecto (es deseable más no indispensable). Se trataba de encontrar una herramienta de programación ventajosa para el principiante y Visual Basic además de cumplir este requisito brinda una gran facilidad de manejo de bases de datos (ADO, ActiveX data objects), potente conexión con otros programas de Microsoft (en particular, MS-Excel y Access) y una documentación muy completa.

### MSDN:

La biblioteca de ayuda de Microsoft contiene más de 1GB de información y un sinnúmero de enlaces a sitios web. Esta documentación se utilizó básicamente para la creación de los gráficos en Excel (bajo el tópico de Excel automation) y el manejo de la tecnología ADO.

Visual InterDev:

El InterDev es una herramienta para el desarrollo de aplicaciones web. Se utilizó para escribir el código html y asp necesario para desplegar en web la información de la base de datos. En este punto fue fundamental el aporte de uno de los ingenieros en computación de la compañía, Ing. Alexander García S.

MS Front Page:

El Front Page es parte del conjunto de programas incluidos en el MS-Office y facilita la publicación de páginas web. Se empleó para elegir la interfaz con el usuario de la página web, así como algunas pruebas pequeñas.

## 6. Análisis de resultados

### 6.1 Descripción del diseño

El diseño se debe dividir en sus principales componentes. De hecho, durante el transcurso del mismo se tomaron varias decisiones importantes, las cuales se detallan enseguida.

#### 6.1.1 Elección del tipo de subred a utilizar:

Una vez determinada la cantidad de recursos de red necesarios (espacio en hubs y en switches) para la conexión de los handlers se procedió a elegir la forma en que estos iban a ser conectados. En particular existían dos opciones:

- Configurar en cada máquina el estándar de Intel para garantizar que la conexión es segura (es decir, que la máquina tiene la protección mínima contra virus y que la información no se va a fugar).
- Utilizar una VLAN (virtual local area network, subred virtual) cerrada, donde sólo el servidor tendría acceso a cada máquina. De esta forma, sólo es necesario proteger al servidor con antivirus.

A continuación se muestran las ventajas de cada opción:

VLAN cerrada:

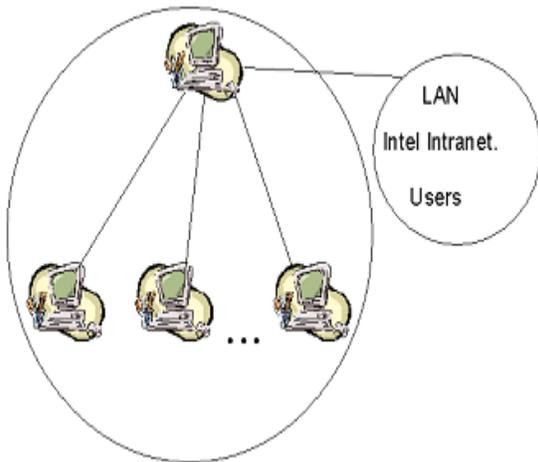


Figura 3 Subred cerrada para conexión de los handlers.

Una subred cerrada es una opción muy confiable y segura. En esta configuración el servidor funcionaría como un puente entre los usuarios de la página web y los Handlers. La figura ilustra esta situación de forma clara: el servidor toma los datos de cada robot y los almacena en el disco duro, cuando el usuario accede a la aplicación web en realidad está utilizando datos almacenados en el servidor. No hay contacto usuario-robot. En general es deseable que el acceso a cada robot esté restringido, puesto que un mal manejo (cambios en ciertos archivos de configuración) puede conducir a que la máquina se detenga, sin embargo la imposibilidad de acceso constituye una desventaja importante. La idea básica es permitir el acceso a el disco duro del robot mediante una clave.



revisar ciertos archivos de configuración. Esta configuración de red no requiere que los recursos se destinen en forma exclusiva, como el caso anterior. En este caso el servidor es una parte más del sistema y no funcionaría como un puente (aunque, evidentemente, es el computador con más funciones). En esta alternativa no se necesita una tarjeta de red adicional para el servidor. La mayoría de los usuarios sólo tendrían acceso al sitio web.

El mayor inconveniente es la instalación del software, pues esta tarea toma cerca de 40 minutos y requiere que la PC se reinicie varias veces (por lo tanto se debe detener el proceso de producción). No se necesita una red dedicada en forma exclusiva al servidor, lo que reduce los costos en forma significativa.

La instalación de este software implica un cambio en el Summit Handler, aunque se supone que ello no debe significar un decremento en el rendimiento de la herramienta. Los nuevos programas, cuya función es proteger al sistema de virus, podrían causar problemas a algunos archivos de configuración (debe decirse que el riesgo de que esto suceda es en realidad bajo). Por esta razón, se diseñaron algunos experimentos que comparaban el desempeño del robot antes y después de la instalación del antivirus.

Considerando ambas opciones se decidió instalar el software y usar una conexión abierta por ser una opción mucho más económica y flexible.

### 6.1.2 Descripción del software desarrollado:

El desarrollo del software se divide en 3 partes primordiales:

- Un programa de acceso a las bases de datos locales y almacenamiento en una base de datos central.
- Una librería con todas las funciones para el procesamiento de los datos (construcción de gráficos, obtención de valores promedio)
- La aplicación web, basada en la librería anterior.

Se ahondará en cada componente para obtener una visión más detallada.

#### **6.1.2.1 Programa de carga de la base de datos (Summit Loader):**

Este programa fue desarrollado por Ferdinando Obando, experto en programación de la empresa. El papel del estudiante fue definir claramente cuál era la función que debía tener la aplicación, así como colaboración en el diseño de la base de datos central (que es, en esencia, la base de datos local más algunos campos para el control del programa). En el transcurso del desarrollo del software se hicieron varias reuniones para resolver algunos conflictos de lógica.

Debe considerarse que este elemento del software está íntimamente ligado a las funciones que contiene la librería desarrollada por el practicante, por lo tanto era necesario coordinar múltiples aspectos del diseño con el programador.

El Summit Loader (nombre del programa) usa la dirección IP de cada handler para descargar información de su base de datos y colocarla en la base de datos que se encuentra en el servidor. Los datos de cada robot se encuentran

almacenados en una tabla de la base de datos, tools, que contiene los siguientes campos:

	toolid	description	version	location	sconnection	susername	spassword	dbpath	dbpassword	lastload
▶	R SH 001	IX		CR3	\\172.18.16.31\c\$	administrator	*****	\\summit\statisti	*****	201 6:44:50 PM
	R SH 401	ST		CR3	\\172.18.16.31\c\$	administrator	*****	\\summit\statisti	*****	201 6:44:50 PM
	R1 SH 001	IX		CR1	\\172.18.16.31\c\$	administrator	*****	\\summit\statisti	*****	201 6:44:50 PM
	R1 SH 401	ST		CR1	\\172.18.16.31\c\$	administrator	*****	\\summit\statisti	*****	201 6:44:50 PM
*										

Figura 5 Tabla Tools de la base de datos SummitDB.

**TOOLID:** Nombre del Summit Handler. Usado como identificador, constituye la relación entre esta tabla y el resto de la base de datos.

**DESCRIPTION:** Incluye el tipo de tester a que se conecta el robot. Existen 4 categorías: IX (tester funcional), ST (estructural), ST2(nueva generación del tester estructural, recién implementado) y OLF(prueba de fuse).

**VERSION:** Es un campo reservado para incluir la versión del software controlador del Summit. Una nueva versión puede representar cambios en el formato de la base de datos.

**LOCATION:** relacionada con el sitio de Intel al que pertenece el Handler y, dentro de cada sitio, con el edificio correspondiente. En Costa Rica existen 2 opciones: CR1 y CR3.

**SCONNECTION:** Dirección IP de cada handler

**SUSERNAME:** nombre de usuario de la PC. El Summit admite 4 tipos de usuarios.

**SPASSWORD:** clave del usuario anterior.

**DBPATH:** dirección, dentro del disco duro del robot, de la base de datos.

*DBPASSWORD*: clave de acceso a la base de datos.

*LASTLOAD*: fecha y hora del último acceso a la base de datos. Este dato se usa cada vez que se carga la información para considerar únicamente la información nueva.

Por lo tanto, el sistema realiza una consulta basado en la última actualización, utilizando la dirección IP para direccionar la máquina, el spassword en caso de que necesite abrir una sesión, luego el dbpath para acceder al archivo en cada handler y el dbpassword para abrir la base de datos.

#### **6.1.2.2 SHProject ActiveX dll:**

Este componente contiene todas las funciones, propiedades y métodos que se utilizan para construir la página web. Se utiliza dentro del diseño de la aplicación web, la idea de elaborar una librería dll es simplificar ese diseño y hacerlo más robusto.

Esta librería contiene las siguientes funciones:

*CONECTAR*: Se utiliza para abrir la base de datos local. Contiene todo el protocolo de acceso a la información almacenada en la base de datos.

*Sql din*: el nombre alude a sql dinámico. La consulta de información a una base de datos consiste en elegir cuáles campos, en determinada(s) tabla(s), se desean obtener y que cumplan una serie de condiciones. Por ejemplo, todas los errores producidos entre dos fechas para un handler específico.

El comando sql (structural query language) se ejecuta utilizando tecnología ADO (ActiveX Data Object). Básicamente, se establece una variable de tipo recordset (tipo especial para el manejo de bases de datos), la cual permite almacenar datos de la base local en forma temporal. Esta variable se carga

usando un comando sql.

El comando sql es un string (cadena de caracteres) que sigue una serie de reglas de sintaxis. No se permite utilizar variables dentro de la cadena, lo que obliga a construir dicha cadena de forma dinámica.

Lo anterior queda claro con un ejemplo. Si se desea obtener los registros que se encuentren entre dos fechas elegidas por el usuario, el sistema debe recibir esas fechas y construir con ellas el comando sql. Como no es posible usar las variables dentro del comando, la cadena se crea por partes, convirtiendo el valor de la variable a un tipo cadena.

Comando sql = estatutos & str(variables) &... & estatutos & str(variables)

Cada vez que se desea introducir una variable dentro de los estatutos propios de un comando sql se convierte el valor de esta variable a una cadena (con la función str) y este nuevo string se concatena (usando el operador &) al comando sql.

La función descrita (sqldin) concatena las dos fechas elegidas por el usuario en el comando sql.

*TOP5:* Selecciona la información de las 5 interrupciones más importantes del día. Se utilizó para la prueba del componente en su etapa de diseño, permitirá desplegar las 5 interrupciones más importantes del día en caso de considerarse importante.

*TOP10GRAPH:* El procedimiento top 10 graph se encarga de construir un gráfico en Excel con las 10 interrupciones de más duración según el Handler(s) escogido(s) y el intervalo deseado (fechas). Para el gráfico se utiliza una hoja de cálculo en

Excel que contiene una plantilla del mismo (véase figura 6.4).

