

Informe Final

Indice

1	INTRODUCCIÓN	6
1.1	OBJETIVO	6
1.2	AUDIENCIA	6
1.3	ALCANCE	6
1.4	DEFINICIONES Y SIGLAS	6
1.5	AVANCES	7
1.6	REFERENCIAS	7
2	DESCRIPCIÓN GENERAL	9
2.1	ORGANIZACIÓN DE LA DOCUMENTACIÓN	9
2.2	CÓMO SE DEBE LEER EL INFORME FINAL	9
3	PRESENTACIÓN DEL PROYECTO	11
3.1	INTRODUCCIÓN	11
3.2	PRESENTACIÓN DEL PROBLEMA	12
3.3	INTEGRANTES	12
3.4	OBJETIVO	13
4	DESARROLLO DEL PROYECTO	14
4.1	ORGANIZACIÓN	14
4.2	ETAPAS	14
4.3	CRONOGRAMA	14
5	PRIMERA ETAPA – INVESTIGACIÓN	17
5.1	OBJETIVOS	17
5.1.1	ENTREGABLES	17
5.1.2	CRONOGRAMA	17
5.2	DESARROLLO	17
5.2.1	SISTEMAS OPERATIVOS Y WINDOWS NT	18
5.2.2	LA ARQUITECTURA DE WINDOWS NT	18
5.2.3	LA API DE WIN32	19
5.2.4	PROCESOS E HILOS	19
5.2.5	MEMORIA	20
5.2.6	SERVICIOS	21
5.2.7	PERFORMANCE EN SISTEMAS OPERATIVOS WINDOWS	21
5.2.8	PERFORMANCE UTILIZANDO SNMP	22
5.2.9	PERFORMANCE EN LOS CPU INTEL	22
5.3	RESULTADOS Y CONCLUSIONES	23

5.4	EVALUACIÓN	23
6	SEGUNDA ETAPA – PROTOTIPO	25
6.1	OBJETIVOS	25
6.1.1	ENTREGABLES	25
6.1.2	CRONOGRAMA	25
6.2	DESARROLLO	25
6.3	RESULTADOS	26
6.4	EVALUACIÓN	26
7	TERCERA ETAPA – DESARROLLO	27
7.1	OBJETIVOS	27
7.1.1	ENTREGABLES	27
7.1.2	CRONOGRAMA	27
7.2	DESARROLLO	28
7.2.1	ANÁLISIS DE REQUERIMIENTOS	28
7.2.2	ESPECIFICACIÓN DE DISEÑO	28
7.2.3	PROGRAMACIÓN	29
7.2.3.1	Módulo Servicio	29
7.2.3.2	Generador	32
7.2.4	MANUAL DE USUARIO	32
7.2.5	PLAN DE PRUEBAS	32
7.3	RESULTADOS Y EVALUACIÓN	33
8	CUARTA ETAPA – INFORME FINAL Y PRESENTACIÓN	34
8.1	OBJETIVOS	34
8.1.1	ENTREGABLES	34
8.1.2	CRONOGRAMA	34
8.2	DESARROLLO	34
8.2.1	INFORME FINAL	34
8.2.2	PÁGINA WEB	35
8.2.3	PRESENTACIÓN	35
8.3	RESULTADOS Y EVALUACIÓN	35
9	RESULTADOS Y CONCLUSIONES	36
9.1	RESULTADOS OBTENIDOS	36
9.2	PROBLEMAS ABIERTOS	36
9.3	EVALUACIÓN	37

Indice de Tablas

<i>Tabla 1: Cronograma Original</i>	14
<i>Tabla 2: Cronograma Real</i>	15
<i>Tabla 3: Clasificación de las distintas interfaces existentes según la cantidad de información que las mismas brindan para cada sistema operativo.</i>	22
<i>Tabla 4: Datos Interesantes para evaluar el proyecto</i>	37

Indice de Ilustraciones

<i>Ilustración 1: Duración de las etapas</i>	16
<i>Ilustración 2: Horas – hombre por integrante destinadas al proyecto</i>	16

1 Introducción

Este documento se presenta como un informe ejecutivo sobre el proyecto de Taller V “Medidor de Performance en Tiempo Real para Windows NT”.

En este capítulo se realizará una introducción al documento, dejando en claro su objetivo, audiencia y alcance. Además se incluirán definiciones y siglas que serán utilizadas a lo largo del mismo. Luego, se describe brevemente su contenido, y finalmente se incluyen las referencias a todos los documentos relacionados.

1.1 Objetivo

Este documento tiene como objetivo permitir al lector obtener una visión global del proyecto y servir de nexo entre los diferentes documentos obtenidos a lo largo del mismo, haciendo referencia a cada uno de ellos cuando corresponda. De este modo si se desea profundizar en algún tema en particular del proyecto, es posible recurrir al documento que trata dicho tema específicamente.

1.2 Audiencia

Este producto esta dirigido a aquellas personas interesadas en monitorear el sistema operativo Windows NT, los procesos que ejecutan en él, el CPU y/o los dispositivos accesibles a través de SNMP.

Debido a las características que presenta este sistema, el tipo de usuario al que está dirigido este producto es de carácter técnico con conocimientos de Windows NT, Redes y/o procesadores Intel Pentium. Sin embargo, al tener este documento un carácter introductorio, podrá ser comprendido por personas no expertas en computación.

1.3 Alcance

En este documento se puede encontrar una introducción al proyecto, relatando las distintas etapas del mismo.