Top 10 Interrupts (sort by time duration)

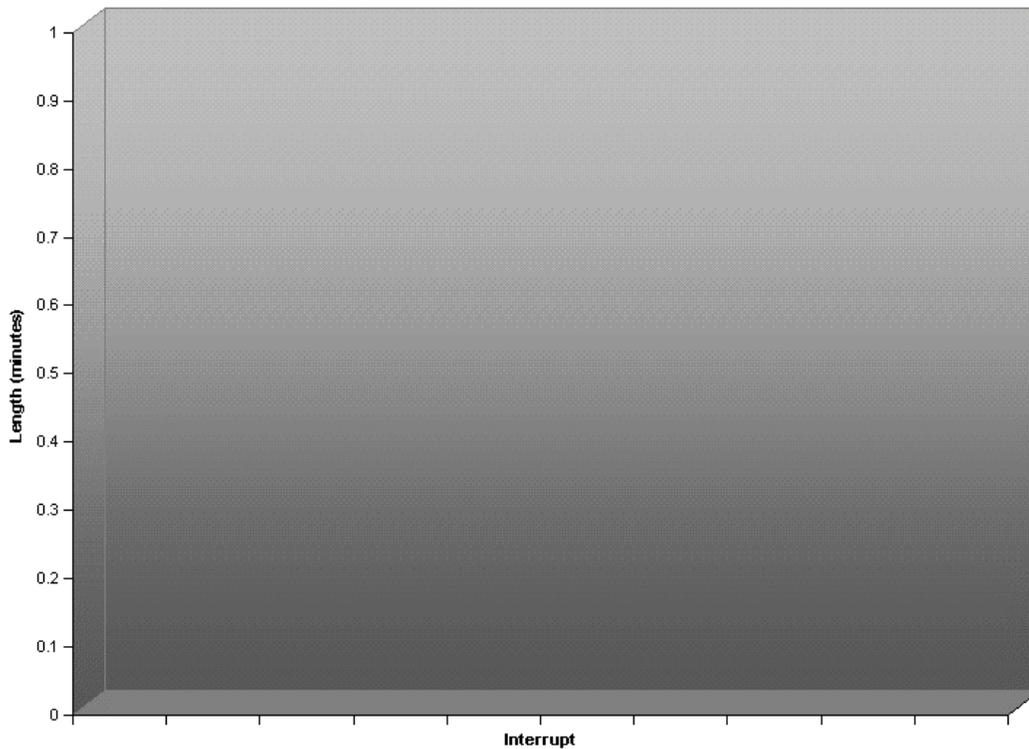


Figura 6 Plantilla del gráfico de las 10 interrupciones de más larga duración.

Como se aprecia en la figura esta plantilla contiene todas las características del gráfico a excepción de los datos. La idea es que el programa no tenga que crear el gráfico desde el inicio sino sólo agregarle los datos. Como el número de valores es fijo en este caso (10 valores para cada eje), la plantilla también incluye referencia a la posición de los datos dentro de la hoja de cálculo.

El procedimiento utiliza una variable recordset para acceder a los datos de la base local, elaborando una consulta que elige los 10 valores de mayor longitud de la tabla que se encuentra entre las dos fechas deseadas.

Este recordset únicamente almacena el nombre, el identificador y la duración de la interrupción. El nombre y el identificador se concatenan y se introducen en una columna de la base de datos (correspondiente al eje de valores x), mientras que el dato de la duración se convierte a minutos (está inicialmente en segundos) y se inserta en la columna contigua (correspondiente al eje de valores y). El gráfico se crea de forma automática en Excel.

Como el componente está diseñado para usarse en una aplicación web, debe contemplar la posibilidad de que múltiples usuarios estén corriendo la aplicación al mismo tiempo. Es claro que no es posible abrir dos veces el mismo archivo, por lo tanto el sistema abre la plantilla del gráfico y de inmediato la salva usando otro nombre, que está relacionado con el identificador del empleado, liberando la plantilla.

Otra característica a tener en cuenta es el que el sistema no puede rescribir archivos una vez implementado en la web. Por lo tanto, se agregó un ciclo que pregunta si el archivo existe y, en caso necesario, agrega el valor de un contador al final del nombre. El proceso se repite incrementando el contador hasta encontrar un nombre de archivo válido.

La elección de Excel para construir los gráficos se realizó pensando en la comodidad del usuario, dado que la configuración estándar de la PC de un empleado incluye este software y su uso es muy difundido. Sin embargo, esta escogencia hace más lento al sistema, pues se deben abrir y cerrar aplicaciones Excel frecuentemente. La velocidad del sistema, sin embargo, no es crítica según la opinión de los usuarios, que probablemente utilicen la página web una o dos veces a la semana.

Una vez que se termina el proceso de creación de la página web el workbook u archivo de trabajo se guarda como un archivo html, lo que facilita mucho su inserción en la página web. Con este pequeño truco se crea en forma automática una imagen, de extensión gif, del gráfico, lo que permite mostrarlo en la aplicación web sin necesidad de abrir Excel.

*TOP10FREQUENCY:* Este módulo funciona en forma muy similar al anterior. También está orientado a sistemas multiusuario y utiliza una plantilla muy similar.

La diferencia radica en la forma en que se ordenan los datos. En este caso sólo se almacena el nombre y el identificador de cada interrupción, implementándose un contador para conocer el número de veces que se produce cada una. El nombre y el identificador se concatenan y se escriben en la hoja de datos, con el número de veces que aparecen en la columna contigua. Luego se utiliza la función sort de Excel para hacer el reordenamiento, colocando en la parte superior de la hoja las interrupciones más frecuentes. El gráfico se crea tomando en consideración dos celdas de las 10 primeras filas, por lo tanto, toma las 10 interrupciones que ocurren más veces.

*TIMESTOPPED:* El último de los procedimientos para construcción de gráficos. Este módulo toma los datos de cada handler y obtiene la cantidad de tiempo detenido por máquina (en segundos), luego lo convierte a minutos y presenta la información del tiempo perdido por reparaciones para los 10 robots más ineficientes (con mayor cantidad de tiempo “caído”).

El método para hacer el gráfico en Excel es muy similar a lo que se ha descrito para los dos módulos anteriores.

*GETNUMDAYS*: Obtiene la cantidad de días entre dos fechas determinadas. Útil para el cálculo de los indicadores.

*GETHANDLERS*: Obtiene todos los Handlers disponibles que cumplan ciertas condiciones (tipo de tester, ubicación, etc.). Se utiliza para permitir al usuario agrupar los datos y construir los gráficos para varios Handlers al mismo tiempo.

*FUNCIONES DE OBTENCIÓN DE INDICADORES*: En el código existen varias funciones para la obtención de los indicadores deseados. Se obtienen valores que corresponden al resultado de la aplicación de una función matemática, a saber:

*MTBA (mean time between assists)*: Tiempo medio entre asistencias (interrupciones menores a 5 minutos ). Igual al tiempo transcurrido (en minutos) entre dos fechas dividido entre la cantidad de asistencias ocurridas en el mismo lapso.

*MTTRA (mean time to repair assists)*: promedio de la duración de cada asistencia.

*MTBJ (mean time between jams)*: Tiempo medio entre jams (interrupciones mayores a 5 minutos y menores a 1 hora). Se calcula en forma análoga al MTBA.

*MTTRJ (mean time to repair jams)*: promedio de la duración de cada jam.

*MTBF (mean time between failures)*: Tiempo medio entre fallas (interrupciones mayores a 1 hora). Se calcula en forma análoga al MTBA.

*MTTRF (mean time to repair failures)*: promedio del tiempo que toma reparar cada falla.

*Idle time*: suma del tiempo total perdido debido a asistencias, jams y fallas del sistema.

Para cada una de las funciones descritas anteriormente es necesario acceder a la base de datos, realizar el cálculo matemático y compartir el resultado para su uso fuera del componente dll. De esta forma, cuando se hace referencia a la librería se pueden usar estas funciones como propiedades de la misma, agregando al nombre del objeto al que se le asignó la librería, un punto y el nombre de la función como una extensión.

#### *6.1.2.3 Diseño de la página web:*

La aplicación web fue construida en conjunto con el Ing. Alexander García Sanabria, profesional en computación que trabaja para la compañía. La idea primordial en el desarrollo de la web es facilitar al usuario una serie de gráficos y valores promedio de un Handler en particular, entre dos fechas dadas.

Toda la aplicación web está construida alrededor de la librería dll. Básicamente se trata de capturar los datos que inserta el usuario (su WWID o identificación en Intel, el identificador del Handler para construir el gráfico y las fechas entre las cuales desea el reporte) y llamar a las diferentes funciones de la librería. Luego desplegar los resultados en pantalla (3 gráficos y 7 indicadores)

La página se programó utilizando MS Interdev y los lenguajes html (hipertext markup language), Visual Basic Script (subconjunto de Visual Basic que se emplea en aplicaciones web) y Java script (subconjunto de Java permitido en la web).

La página de entrada se presenta a continuación:



Figura 7 Página de entrada del sitio web.

Las funciones son

**START DATE:** despliega un calendario para elegir la fecha inicial. El calendario se implementó usando Java Script.

**END DATE:** la misma función para escoger la fecha final

**TESTER TYPE:** Botón de tipo radio que permite seleccionar el tipo de tester (IX, ST, ST2 y OLF). Dependiendo de esta elección se despliegan los Handlers disponibles para ese tipo de tester.

**SITE:** Edificio o planta. Permite escoger los handlers de acuerdo a su ubicación física.

**Handler ID:** selecciona el identificador específico del handler.

*Submit:* Enviar la información para la construcción de los gráficos.  
Una vez enviada la información, la página cambia (véase figura 6.6)

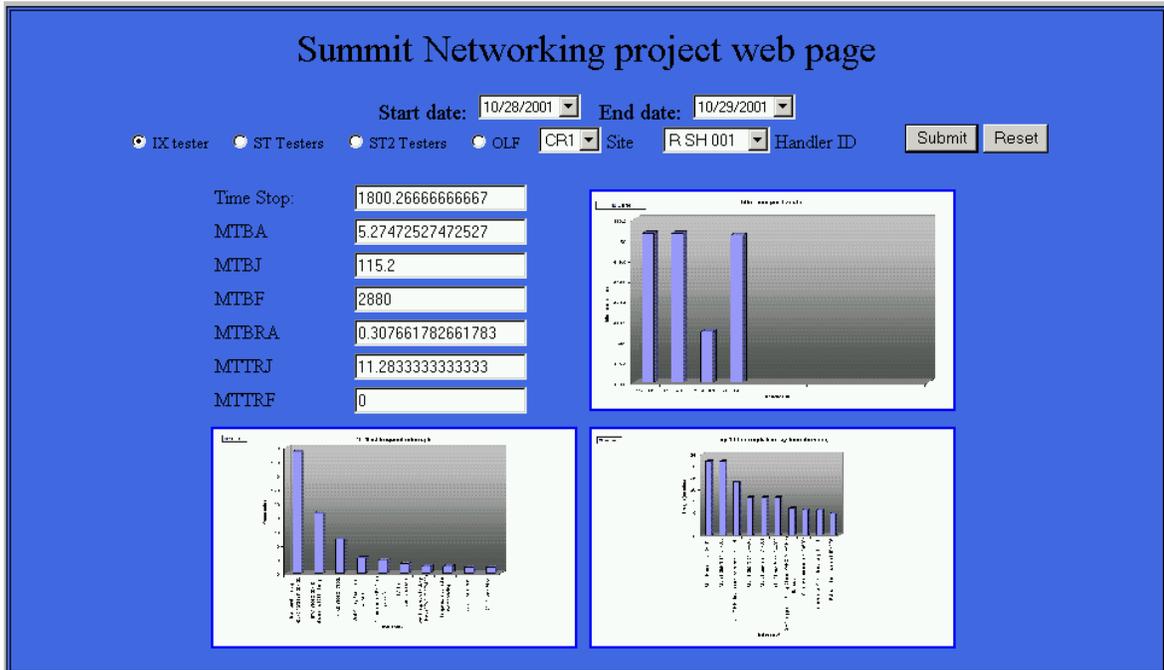


Figura 8 Despliegue de indicadores en la página web.

La imagen del Summit ha sido sustituida por los indicadores de desempeño y tres gráficos, como se ha descrito. Todas estas funciones se explican con detalle en la descripción del componente SHProject dll.

Dada la imposibilidad de acceder de forma remota todos los handlers (la empresa no ha entregado las direcciones IP) en este momento la página funciona basada en datos almacenados en el disco duro, en una copia exacta de la base de datos que se guardará en el servidor. En realidad, el sistema siempre va a

acceder a una base en el disco duro y no en el handler (con un programa que efectúa refrescamientos, el Summit Loader) por lo que no habrá cambios cuando los handlers se encuentren conectados.

### 6.3 Características del proceso de solución:

Debido a las características de la empresa Componentes Intel de Costa Rica el trabajo no se limitó al área técnica solamente sino que el estudiante debió promocionar el proyecto ante diversas instancias y coordinar el trabajo de varias personas. El trabajo en equipo fue fundamental en el desarrollo y el aprendizaje en manejo de proyectos es invaluable. Algunas de las gestiones realizadas fueron:

#### *6.1.3 Coordinación con el fabricante del Summit Handler:*

La empresa Delta Designs brinda soporte técnico relacionado con el Summit, de hecho, mantiene varios ingenieros en la planta para ese propósito. Uno de los primeros pasos fue contactar a los representantes de Delta para aclarar algunos aspectos del funcionamiento del Handler y de las bases de datos. Básicamente, se trataba de verificar la forma en que se guardaban los datos en esas bases, el significado de algunos campos y obtener derechos sobre su uso (las bases estaban protegidas mediante una clave). Delta se mostró interesada en el proyecto y no tuvo reparos en conceder los derechos, sin embargo, la información brindada sobre los datos fue muy escasa. De hecho, los indicadores a desplegar se obtuvieron de acuerdo a lo que pudo inferir el practicante.

Otro punto tratado fue la instalación de software adicional en los Handlers, a fin de proteger a la herramienta contra los virus. Este cambio representaba un pequeño riesgo: algunos archivos del sistema se podían corromper y la herramienta estaría caída por varias horas (lo cual es una pérdida importante

dado el precio del sistema). Por ello, fue necesario diseñar un conjunto de experimentos que garantizaran que el rendimiento del equipo no se iba a afectar. El fabricante se mostró anuente a colaborar y concedió permiso para utilizar máquinas con garantía para dichos experimentos (y así contar con el apoyo de los ingenieros de Delta).

#### *6.1.3.1Diseño de experimentos:*

Se implementaron 4 experimentos como garantía de que la adición de software antivirus no iba a impactar el rendimiento. Los programas a instalar eran:

- Service Pack 6a: Actualización de Windows NT 4.0
- McAfee Antivirus.
- McAfee Service Pack 1
- McAfee Sdat. Última definición de virus.
- Hot Fix. Parche contra diversos virus.
- Nimda patch. Parche contra el virus Nimda.

En los apéndices se puede encontrar la guía de instalación de este software, para su implementación en otros sitios de Intel.

El detalle de cada experimento se reserva por cuestiones de confidencialidad. A continuación una breve reseña de cada uno de ellos.

*VERIFICACIÓN DE COMUNICACIÓN ETHERNET:* El Delta Summit Handler tiene varias tarjetas que se comunican con el ordenador central usando el protocolo TCP/IP. La idea era verificar que el ordenador tenía comunicación exitosa con todos los demás componentes, usando el comando ping.

*COMUNICACIÓN GPIB:* El Handler utiliza un puerto GPIB como interfaz con el tester. Se verificó dicha comunicación usando una herramienta de diagnóstico suministrada por el fabricante. Para ello se utilizan unidades especiales.

Se corrieron 120 piezas sin ningún problema.

*ULT READER:* El ULT reader es una cámara que obtiene el número de cada unidad. Esta prueba fue realizada en forma simultánea con la anterior.

*MEDICIÓN DE UNIDADES POR HORA:* En este apartado se contó con el apoyo del departamento de Ingeniería Industrial. A grandes rasgos, se debía contar el número de unidades por hora que procesaba el Summit antes y después de la instalación. Los resultados fueron bastante similares, con una diferencia menor al 5% (normal debido a la influencia de ciertos factores externos, como pequeños jams o errores) y el cambio fue validado por el departamento respectivo. Los datos exactos se reservan por confidencialidad.

Todos estas pruebas fueron aprobadas en diferentes foros de la empresa utilizando un procedimiento especial (que incluye formularios con el detalle de cada cambio planeado). En primer término se someten al grupo de dueños (tool owners) y usuarios del Summit, y luego se revisan en un foro constituido especialmente para aprobar o desechar ese tipo de documentos. El proceso fue exitoso, sin embargo representó un retraso en los demás objetivos del proyecto.

## 6.2 Alcances y limitaciones

Al momento de presentarse el proyecto se han alcanzado los siguientes objetivos:

- Todos los handlers cuentan con el respectivo upgrade de software. Esto significa que sólo se requiere el cambio de la dirección IP y colocar el cable para lograr la conexión. Lamentablemente, la asignación de las direcciones IP implica la creación de nuevos segmentos de red por lo que se debe esperar a que estos segmentos estén debidamente cableados para proceder a conectar todos los handlers.
- El servidor está configurado y listo para su inserción en el centro de cómputo. Es deseable realizar una prueba con todos los handlers conectados, por ello la instalación en dicho centro se haría después de la conexión de los Handlers.
- El programa de carga de la información está listo.
- La librería que contiene todas las funciones de procesamiento de los datos (construcción de gráficos, obtención de promedios, etc.) está lista.
- La página web funciona desde un servidor provisional.
- La guía de instalación del upgrade de software está terminada.

El único objetivo por cumplir es finalizar la conexión de los Handlers a la red.

### 6.2.2 Limitaciones:

El sistema funciona con una base de datos Access, que no es óptima en velocidad ni en capacidad. Para mejorar este aspecto, se debe migrar esta base de datos a SQL Server. Sin embargo, este cambio no es urgente puesto que el sistema es empleado por cada usuario una o dos veces a la semana.

El hecho de que los gráficos se construyan en Excel, que es una aplicación muy pesada (de varios MB, el proceso de abrir y cerrar archivos es lento) restringe significativamente la rapidez de ejecución de la página web. En contraposición, el usuario puede guardar la información en un formato conocido, o bien modificarla. La velocidad del sistema no es crítica, como sí la facilidad de manejo.

Otra restricción es la falta de control sobre el apagado de los robots. Los datos que se generan corresponden únicamente a lo que el robot registra. Si la PC se apaga, el robot no produce ni registra errores. El sistema, por lo tanto, no va a reportar ningún dato durante el tiempo de apagado. Esto produce la ilusión de un rendimiento perfecto –sin errores- durante el apagado lo cual es claramente erróneo. Para detectar el apagado de las máquinas se requiere un programa extra que se sale de los objetivos del proyecto.

El sistema tampoco sincroniza los relojes de las computadoras. Esto es necesario para contar con solo un tiempo de referencia. Uno de los programadores de la empresa ya trabaja sobre este aspecto y antes de arrancar el sistema con todos los handlers estos van a ser sincronizados manualmente. Una vez mas, este aspecto no forma parte de los objetivos del proyecto.

La última limitación es la subutilización de la base de datos. Si bien es cierto que se extrae la información más importante de ésta, algunas tablas no usadas contienen datos que podrían ser útiles. El problema radica en que no es claro el

significado de dichos datos ni la forma en que se obtienen. Por ejemplo existen dos tablas que presentan el mismo formato, el mismo nombre de los campos y valores diferentes. Además, algunas veces se obtienen promedios sin indicar el intervalo abarcado (no se puede determinar si se trata de un promedio diario, semanal o mensual). La ayuda del fabricante en este sentido ha sido muy escasa.

## 7. Conclusiones

1. Para protección antivirus del Delta Summit Handler se utilizó software adicional en lugar de una conexión exclusiva, a fin de reducir costos.
2. Los datos del Summit Handler se extraen descargando en forma remota una base de datos local, mucho más rápidamente que si se emplea líneas de comando del software.
3. La página web de acceso despliega un conjunto de 7 indicadores y 3 gráficos.
4. El acceso a las bases de datos Ms-Access se implementó usando tecnología ADO (ActiveX Data Objects) de Microsoft.

## 7.2 Recomendaciones

1. Incrementar el uso de la base de datos, coordinando con el fabricante (Delta) la forma en que se pueden procesar los datos no considerados, en especial los datos relativos a la entrada / salida de usuarios (mediante el password y el logon) y al comportamiento de los chucks (componentes del Summit).
2. Cambiar el formato de la base de datos de Access a SQL Server.
3. Automatizar la descarga de algunos archivos de configuración del Summit Handlers, como las “recetas” (configuración del handler según el producto).
4. Crear un programa que detecte si el Handler está apagado. Esto haría más completa la información de la página.
5. Automatizar la sincronización de los relojes de cada Summit. Actualmente esto se hace en forma manual. El programa para tal fin debe correr en el servidor.
6. Actualizar el servidor cuando el tamaño de la base de datos crezca.
7. Incorporar los datos reportados en la base de datos corporativa AEPT.

# Bibliografía

**Visual Basic 6: Curso de programación.** Fco. Javier Ceballos. Editorial Alfaomega, México 1999.

**Bases de datos con Visual Basic.** Jeffrey McMannus. Editorial Prentice Hall, España 1999.

**Inside Microsoft Visual Basic, Scripting Edition.** Scot Hillier. Editorial Microsoft Press, Estados Unidos de América, 1996.

**HTML, la guía completa.** Chuck Musciano y Bill Kennedy. Editorial McGraw-Hill, México 1999.

Enlaces de internet.

[www.lawebdelprogramador.com](http://www.lawebdelprogramador.com) Cursos completos relativos a lenguajes de programación.

## **Apéndices**

# A1 Glosario

Asistencia: cualquier error en el sistema que toma menos de 5 minutos y que es reparado por el operario sin intervención de personal técnico especializado.

Assembly/Test: Ensamblaje y prueba.

Delta: hace referencia al fabricante del robot Summit Handler, Delta Designs.

Dll: formato con que se guarda y utiliza la librería de funciones para la aplicación web.

Ethernet: Tipo de red de computadoras.

Excel: Aplicación que pertenece al conjunto de programas Microsoft Office. Hoja de cálculo utilizada para el procesamiento de los datos.

Falla: Cualquier interrupción que toma más de 1 hora para ser reparado por personal especializado.

Gif: formato de compresión de archivos que contienen imágenes.

GPB: protocolo de comunicación para cierto tipo de puerto de datos. Se utiliza en la comunicación entre el tester y el handler.

Handler: Hace referencia al robot Delta Summit Handler, un sistema electromecánico usado para someter cada procesador a distintas pruebas. El Handler no realiza la prueba sino que coloca las condiciones necesarias y mueve el chip hasta la tarjeta de prueba.

Html: hypertext markup language. Lenguaje de programación usado en páginas web.

Hub: repetidor de red o concentrador. Se utiliza para regenerar la señal eléctrica en redes de computadoras.

Indicador: valor numérico que representa el comportamiento de alguna variable del sistema.

Intranet: Red de computadoras interna a una compañía u organización.

IP: identificador único necesario para la inserción de cualquier componente de una red de computadoras.

Jam: interrupción en el sistema que toma entre 5 minutos y hora para ser reparado.

MSDN: librería de ayuda para la programación con el conjunto Visual Studio, de Microsoft.

Pentium: Nombre que reciben las últimas generaciones de procesadores de Intel.

Receta: Archivo de tipo texto que se utiliza para definir las condiciones de prueba del procesador.

Recordset: tipo de variable en Visual Basic que permite hacer consultas a bases de datos

Sql: structured query language. Conjunto de instrucciones para acceder bases de datos relacionales.

SUN: estación de trabajo. Utiliza sistema operativo UNIX

Switch: componente de red que permite ciertas funciones para el enrutamiento de los datos.

Test: Prueba.

Tester: Probador

TO: Caja de conectadores. Contiene dos puertos de datos y un puerto telefónico.

Upgrade: Actualización.