Si bien la idea es lograr una visión general de todo el proyecto, no se pretende entrar en detalle en ninguno de los temas específicos.

1.4 Definiciones y siglas

ADO: Activex Data Object

API: Application Programming Interface

CPU: Central Process Unit

DBMS: DataBase Management System.

DLL: Dynamic Link Library

HAL: Hardware Abstraction Layer

MIB: Management Information Base

SCM: Service Control Manager

SCP: Service Control Program

SNMP: Simple Network Management Protocol.

1.5 Avances

El Informe Final se organiza en 9 capítulos, cuyo contenido se presenta a continuación.

En este capítulo se presenta el Informe Final sobre el proyecto de Taller V “Medidor de Performance en Tiempo Real para Windows NT”.

En el capítulo 2 (Descripción General) se describe brevemente el resto de la documentación del proyecto, así como también la forma en que se relaciona con el presente documento. Este capítulo resultará de gran utilidad para que se identifiquen los temas de interés del proyecto y se recurra a la documentación correspondiente, mientras se obtiene una visión general del mismo. Eso evitará que el lector deba leer la documentación en su totalidad para encontrar por su cuenta los tópicos de interés.

En el capítulo 3 (Presentación del Proyecto) se presenta el proyecto y se da una clara definición de los objetivos del mismo. Además se ubicará temporalmente y se informará sobre las personas involucradas en él.

En el capítulo 4 (Desarrollo del Proyecto) se describe brevemente la organización del grupo de trabajo durante el proyecto. Se comentan las distintas etapas del mismo, así como también se presenta el cronograma de trabajo.

En los capítulos 5, 6, 7, y 8 se presentan las diferentes etapas del proyecto: Investigación, Prototipo, Desarrollo e Informe Final y Presentación. En cada uno de ellos se detallan los objetivos de la etapa correspondiente, destacándose los entregables que la misma genera. Se comenta el cronograma original, así como también se describe el desarrollo del trabajo, incluyendo los detalles de mayor relevancia. Finalmente, se incluyen los resultados obtenidos y se realiza una evaluación general.

Por último, en el capítulo 9 (Resultados y Conclusiones) se informa sobre los resultados obtenidos. Luego, se comentarán los puntos que se considera importante tomar en cuenta para posibles extensiones del producto y finalmente se hace una evaluación del proyecto.

1.6 Referencias

- [ESTABLE4072]: Formulación del Proyecto “Modelado y construcción de una máquina paralela virtual basada en componentes de bajo costo” (Proyecto Conicyt Fondo “Clemente Estable” 4072) cuyos responsables son Héctor Cancela y Ariel Sabiguero.
- [PRW] “Estado del Arte de Windows NT” del proyecto de Taller 5 “Medidor de Performance en Tiempo Real para Windows NT” escrito por Alejandro Rettig, Aron Winokur y Martín Pedemonte.
- [PERFMON] Aplicación Performance Monitor (Monitor de Performance) incluida en Windows NT, cuyos fuentes se encuentran como ejemplo en el paquete MSDN.
- [TASKMAN] Aplicación Task Manager (Administrador de Tareas) incluida en Windows NT.

- [CPUMON] Aplicación CPU Monitor (Monitor de CPU) desarrollada por Mark Russinovich, Systems Internals, <http://www.sysinternals.com>
- [SNMPUTIL] Aplicación SNMP Utility (Utilitario de SNMP). Sus fuentes se encuentran a modo de ejemplo en el paquete MSDN.
- [Solomon99] "A fondo la arquitectura de Windows NT" David A. Solomon. Microsoft Press. McGraw-Hill (1999) ISBN: 84-481-2202-X.
- [Richter97] "Programación Avanzada en Windows" Jeffrey Richter. Microsoft Press. McGraw-Hill (1997) ISBN: 84-481-1160-5.
- [VC++] Manual de ayuda para Microsoft Visual C++. Este manual está incluido en el paquete MSDN.
- [CLASE] "Creating a simple Windows NT Service in C++" por Nigel Thompson del Microsoft Developer Network Technology Group (Noviembre 1995).
- [PLPR] "Plan de Pruebas" del proyecto de Taller 5 "Medidor de Performance en Tiempo Real para Windows NT" escrito por Alejandro Rettig, Aron Winokur y Martín Pedemonte.
- [MUSR] "Manual de Usuario" del proyecto de Taller 5 "Medidor de Performance en Tiempo Real para Windows NT" escrito por Alejandro Rettig, Aron Winokur y Martín Pedemonte.
- [ESDN] "Especificaciones de Diseño" del proyecto de Taller 5 "Medidor de Performance en Tiempo Real para Windows NT" escrito por Alejandro Rettig, Aron Winokur y Martín Pedemonte.
- [ANRQ] "Análisis de Requerimientos" del proyecto de Taller 5 "Medidor de Performance en Tiempo Real para Windows NT" escrito por Alejandro Rettig, Aron Winokur y Martín Pedemonte.
- [MSDN] MSDN Library.

2 Descripción General

Este capítulo describe brevemente el resto de la documentación del proyecto, así como también la forma en que se relaciona con el presente documento. Resultará de gran utilidad para que se identifiquen los temas de interés del proyecto y se recurra a la documentación correspondiente, mientras se obtiene una visión general del mismo. Eso evitará que el lector deba leer la documentación en su totalidad para encontrar por su cuenta los tópicos de interés.