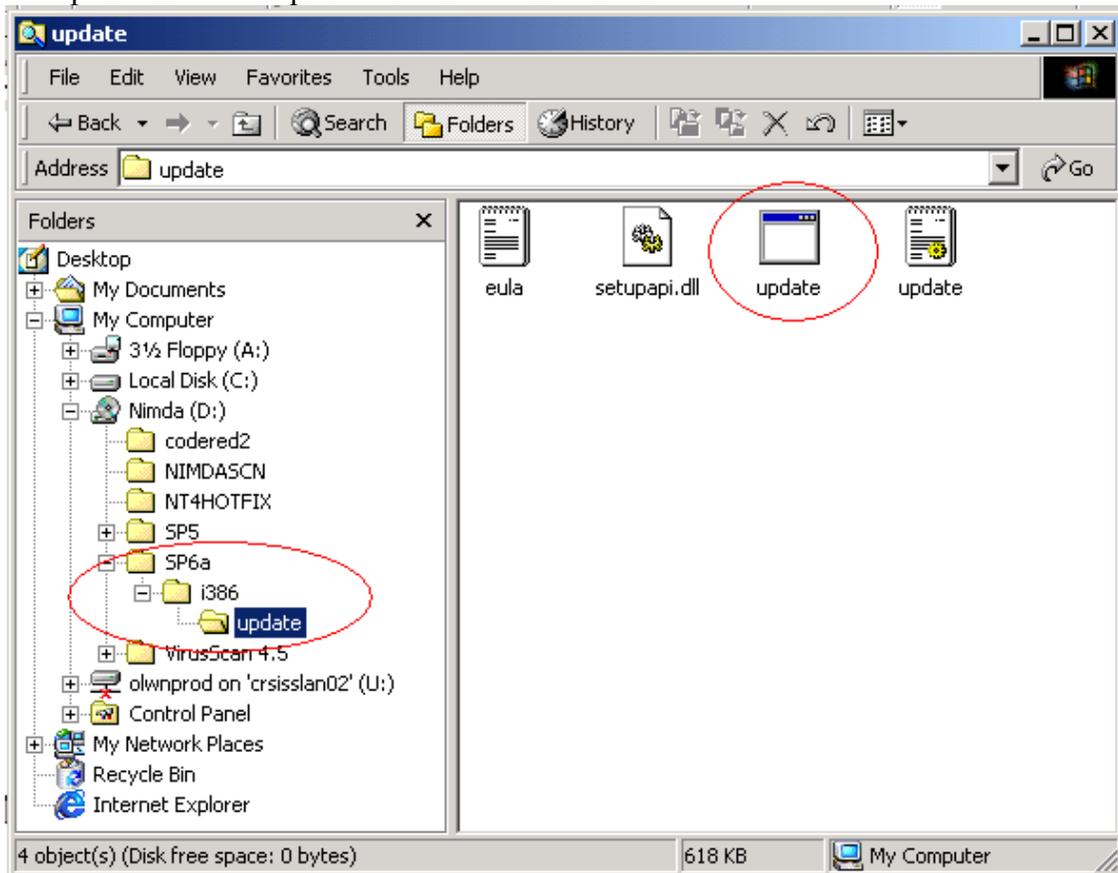
VLAN: red virtual de área local.

## A2 Guía de instalación del Software antivirus para el Summit Handler

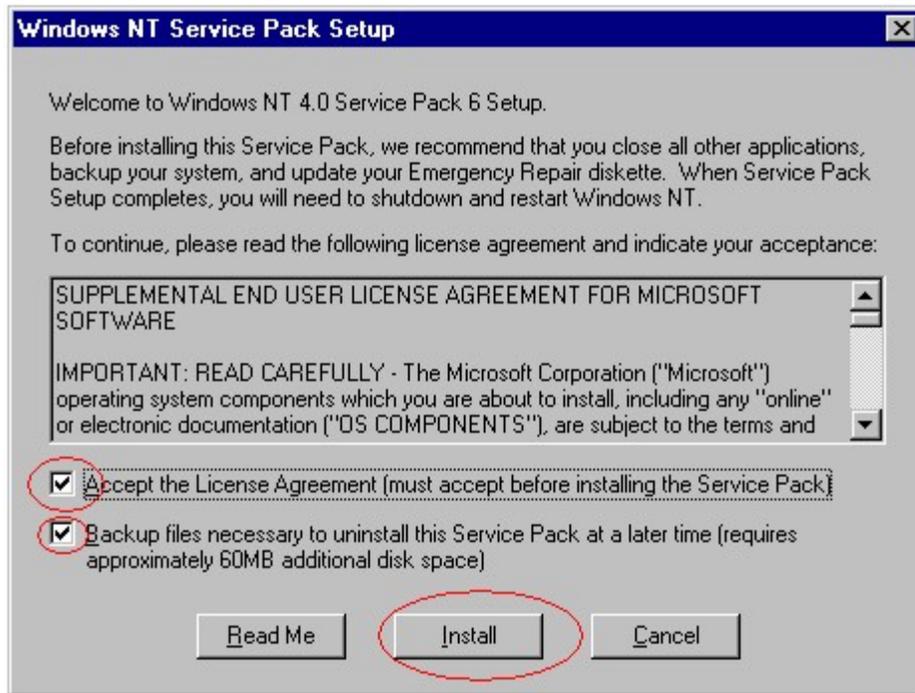
### System OS and antivirus upgrade. Installation guide

#### Service Pack 6a

1. Open Windows Explorer and choose the CD drive.

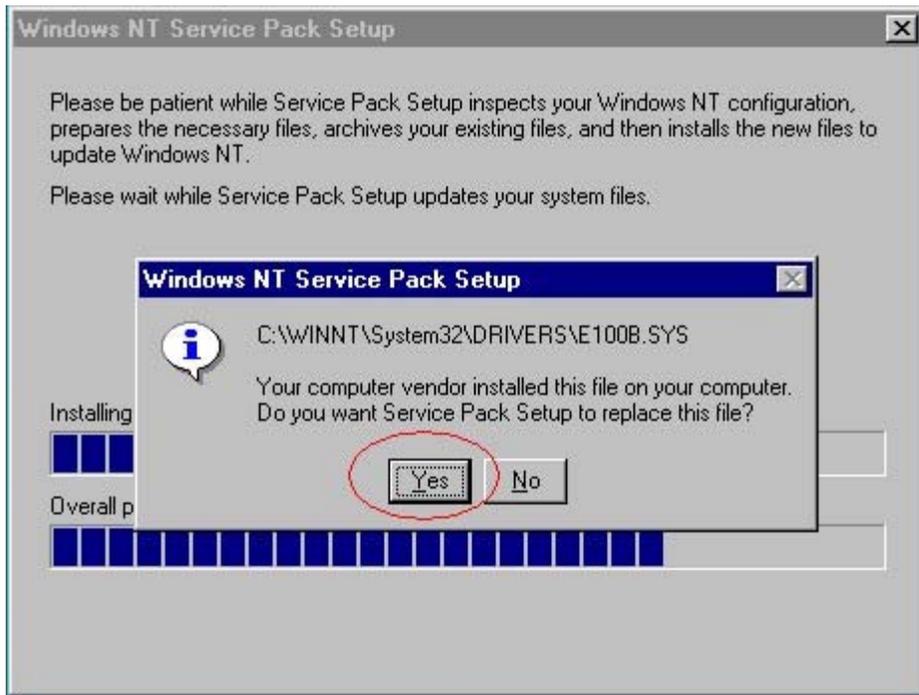


2. Choose the SP6a folder. Then open the ...\i386 folder, update folder and update.exe
3. The following box will appear



When you accept the License Agreement the Install button will be enabled. You must choose the backup option because it is more reliable. Although the risk is very low (we didn't have problems at Costa Rica) some Windows files could be corrupted, and the uninstall is an option to repair the problems.

3. When the system asks for overwrite, choose yes always. Two files with the same name could be different.



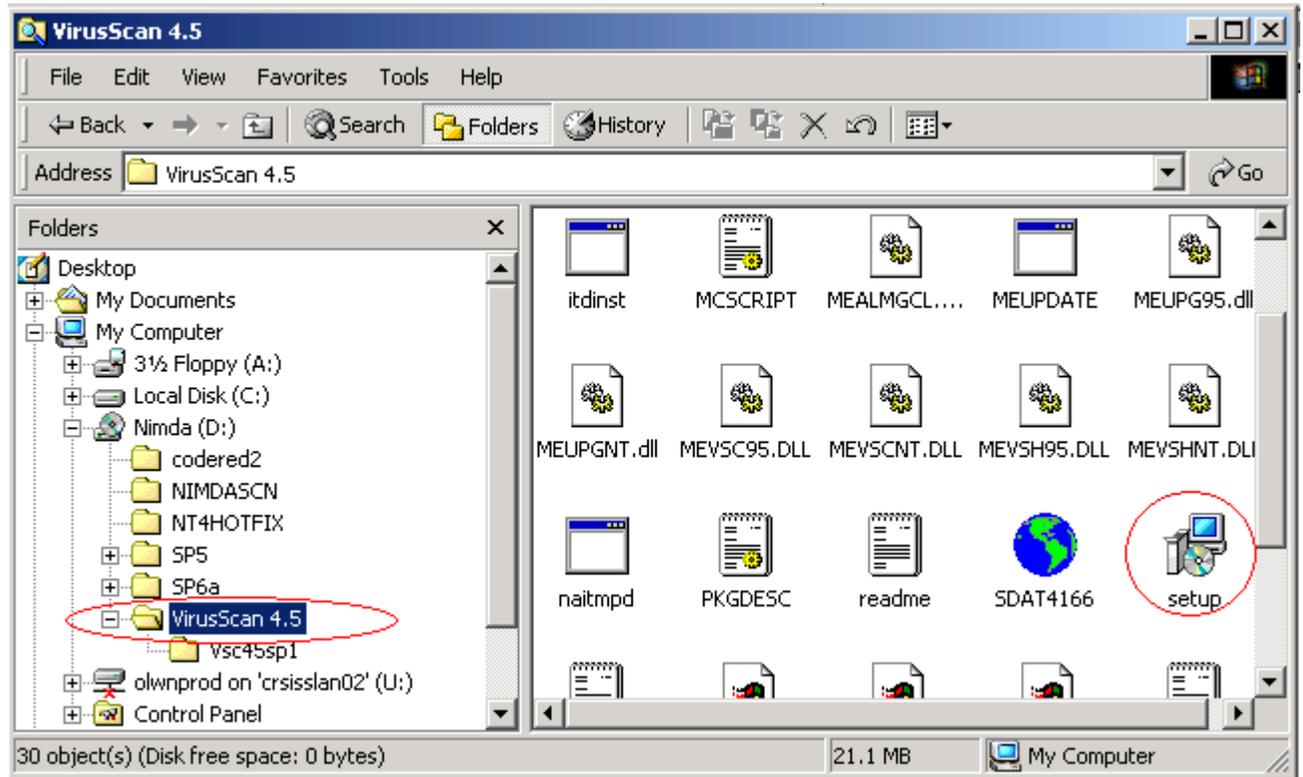
4. Finally, restart the system.



5. You can check the installation was successful opening the Windows explorer again. Click on Help menu, the Choose about Windows NT and the Service Pack version will appear into the box.

## McAfee Virus Scan.

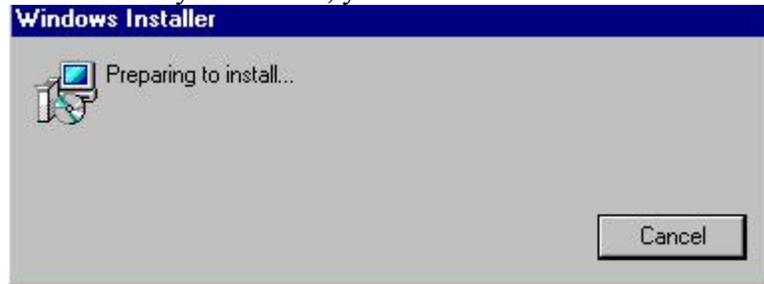
1. Open the Windows Explorer program and choose the CD-drive. Then, click on setup.exe file.