2.1 Organización de la documentación

Se presentan en esta sección los documentos que son generados durante el proyecto. Se detallan a continuación en el mismo orden en que se encuentran en esta carpeta de documentación final del proyecto.

El primer documento que se presenta al lector es el Informe Final. Este documento tiene como objetivo permitir al lector obtener una visión global del proyecto y servir de nexo entre los diferentes documentos obtenidos a lo largo del mismo, haciendo referencia a cada uno de ellos cuando corresponda. De este modo si se desea profundizar en algún tema en particular del proyecto, es posible recurrir al documento que trata dicho tema específicamente.

A continuación se incluye “Estado del Arte de Windows NT” ([PRW]) en el cual se presentan los conceptos básicos del sistema operativo Windows NT, para lograr una familiarización mínima con el mismo. Dicha familiarización es fundamental para lograr un cabal entendimiento del problema ingenieril planteado, así como las limitaciones para solucionarlo. También puede constituirse en una guía rápida de las principales características de Windows NT para el lector con experiencia en el área de los sistemas operativos. Además se describe brevemente el protocolo SNMP, así como el funcionamiento de SNMP sobre Windows. Finalmente, se presenta el caso de monitoreo de performance en procesadores Intel.

Posteriormente se incluye el “Análisis de Requerimientos” ([ANRQ]). En dicho documento se establecen las características que tendrá el producto a ser desarrollado durante el proyecto. Busca servir de contexto para un entendimiento cabal del proyecto, sus alcances y limitaciones, tanto como una guía para las decisiones técnicas.

El siguiente documento es “Especificaciones de Diseño” ([ESDN]). En él, se establecen las bases para la etapa de implementación y verificación del sistema y servirá de guía a la hora de desarrollarla. Su cuidadosa revisión ayudará a reducir los esfuerzos en la etapa de desarrollo, facilitará la detección de inconsistencias y errores que se hayan producido en la etapa de análisis de requerimientos y en el proceso de verificación del sistema.

A continuación se incluye el “Plan de Pruebas” ([PLPR]). Dicho documento servirá de guía para los encargados de realizar las verificaciones. Con él se busca cubrir un conjunto mínimo de pruebas que permitan descubrir los principales errores que eventualmente contenga el programa. Los resultados de las pruebas servirán para realizar correcciones sobre el producto.

El último documento es el “Manual de Usuario” ([MUSR]). Dicho documento servirá como guía a los usuarios, ya sea para aprender a utilizar la aplicación, o como fuente de consulta para el usuario que ya conoce como utilizarla.

2.2 Cómo se debe leer el Informe Final

El presente informe es ejecutivo, por lo tanto no pretende entrar en detalle con respecto a los distintos tópicos que trata. Los detalles se encuentran en el resto de los documentos descriptos en la sección anterior, y se debe recurrir a ellos cuando se busque profundizar en el conocimiento de un determinado tema.

Se da a continuación una breve guía de que documentos resultan recomendables leer en forma conjunta a los distintos capítulos.

En los capítulos 5 y 6, se recomienda profundizar leyendo el documento [PRW].

En el capítulo 7, se recomienda leer los documentos [PRW], [ANRQ], [ESDN], [PLPR] y [MUSR] para obtener mayor información en los temas allí tratados.

3 Presentación del Proyecto

Por intermedio del presente capítulo, se introduce al lector en el proyecto y se busca que obtenga una familiarización general con el mismo. Se ubica al proyecto temporalmente, se da el contexto del mismo, se informa sobre las personas que estuvieron involucradas en él y finalmente se definen claramente sus objetivos.

3.1 Introducción

El presente proyecto se enmarca dentro de la asignatura Taller V, que puede considerarse como el proyecto final para la carrera de Ingeniería en Computación. Dicho proyecto comenzó en Abril de 1999 y su plazo máximo de finalización es Marzo de 2000.

El mismo se encuentra inmerso en un proyecto mayor llamado: "Modelado y construcción de una máquina paralela virtual basada en componentes de bajo costo" (Proyecto Conicyt Fondo "Clemente Estable" 4072) cuyos responsables son Héctor Cancela y Ariel Sabiguero. Se extracta a continuación un pasaje del informe [ESTABLE4072]:

"En diversas ramas de los sectores científico, productivo y de servicios existen problemas (planificación de la producción, asignación de recursos, diseño de equipos, cálculo en ingeniería, modelos climatológicos y de impacto ambiental, etc.) cuya resolución requiere una gran potencia de cálculo. Ante esa demanda, la respuesta clásica ha sido la utilización de computadoras de gran porte, sumamente costosas tanto en adquisición como en mantenimiento.

Las redes de PC son una solución tradicionalmente adoptada en ambientes administrativos. Los sistemas son fácilmente escalables con equipos de múltiples proveedores e incluso con diferentes sistemas operativos. Es usual que incluso empresas pequeñas cuenten con ellas, por lo que ya están familiarizados con aspectos de su administración, operación, etc. Esas empresas no podrían financiar/justificar inversiones en estaciones de trabajo y/o sistemas operativos más sofisticados, pero sí tienen necesidades de cálculo intensivo a satisfacer.

Esas máquinas conectadas en red pueden usarse como si fuesen una única máquina virtual. Así se alcanzan potencias de cálculo comparables a las obtenidas por supercomputadores clásicos; sin embargo, hay preguntas vinculadas a aspectos no analizados en la literatura como ser: dada una lista de precios, es más conveniente mejorar la velocidad de comunicaciones, o agregar una CPU adicional, o mejorar la velocidad de los procesadores disponibles? Para contestarlas, se requiere un modelo que describa las capacidades de un equipo antes de construirlo.

El objetivo de este proyecto apunta al aprovechamiento de la potencia de cálculo ociosa de los equipos PC utilizados en aplicaciones de oficina, mediante el desarrollo de una "máquina paralela virtual" capaz de ejecutar las aplicaciones de cálculo intensivo. Para ello, se plantea analizar y resolver dos aspectos: por un lado el desarrollo de un modelo que permita estimar su potencia de cálculo, dimensionando así la configuración necesaria para cada aplicación en particular y por otro la construcción de la "máquina paralela virtual", que implica el estudio y la adaptación de diversos componentes de hardware y de software para el funcionamiento de la misma."

Es en este contexto que resulta necesaria una herramienta de monitoreo capaz de extraer en tiempo real información sobre las estaciones de trabajo y la red para permitir una evaluación y/o ajuste del modelo que se proponga.

3.2 Presentación del problema

Windows NT es hoy en día uno de los sistemas operativos más comunes para redes en el mercado, sin embargo su inserción en el ámbito académico ha sido reciente. Es por esto, que resulta un terreno nuevo para la investigación.

En este marco, se plantea la necesidad de construir una herramienta capaz de obtener la información de performance disponible en un sistema Windows NT.

Ya avanzado el proyecto se plantea la posibilidad de extender las funcionalidades de la herramienta para permitir una mayor potencia a la hora de monitorear una computadora, o incluso una red. Para lograr esto se estudiará la factibilidad de medir los Contadores de Performance de los procesadores Pentium de Intel y obtener información sobre todos los dispositivos existentes en una red a través de SNMP.

Actualmente existen herramientas que permiten monitorear el sistema operativo Windows NT. Son buenos ejemplos de ello: el Performance Monitor¹ y el Task Manager². Ambas herramientas, permiten monitorear una gran diversidad de contadores; en el caso del PerfMon también permite elegir sobre que objetos del sistema se realizarán las mediciones, mientras que en el Task Manager solo se permite realizar mediciones sobre los procesos.

En cuanto al estudio de los procesadores Pentium existen también varias herramientas que permiten monitorear dichos contadores; un ejemplo es CpuMon³ el cual permite medir los diferentes contadores de performance de un procesador Pentium en forma interactiva, mostrando los resultados en pantalla.

En lo que respecta a SNMP, una aplicación que se acerca a los requerimientos planteados es SnpUtil⁴, la cual permite realizar consultas a través de SNMP desplegando los resultados obtenidos en pantalla.

Sin embargo todas estas herramientas cuentan con importantes defectos para realizar la tarea requerida. Es por tanto que surge la necesidad de desarrollar una nueva herramienta que tenga las siguientes características:

- Almacenamiento de la información en una base de datos. Esto permite guardar información desde distintas máquinas a la vez y que la misma quede en un único sitio permitiendo consultar dicha información en tiempo real, de forma concurrente.
- Flexibilidad para seleccionar los datos a medir de diversas formas.
- Permitir realizar mediciones aunque ningún usuario haya iniciado sesión en el sistema operativo.
- Poder incluir mediciones de datos que no se encuentren disponibles en el momento de la configuración.
- Combinar todas las funcionalidades en una única aplicación.

El objetivo principal de este proyecto es el de realizar una herramienta de esas características.

3.3 Integrantes

El grupo encargado de desarrollar el proyecto esta formado por 3 integrantes: Alejandro Rettig, Arón Winokur y Martín Pedemonte. Además de los mencionados integrantes, Héctor Cancela y Ariel Sabiguero se encargan de las tareas de tutoría.

¹ La versión en español de este producto se llama Monitor de Performance y esta descrito en la referencia [PERFMON]. De aquí en más se utilizará PerfMon para referenciarlo.

² La versión en español de este producto se llama Administrador de Tareas y esta descrito en la referencia [TASKMAN].

³ El producto CpuMon esta descrito en la referencia [CPUMON].

⁴ El producto SnpUtil esta descrito en la referencia [SNMPUTIL].

3.4 Objetivo

El objetivo de este proyecto es el de obtener una herramienta capaz de realizar mediciones en tiempo real sobre distintos objetos del sistema operativo Windows NT. Dicha información deberá ser almacenada en una base de datos para permitir en instancias posteriores su utilización. La herramienta deberá funcionar aunque ningún usuario haya iniciado sesión en el sistema operativo.

Debe ser posible monitorear los contadores de performance que brinda el sistema operativo Windows NT. Además se debe estudiar la posibilidad de extender la herramienta permitiendo realizar mediciones a través de SNMP y sobre los contadores que brindan los procesadores Pentium.

Se estudia también, la posibilidad de que la aplicación sea compatible con los sistemas operativos Windows 95, 98 y 2000. El caso que resulta más interesante de los mencionados es el de Windows 2000, ya que será el sistema que seguramente reemplazará a las otras versiones de Windows.

4 Desarrollo del Proyecto

En este capítulo se describe brevemente la organización del grupo de trabajo durante el proyecto. Se comentan las distintas etapas del mismo, y se presenta el cronograma de trabajo.