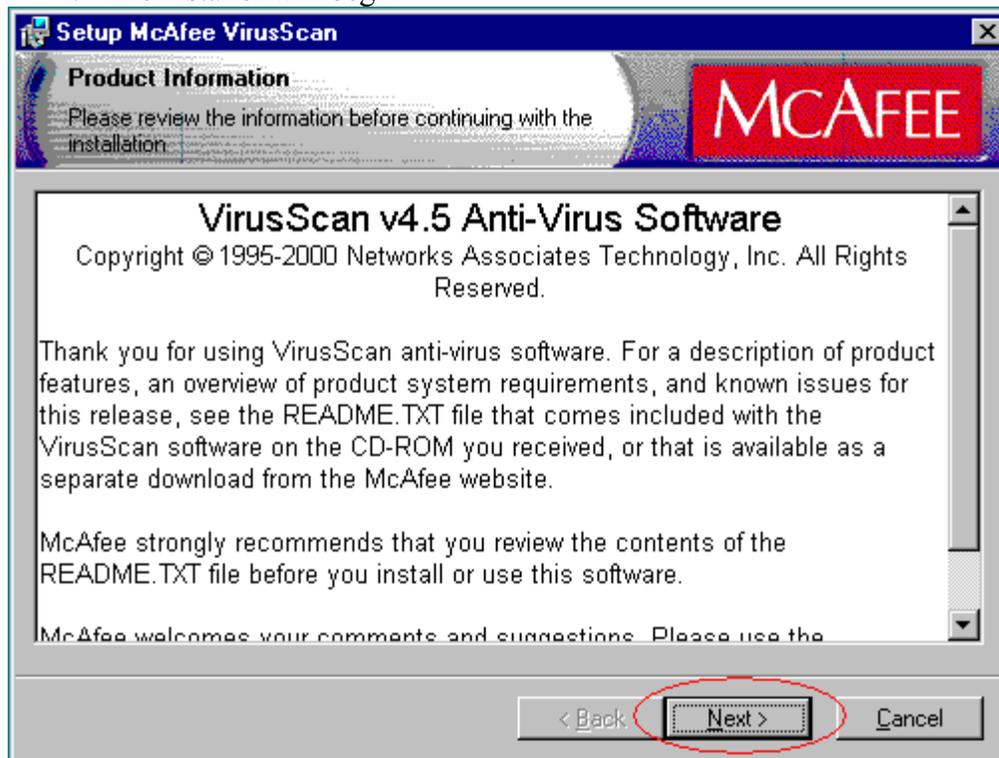
2. You must restart the PC.



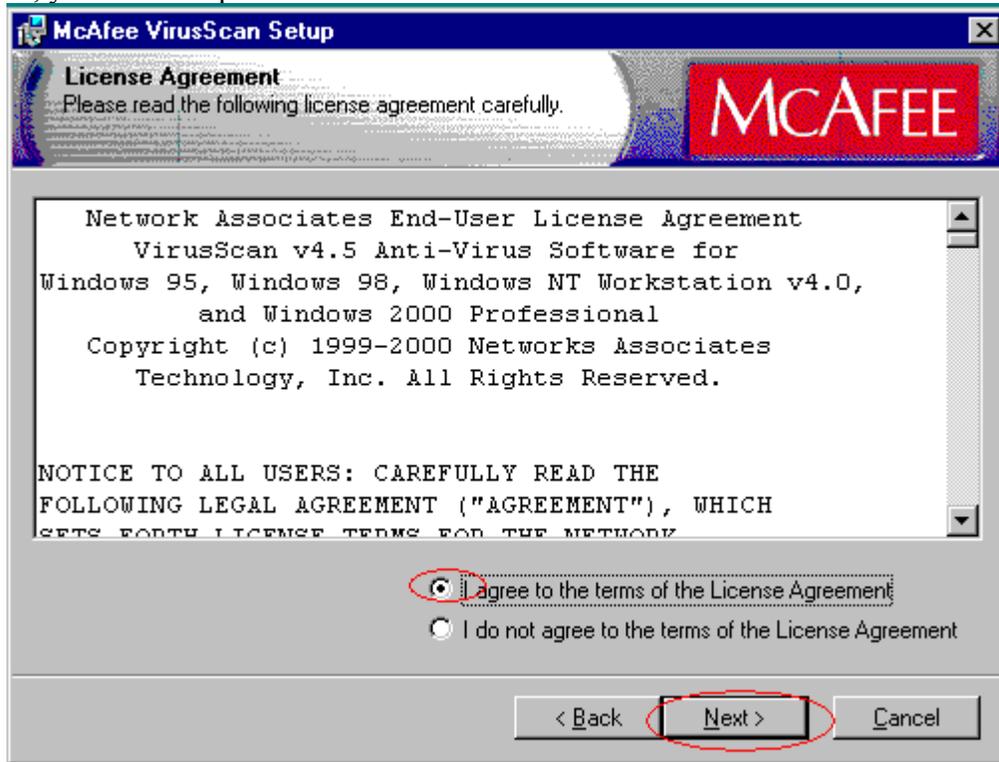
3. When the system starts, you should see this window



4. The installer will begin



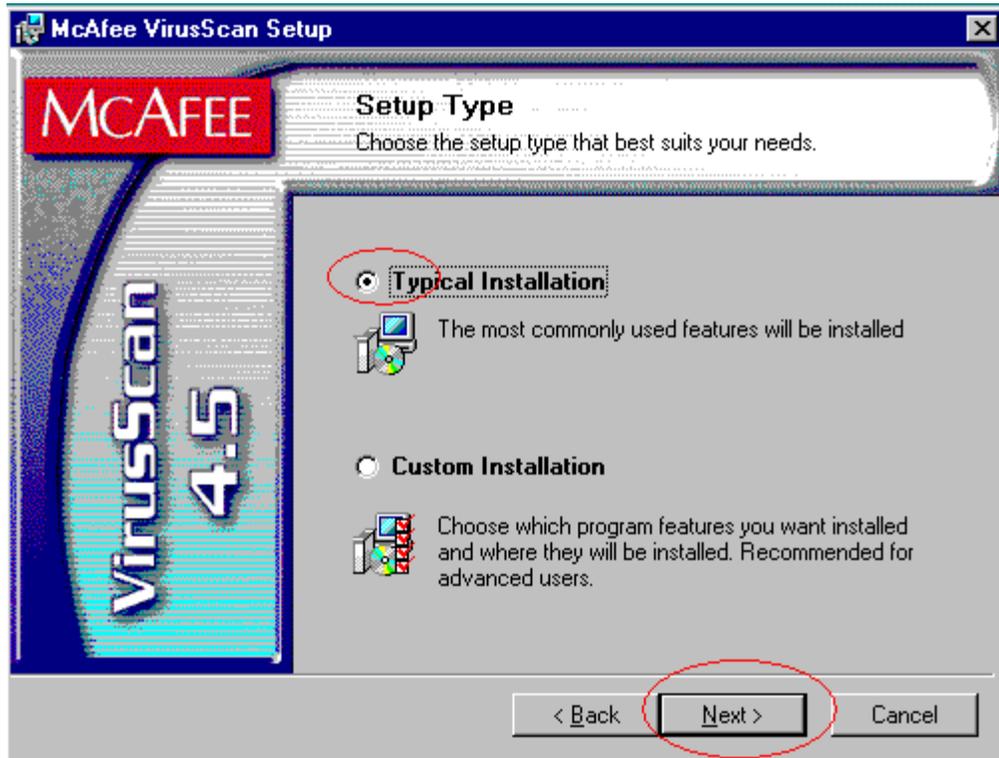
Then, you will accept the license



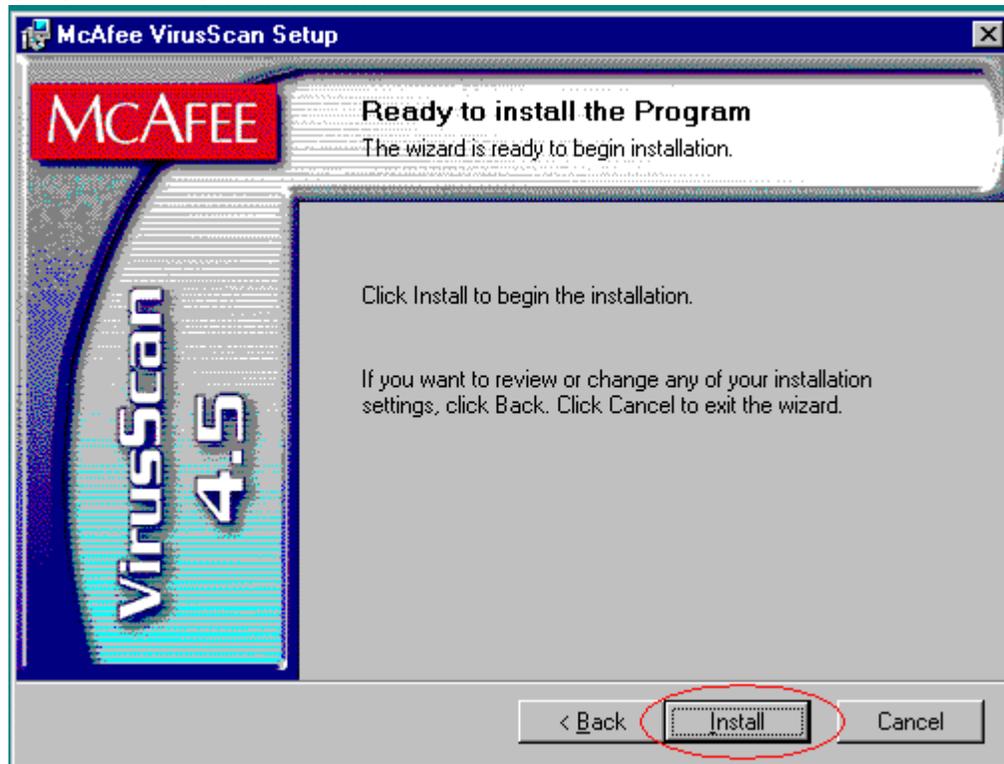
5. After this, you will choose the maximum security option