4.1 Organización

El grupo no tuvo una separación fija de roles, sino que para cada etapa hubo un reparto de tareas particular. En general se intentó separar las tareas tomando en cuenta como referencia la afinidad y conocimiento de cada integrante con las mismas. Si bien en un principio fue sencillo dividir las tareas, en las etapas posteriores fue necesario trabajar constantemente reunidos.

En todo momento hubo un apoyo mutuo de los integrantes para supervisar las tareas realizadas por los demás y así lograr un producto de mayor calidad.

El grupo se reunió una vez por semana con los tutores durante las primeras tres cuartas partes del proyecto y posteriormente se pasó a un régimen de una reunión cada quince días. En el funcionamiento interno se utilizó una reunión de coordinación por semana. Dichas reuniones sirvieron para realizar el seguimiento del mismo y para dividir y ajustar las tareas a realizar en la siguiente semana.

4.2 Etapas

Se identificaron 4 etapas en el proyecto, las cuales se presentan a continuación y se detallan en profundidad en los sucesivos capítulos.

La primera etapa corresponde a la Investigación, en ella se pretende lograr una familiarización con el sistema operativo Windows NT.

Posteriormente, se realiza un Prototipo para lograr una primera aproximación a la aplicación final, que permita evaluar la factibilidad de lograr los objetivos planteados.

La tercera etapa corresponde al Desarrollo, el cual implica la construcción de la aplicación.

Finalmente, en la etapa Informe Final y Presentación, se redacta el Informe Final que resume el desarrollo del proyecto, se diseña una página web para la difusión del mismo y se prepara la presentación.

Cabe aclarar que las diferentes etapas pueden coexistir en el tiempo, como se verá a continuación.

4.3 Cronograma

Se detalla a continuación el cronograma original del proyecto, donde aparece la cantidad de horas estimadas por integrante por semana a las distintas etapas.

Etapas del Proyecto	Meses							
	Abril '99	Mayo '99	Junio '99	Julio '99	Ago. '99	Sept. '99	Oct. '99	Nov. '99
Investigación	X15	X15						
Prototipo			X15	X15				
Desarrollo					X15	X15	X15	
Informe Final								X15

Tabla 1: Cronograma Original

En dicho cronograma se consideraba necesario dedicar 15 horas por integrante por semana. O sea, 45 horas-hombre por semana.

Durante el transcurso del proyecto, fue necesario introducir cambios en el calendario.

En primer lugar se aumenta la duración de la primera etapa, ya que resultó insuficiente el tiempo destinado a esta tarea al comenzar el proyecto. Además fue necesario investigar nuevos temas luego de comenzada la etapa de Desarrollo debido a la introducción de requerimientos adicionales.

Se reduce la duración de la segunda etapa, ya que en el cronograma original se pensaba en un prototipo que fuera una versión avanzada de la aplicación, mientras que en el segundo enfoque se prefirió un prototipo pequeño para poder corroborar rápidamente que los progresos teóricos obtenidos en la etapa de investigación eran aplicables en la práctica.

Se aumenta la duración de la etapa de desarrollo debido al cambio de enfoque de la etapa anterior y a que la herramienta fue más ambiciosa que la originalmente planteada.

Finalmente, se aumenta la duración de la última etapa ya que se consideró insuficiente el tiempo originalmente destinado a tales efectos.

Se detalla a continuación el cronograma seguido durante el proyecto, donde aparece la cantidad de horas destinadas por integrante por semana a las distintas etapas.

Etapas del Proyecto	Meses									
	Abril '99	Mayo '99	Junio '99	Julio '99	Agosto '99	Sep. '99	Oct. '99	Nov. '99	Dic. '99	Ene-Feb '99
Investigación	X20	X20	X20	X5	X5	X5				
Prototipo				X5						
Desarrollo					X5	X10	X15	X20	X20	
Informe Final								X10	X10	X15

Tabla 2: Cronograma Real

La cantidad de horas-hombres reales fue estimada en forma mensual al terminar cada mes. Se optó por esta práctica y no por una más formal, ya que técnicas tales como completar registros de actividad resultaron poco prácticas. Además, únicamente se busca lograr una idea general de las horas destinadas al proyecto en cada etapa, y como se distribuyen las mismas a lo largo de los meses.

Se aprecia en la gráfica de la Ilustración 1, que en general las etapas tuvieron una mayor duración de la planeada originalmente. La única excepción es la etapa Prototipo en la cual hubo un cambio de objetivo, como se explicó anteriormente. Un hecho significativo es que la etapa de Investigación tuvo una duración tres veces mayor de la planeada originalmente.

Se presenta la gráfica de horas-hombre por integrante destinadas al proyecto (Ilustración 2), en la cual se vuelve a apreciar la mayor duración de las tareas reales con respecto a las planeadas originalmente, manteniéndose la excepción en la etapa del prototipo. También se puede apreciar que originalmente se consideró que la tarea de mayor duración sería el desarrollo, a la que seguiría prototipo, posteriormente investigación y finalmente informe final. En el funcionamiento real, las etapas de desarrollo e investigación tuvieron la misma carga horaria aproximadamente, siguiendo el informe final y finalmente el prototipo.