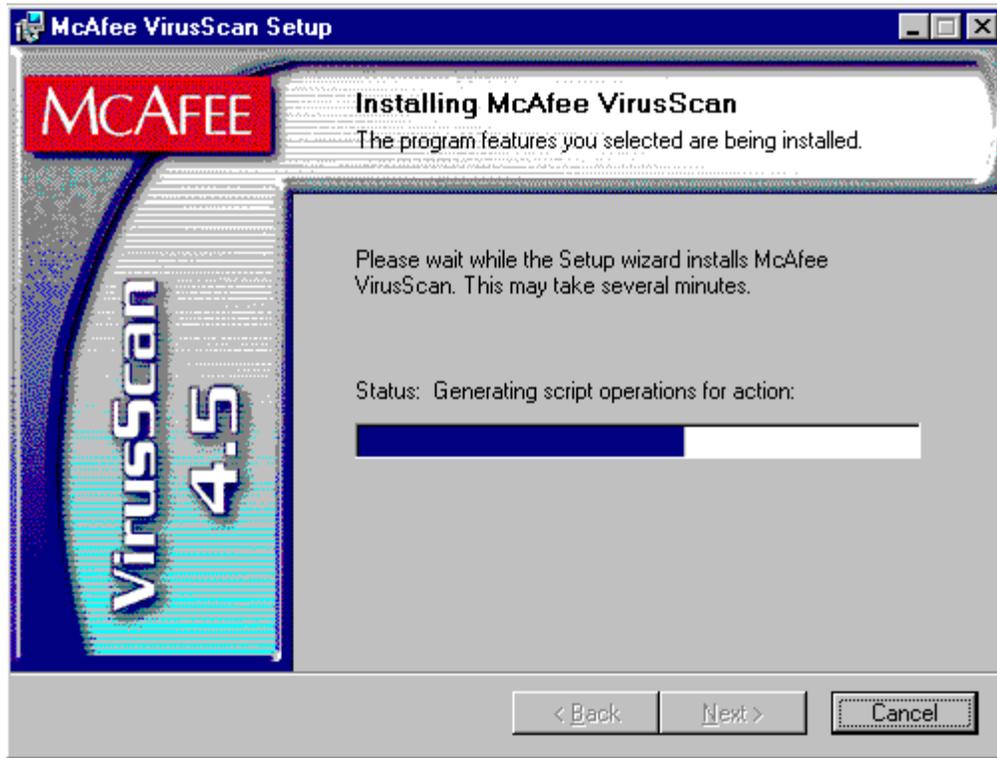
7. Then, click on Typical Installation



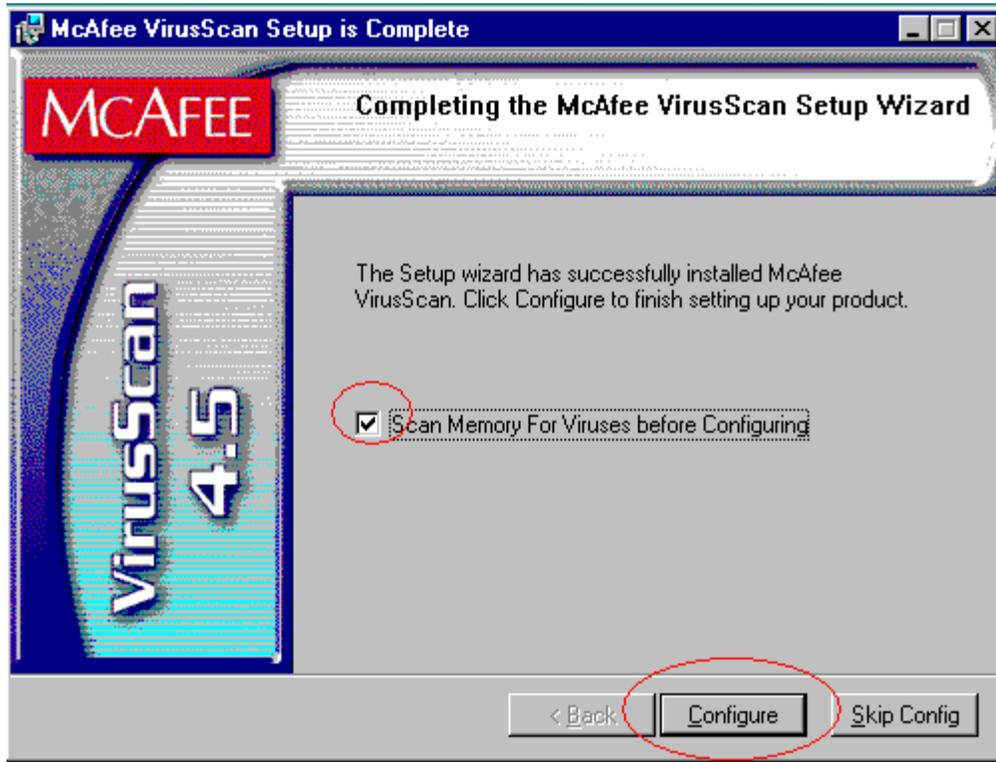
8. Start the installation



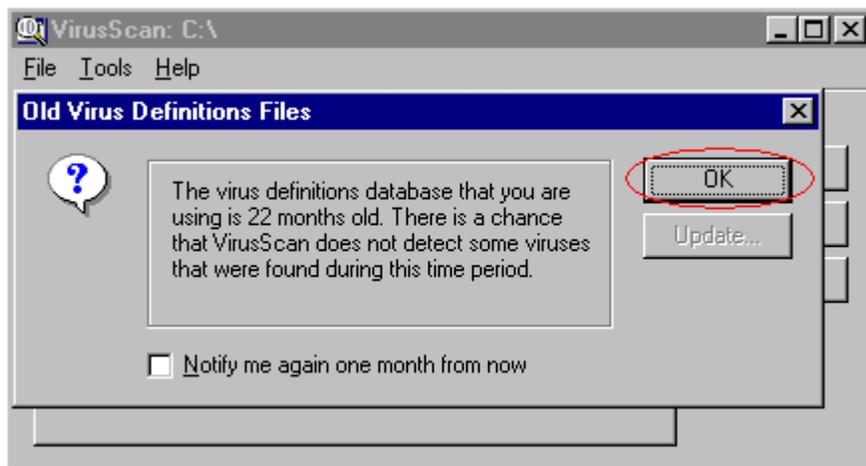
9. The following window will be visible



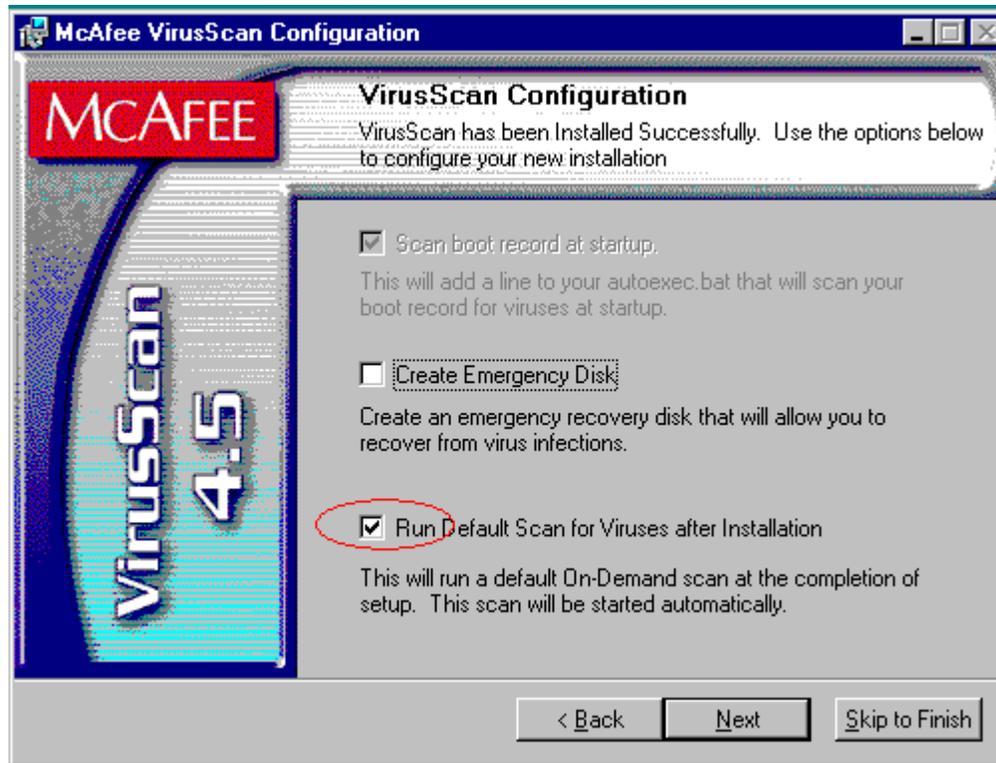
10. Choose the scan before configuring option



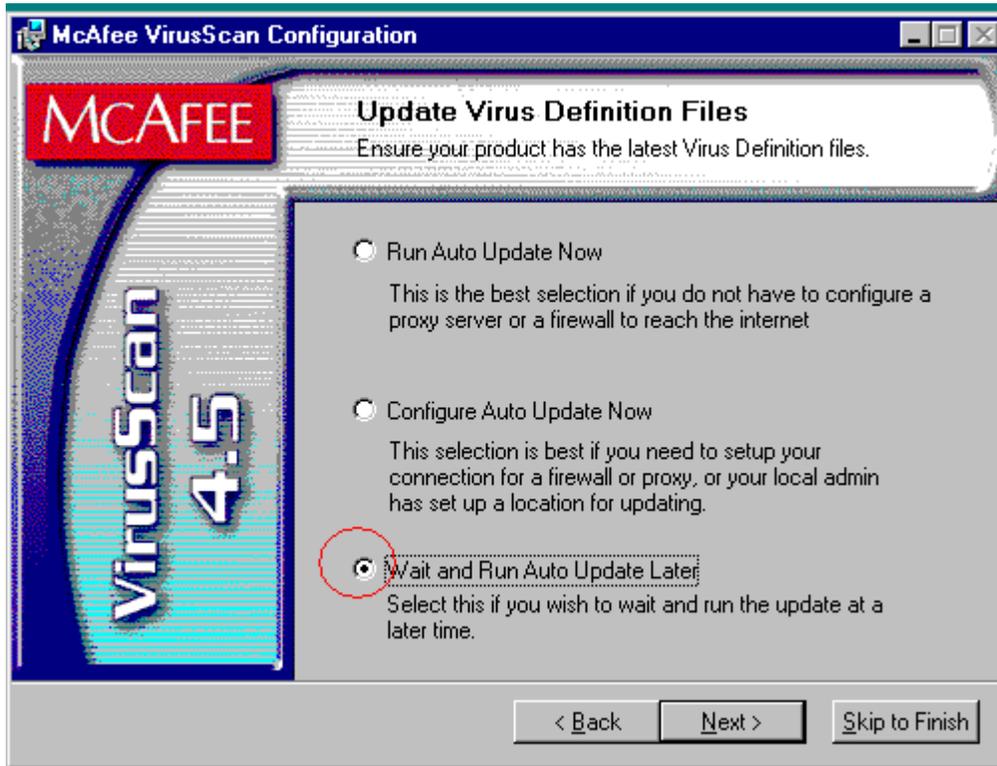
11. Click on OK. The virus definition database will be updated later.



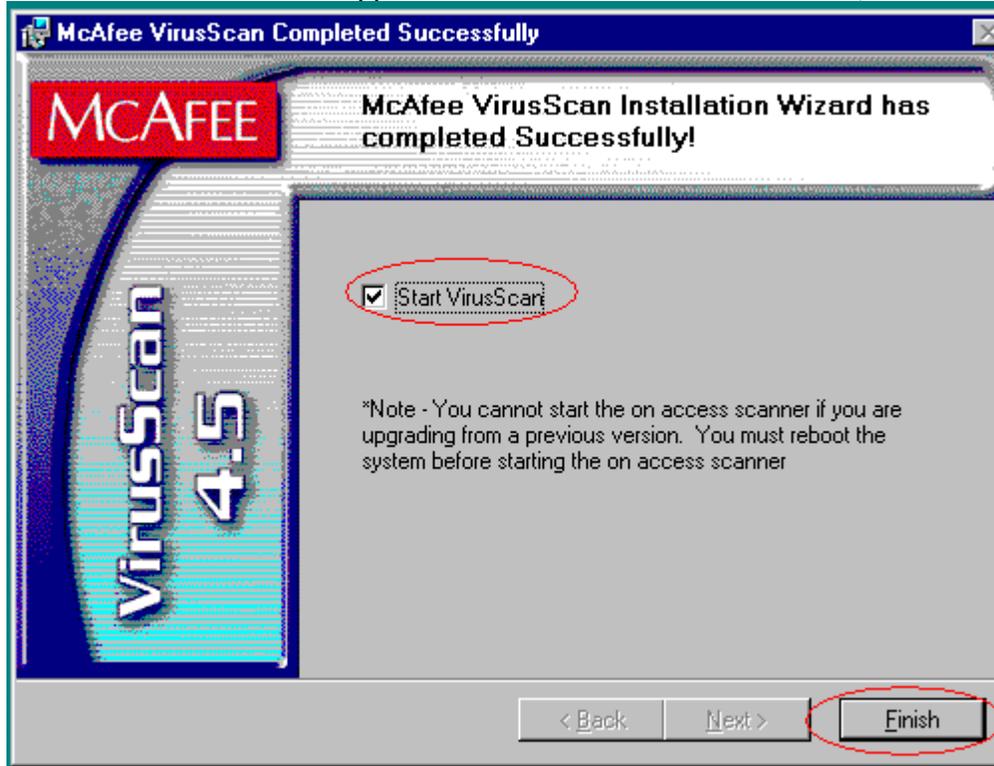
12. Choose Run Default Scan for Viruses after Installation



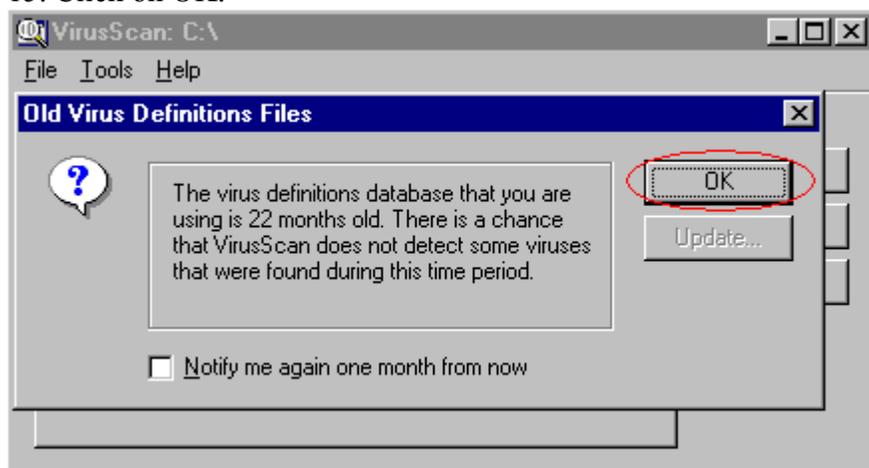
13. Choose Wait and Run AutoUpdate later. This task may have problems without internet connection.