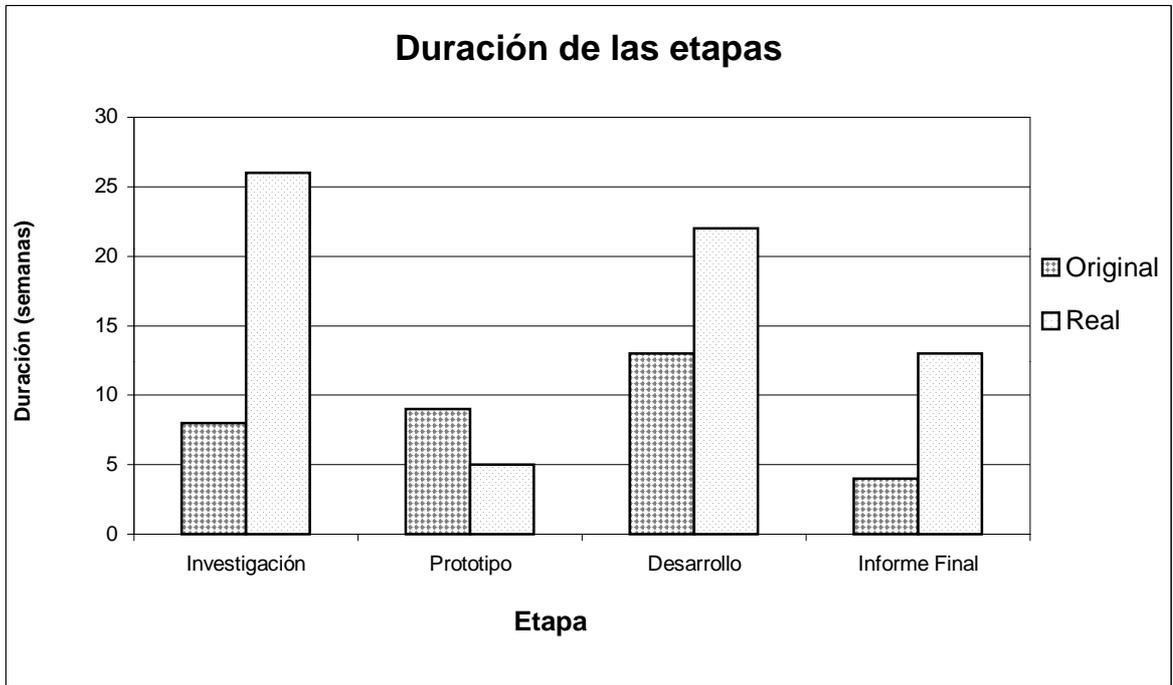


Ilustración 1: Duración de las etapas

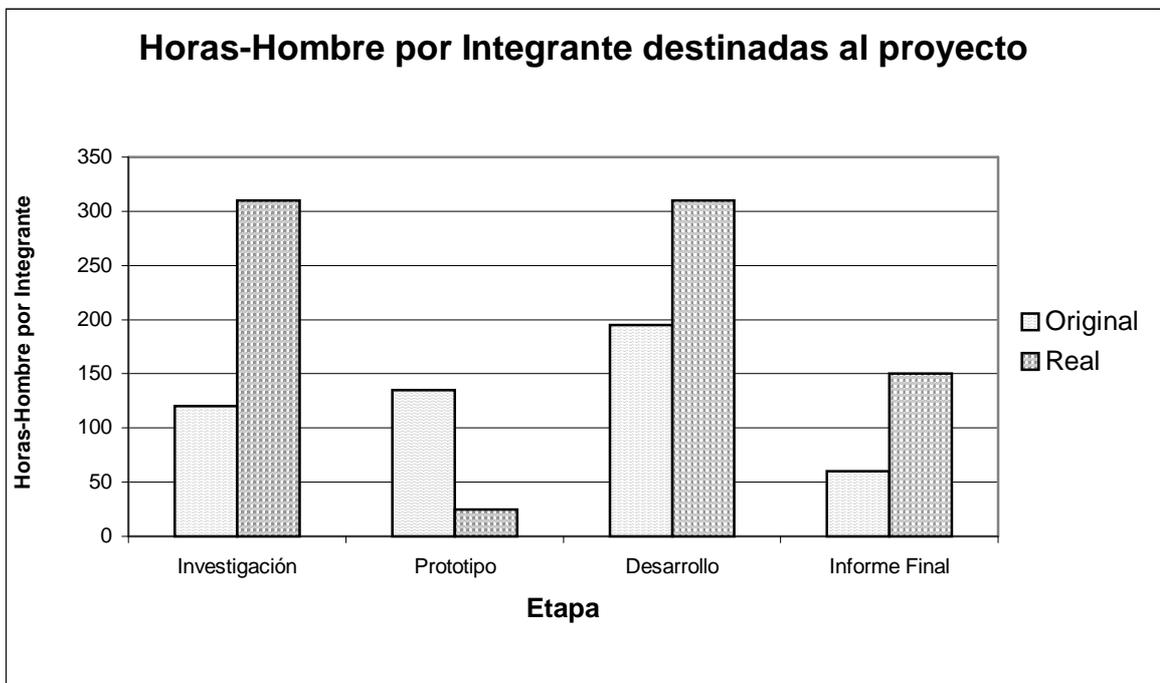


Ilustración 2: Horas – hombre por integrante destinadas al proyecto

5 Primera Etapa – Investigación

Se describe a continuación la primera etapa correspondiente a la Investigación. Se detallan sus objetivos y los entregables que la misma genera. Luego, se comenta el cronograma y se presenta el desarrollo donde se describen con todo detalle las situaciones originadas, así como los conocimientos obtenidos que requieran ser destacados. Finalmente, se comentan los resultados y se realiza la evaluación de la etapa.

5.1 Objetivos

El objetivo fundamental de la presente etapa es obtener los conocimientos necesarios sobre Windows NT, para poder desarrollar una herramienta con las características descritas en la sección 3.4. Además se busca lograr una base conceptual sobre el tema performance y el problema de medición de performance.

Existen dos objetivos secundarios que son investigar la forma de extender la aplicación para permitir realizar mediciones sobre SNMP y sobre los contadores de performance de los procesadores Pentium. Se debe realizar un estudio sobre la factibilidad de implementar las mencionadas funcionalidades.

5.1.1 Entregables

Se espera generar en la presente etapa un documento que brinde una breve introducción al sistema operativo Windows NT. Dicho documento, debe ser capaz de sumergir al lector en el complejo mundo de la implementación de este sistema operativo y no deberá escatimar detalles, siendo fundamental que cuente con una amplia bibliografía a la cual el lector inquieto podrá recurrir para profundizar en los diferentes tópicos.

Adicionalmente, se incluirá información sobre los diferentes temas que resulten de interés para realizar las mediciones sobre SNMP y los contadores de performance de los procesadores Pentium.

Se identifica a dicho documento con el nombre “Estado del Arte de Windows NT” ([PRW]).

5.1.2 Cronograma

La presente etapa se extiende durante seis meses, comenzando en Abril de 1999. En los primeros tres meses se trabajó únicamente en los temas concernientes a esta etapa, pero en los meses restantes el trabajo se vio solapado con el de otras etapas.

5.2 Desarrollo

En un comienzo se presentaron grandes dificultades, sólo uno de los integrantes del grupo estaba familiarizado con el sistema Windows NT como usuario; en cambio los otros dos nunca habían siquiera utilizado esta plataforma. Es por tanto, que se puso especial hincapié en realizar una extensa investigación, y así lograr los conocimientos necesarios para desarrollar el proyecto. Se consideró que realizar un gran esfuerzo en esta etapa iba a facilitar las siguientes etapas del proyecto.

En principio no se conocía una buena fuente de información excepto Internet. Se pensó que algún libro podría ser de ayuda por lo que se consiguió el catálogo de libros publicados por la Microsoft Press. Se encontraron 2 libros de interés que finalmente fueron adquiridos y que se encuentran en las referencias [Solomon99] y [Richter97].

En un primer momento, los progresos eran escasos y el material que se encontraba aportaba muy poco sobre los detalles íntimos del sistema operativo. En general, la mayoría de la documentación encontrada estaba pensada para el nivel de usuario y no para el de implementador de programas sobre el sistema operativo. Resultó fundamental el aporte dado por la MSDN Library [MSDN], que si bien no brinda la información en forma de manual y hay que hurgar entre miles de artículos para encontrar lo que se quiere exactamente, aportó información imprescindible.

Como se mencionó anteriormente, era necesario obtener información para comenzar con el desarrollo del producto. Pero tal vez más importante que encontrar dicha información, era plantear las preguntas correctas, las cuales permitirían avanzar para lograr el objetivo. Es por tanto, que se comenzó con la elaboración del documento de Estado del Arte [PRW] el cual se pensó que debía cubrir los siguientes puntos: Arquitectura de Windows NT, la API de Win32, Procesos e Hilos, Memoria y un capítulo dedicado a comentar las distintas formas de medir performance en sistemas Windows⁵.

Cuando se comenzó a delinear el documento se consideró una buena idea el incluir un primer capítulo que enmarcara al sistema operativo Windows NT dentro del mundo de los sistemas operativos y describiera brevemente sus principales características.

La lectura de documentos y artículos fue lentamente arrojando luz sobre la investigación y así fue que se consideró necesario el incluir un capítulo sobre Servicios. Luego a medida que fueron creciendo los requerimientos, se decide agregar dos nuevos capítulos: uno sobre SNMP y otro sobre los contadores de performance en los CPU Intel.

A continuación, se detalla una subsección por capítulo del [PRW] para explicar en forma sucinta los conceptos importantes a recalcar de cada uno de ellos.

5.2.1 Sistemas Operativos y Windows NT

En el presente capítulo se busca introducir la familia de sistemas operativos Windows NT y enmarcarla dentro de los distintos tipos de sistemas operativos existentes.

Actualmente, se encuentran disponibles las versiones 4.0 de Windows NT Server y de Windows NT Workstation, y se espera para marzo de 2000 la nueva versión de Windows. Dicha versión se llamará Windows 2000 y unificará la familia Windows 9x con la familia Windows NT. Una característica a remarcar es que se incluirá la Win64 API, resultando así el primer sistema operativo de Microsoft de 64 bits.

5.2.2 La Arquitectura de Windows NT

Windows NT es un sistema operativo híbrido, resultando de la mezcla de uno monolítico con uno cliente/servidor. Los componentes del sistema sensibles al rendimiento se ejecutan en modo kernel, donde pueden interactuar con el hardware y entre sí, sin sufrir la sobrecarga de cambios de contexto y transiciones entre modos, mientras que las aplicaciones de usuario no acceden al núcleo del sistema, sino que interactúan con él mediante el subsistema Win32 en modo usuario.

Se comentan brevemente los módulos más importantes del núcleo de Windows NT :

- El Ejecutivo (NT Executive) - Contiene los servicios básicos del Sistema Operativo, Administración de Memoria, Procesos, Hilos, seguridad, E/S y comunicación entre Procesos.

⁵ Este capítulo toma el nombre de "Performance en Sistemas Operativos Windows".