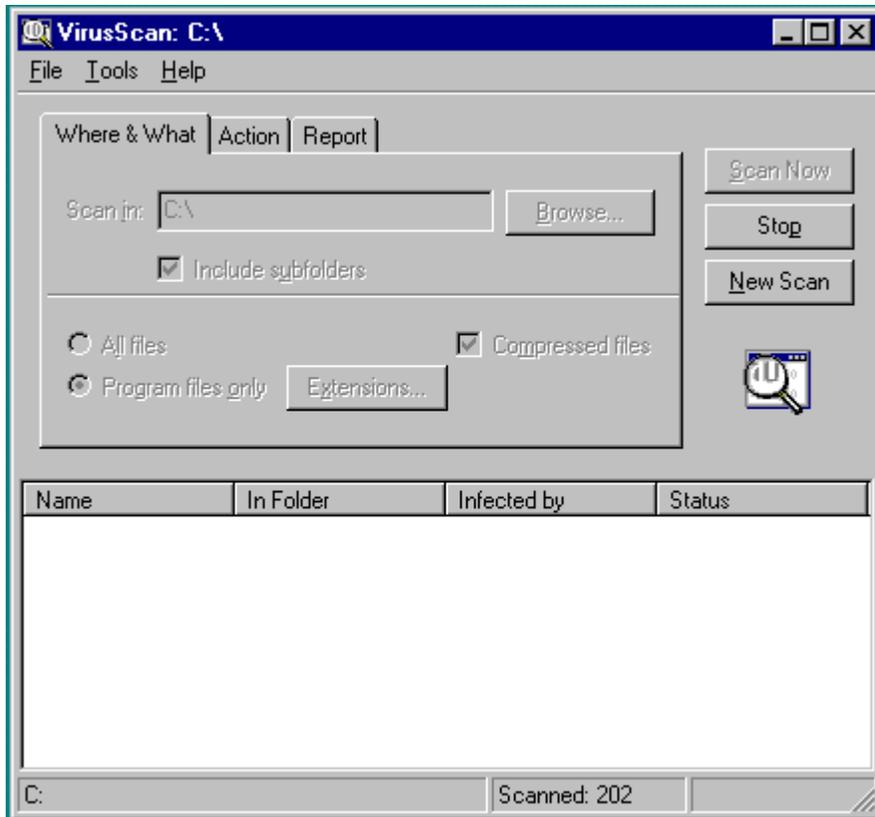
14. The next window will appear. Check Start Virus Scan and then, Finish



15. Click on OK.

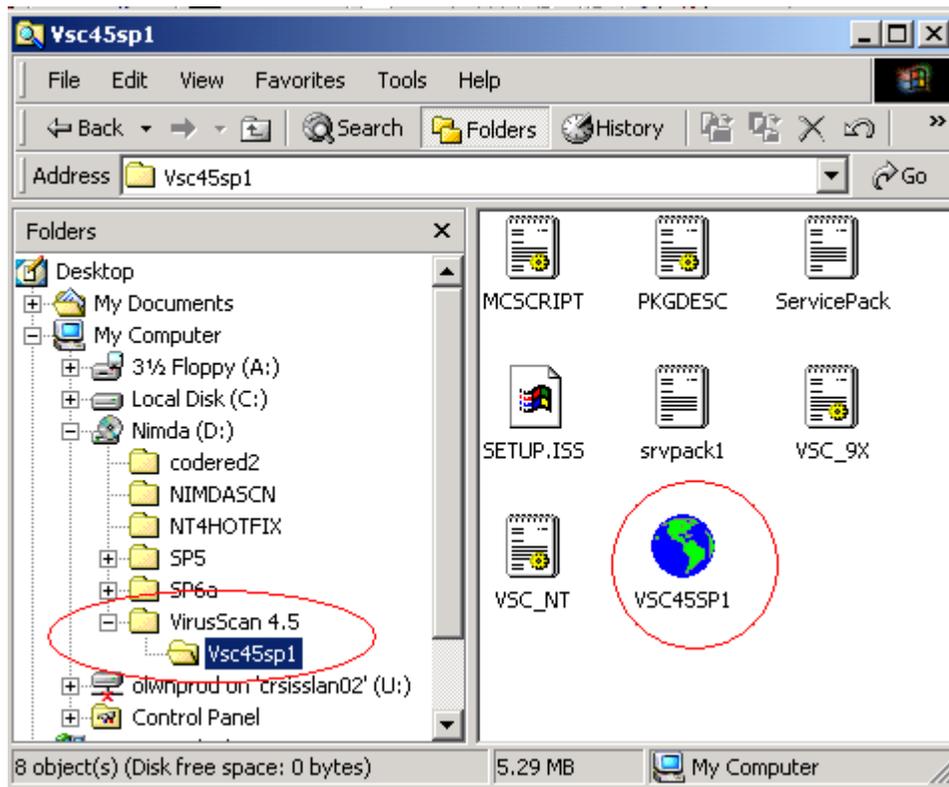


15. The Virus Scan will scan the hard disk

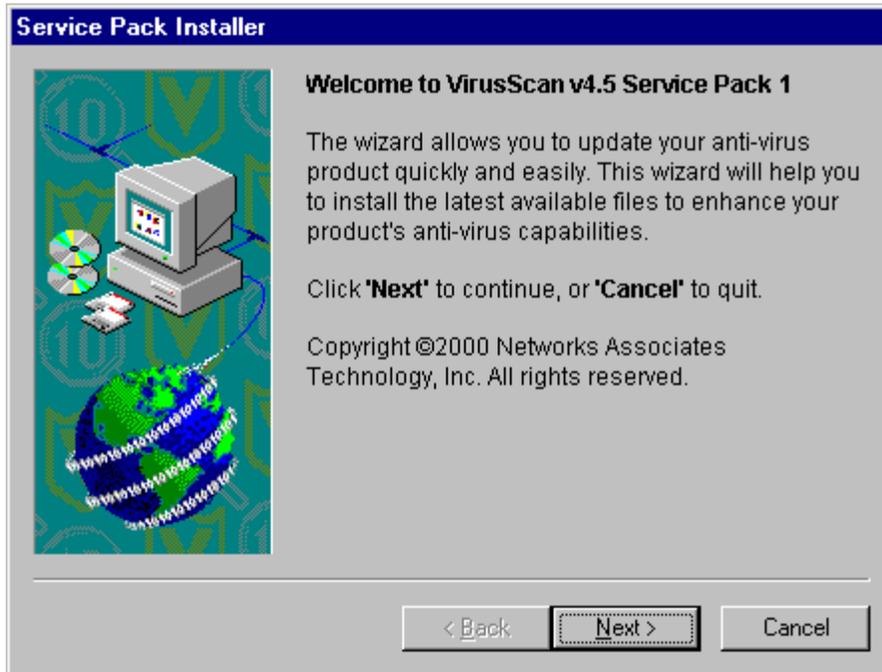


## McAfee Service Pack

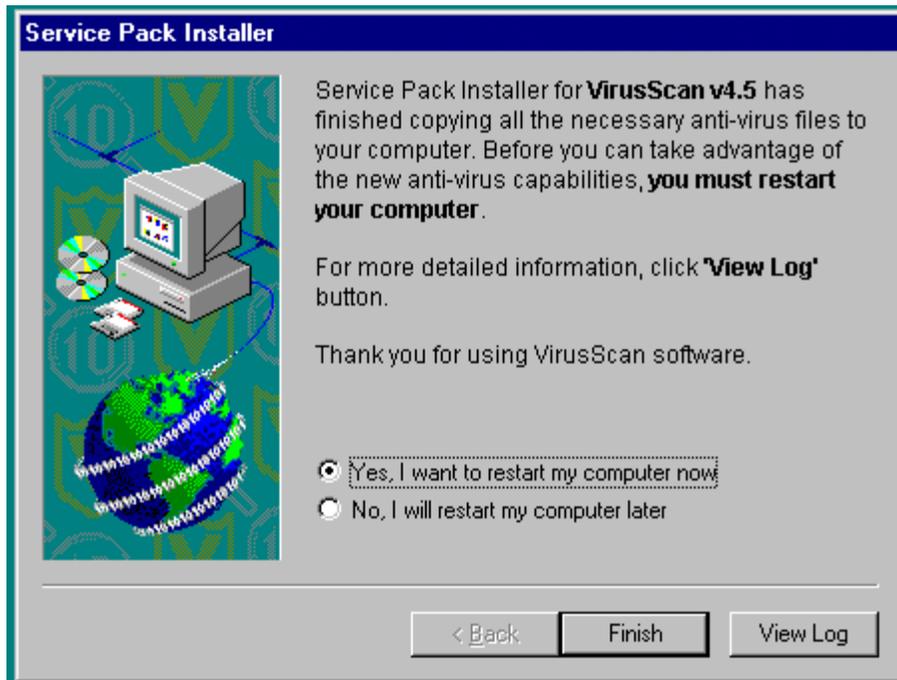
1. After McAfee Installation choose the VSC45SP1 file



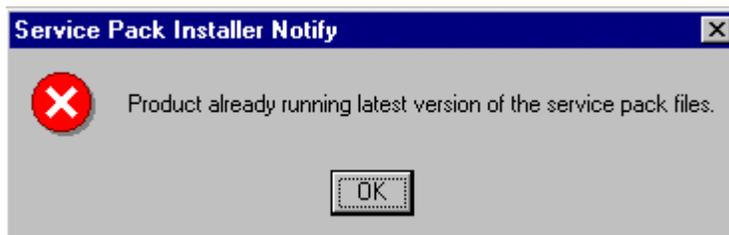
2. The following box will appear



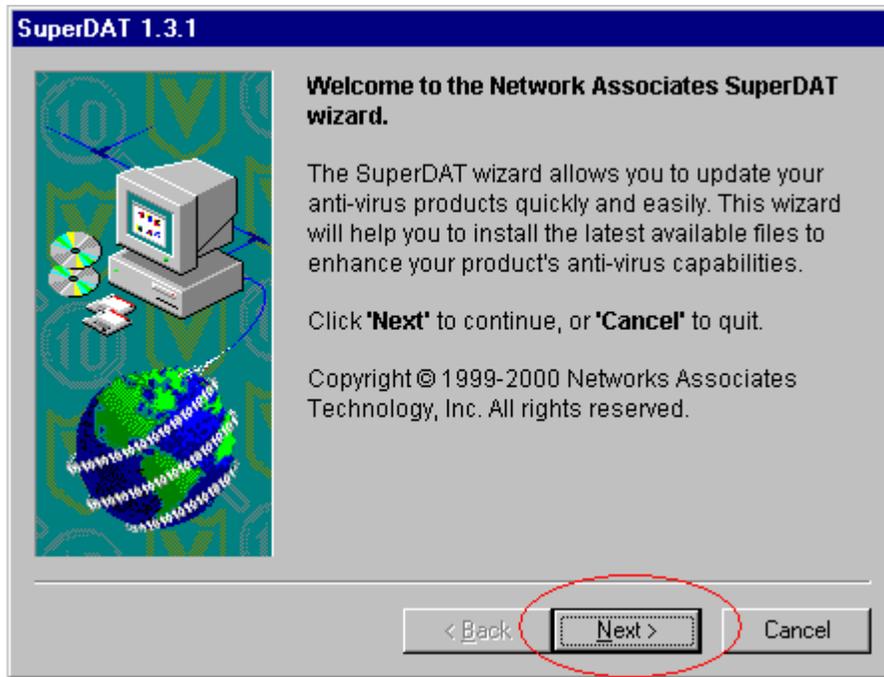
3. When you click Next the files will be installed. After this, you must restart the system.



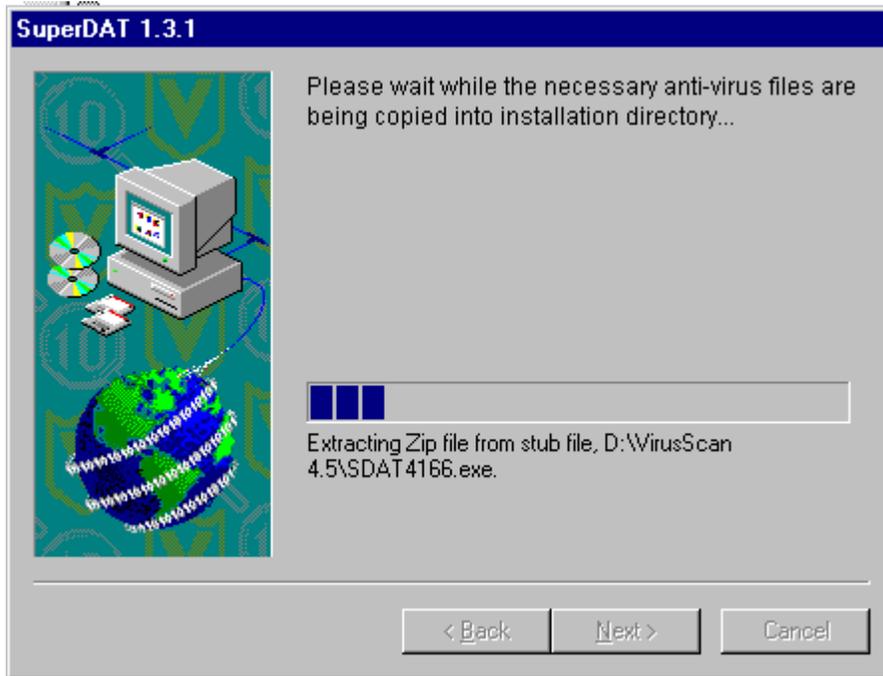
4. You can be sure the installation was successful by redoing the steps 1-3. This message will appear.



## Sdat Installation (Virus List Upgrade)

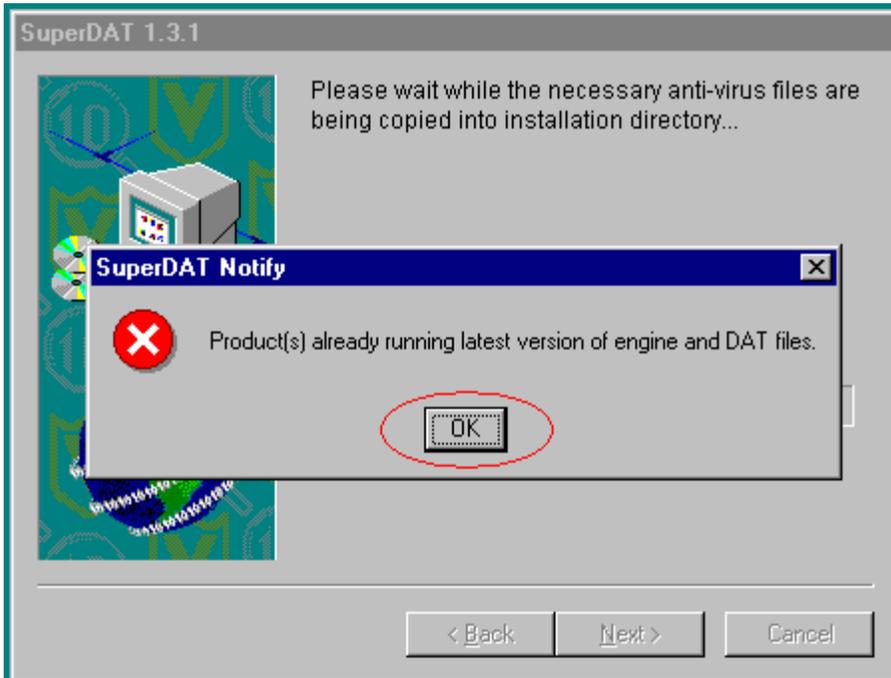


Then you will see:



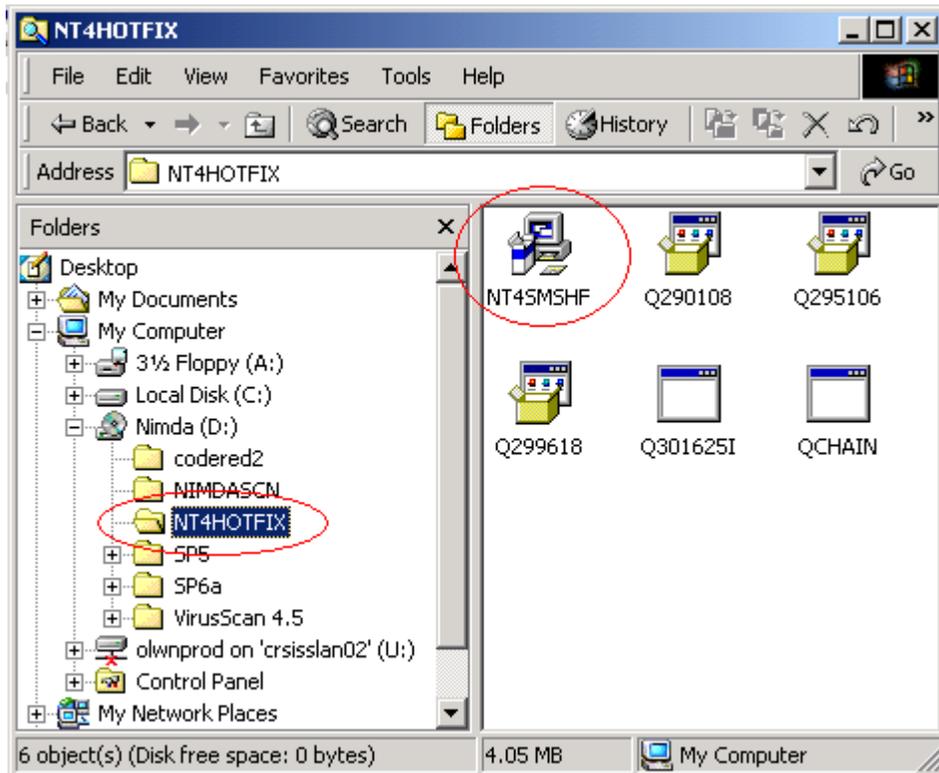
You don't need restart the computer after the Sdat upgrade. You can verify that the installation was successfull by redoing the steps 1-2. You should see the following image.

Finally:



## NT4 Hot Fix:

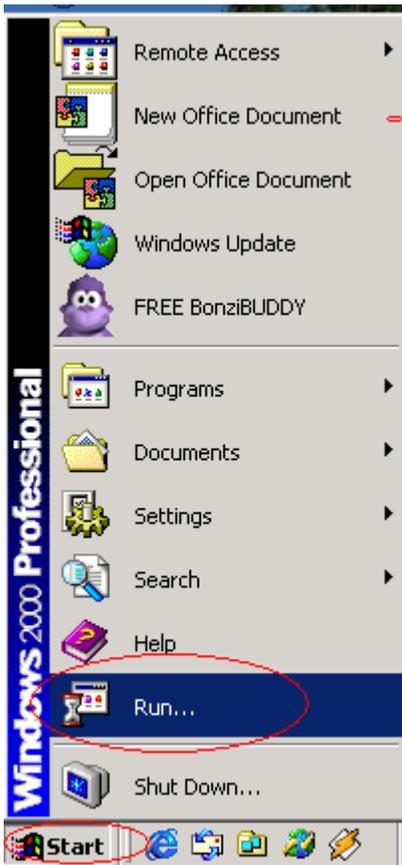
1. Browse for the NT4MSHF.exe file. Double click on this file.



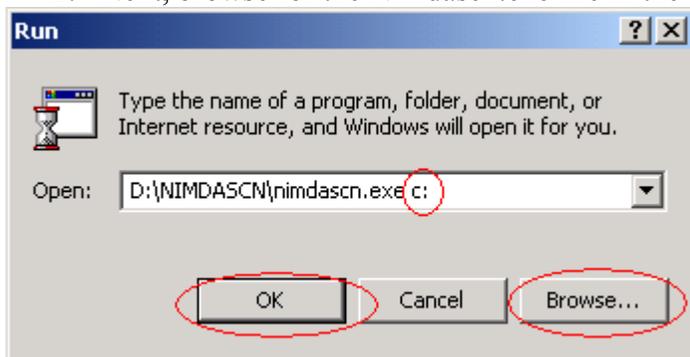
2. Restart the PC.

### Nimda patch.

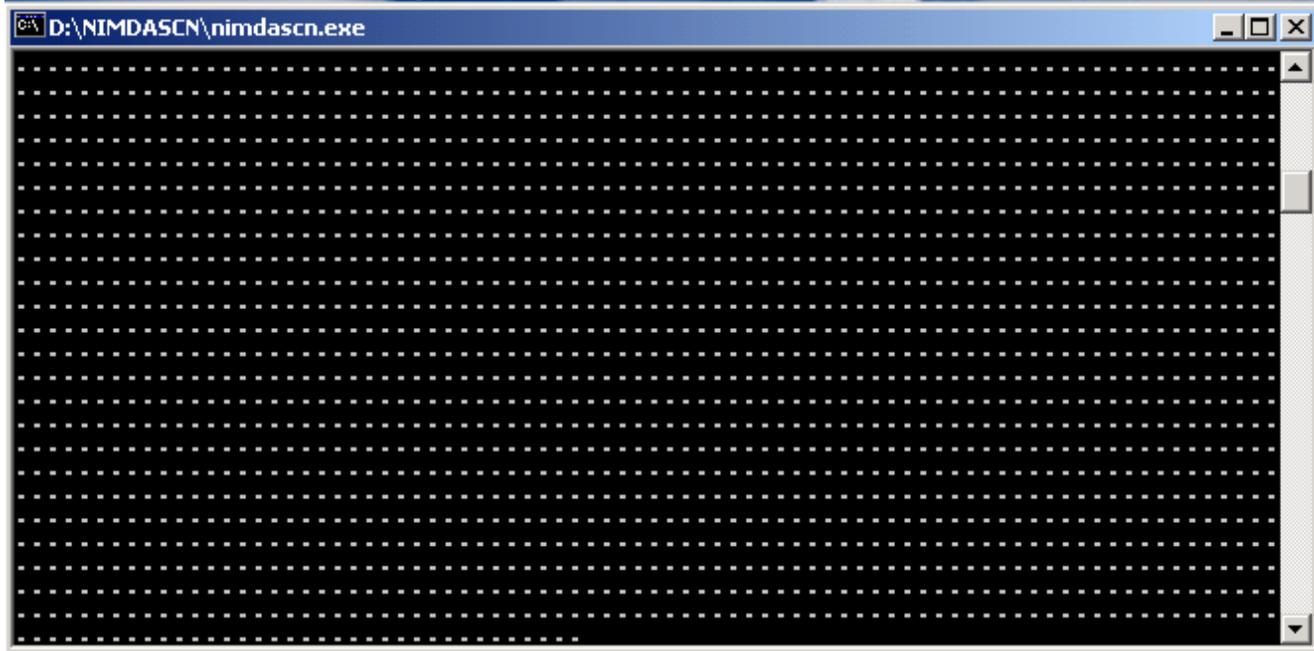
1. You must run the Nimdascn.exe file. For this, make click on Start button (at down left corner of the desktop) and choose Run...



2. Next, browse for the Nimdascn.exe file in the CD o shared drive and add "C:"



3. You should see the following window:



4. When the windows disappears the process is complete.