- El Kernel (NT Kernel) – Es el microkernel o parte del núcleo que describe los objetos más básicos del sistema, realiza las funciones de bajo nivel del Sistema Operativo, Planificación de Hilos, distribución de Interrupciones y excepciones y la sincronización multiprocesador. Aparte tiene un conjunto de rutinas y objetos básicos que el ejecutivo usa.
- El Nivel de Abstracción al Hardware (Hardware Abstraction Layer, HAL) - Aísla el kernel, los controladores de dispositivos y el ejecutivo de las diferencias de Hardware específicas de cada plataforma.
- Administrador de Objetos (Object Manager) – Es el responsable de crear y eliminar objetos del ejecutivo, así como de gestionar la lista de recursos utilizada por el sistema operativo.
- Administrador de Procesos (Process Manager) – Permite crear y destruir procesos y hilos, así como controlar el estado de los mismos, suspendiéndolos y reanudándolos según se precise.
- El Administrador de Memoria Virtual (Virtual Memory Manager)– Es el gestor responsable de proporcionar memoria física y memoria virtual paginable a los diferentes procesos. Permite que las aplicaciones dispongan de espacios de memoria planos y separados, de manera que el comportamiento de una aplicación no pueda afectar a las demás.

5.2.3 La API de Win32

Existen varios subsistemas en Windows NT sobre los cuales pueden ejecutarse aplicaciones, se comentará el principal: la API de Win32.

La API de Win32, al igual que cualquier otra API, es simplemente un conjunto de funciones que pueden ser invocadas al escribir el código fuente de un programa.

La API de Win32 facilita la tarea de realizar aplicaciones de 32 bits para todos los sistemas Windows, permitiendo hacer uso de diversas funcionalidades. Dentro de los grupos de funcionalidades se destacan Administración de ventanas (proporciona medios para crear y administrar la interfaz de la aplicación con el usuario), Interfaz de manejo de Gráficos (permite la generación de una salida gráfica para una aplicación), Servicios del Sistema (se encargan de acceder a los recursos de la computadora y a las posibilidades brindadas por el sistema operativo), Multimedia (manipulan audio y video) y Llamado a Procedimientos Remotos (para ejecutar procesos en máquinas remotas).

Es importante remarcar que mediante la API de Win32 se buscó:

- Proveer a los programadores con una herramienta para diseñar aplicaciones que funcionen en todas las plataformas Windows, evitando así tener que mantener múltiples versiones.
- Contar con una gran variedad de software que funcione correctamente para cada nuevo sistema operativo que Microsoft lance al mercado, apenas el mismo aparezca. En el caso ideal, el usuario final podría realizar una actualización de su sistema operativo, y continuar utilizando los mismos ejecutables que utilizaba anteriormente.

5.2.4 Procesos e Hilos

En Windows NT los procesos son manejados por el Administrador de Procesos. Un proceso está compuesto por un programa, un conjunto de recursos necesario para su ejecución, un espacio de memoria virtual reservado, y por lo menos un hilo de ejecución que será planificado por el kernel.

Windows NT maneja dos modos de ejecución para sus procesos, modo kernel y modo usuario. Todos los procesos en general se ejecutan en modo usuario, pero en el momento que necesiten algún servicio del sistema operativo, el procesador cambia su modo de ejecución a modo kernel mientras se ejecuta el servicio, regresando luego a modo usuario cuando termine.

Es más fácil resolver un problema si lo partimos en problemas más chicos, un hilo sería cada una de las subtareas que debe ejecutar un proceso, un proceso termina la ejecución cuando todos sus hilos terminan de ejecutarse.

Cada hilo se ejecuta en un espacio virtual separado para cada proceso, cuando un proceso tiene varios hilos, todos ellos se ejecutan en el mismo espacio virtual, permitiendo así que los resultados de un hilo, sirvan como entradas de procesamiento de otros.

Windows NT implementa un sistema de planificación basado en prioridades con interrupción de tareas, de manera que siempre se ejecuta el hilo preparado para la ejecución con mayor prioridad.

Cuando se selecciona un hilo para su ejecución, este se ejecuta durante un cierto tiempo llamado quantum. Un quantum es el tiempo que se le permite ejecutarse a un hilo antes de que Windows NT lo interrumpa para buscar otro del mismo nivel de prioridad, que ya este esperando para ejecutarse, o para reducir la prioridad del hilo cuando sea necesario. Los valores del quantum pueden variar entre hilos (y son distintos entre las versiones de Windows NT Server y Workstation). Puede pasar que un hilo no llegue a consumir su quantum. Como Windows NT implementa un planificador con interrupciones, si en un momento dado queda preparado para su ejecución otro hilo de mayor prioridad, el hilo actual en ejecución se interrumpe antes de que termine su tiempo. El código del planificador de Windows NT se encuentra en el kernel.

Windows NT planifica con granularidad de hilo, porque los procesos no se ejecutan sino que proporcionan los recursos y el contexto en el que se ejecutan los hilos.

5.2.5 Memoria

El Administrador de Memoria proporciona un conjunto de servicios del sistema que permite a los subsistemas de entorno reservar y liberar memoria virtual, compartir memoria entre procesos, cargar archivos en memoria, enviar las páginas virtuales al disco, recoger información acerca de rangos de páginas virtuales, cambiar su protección y bloquearlas en memoria. Dichos servicios suministran al solicitante un manejador de proceso que le permite hacer referencia al proceso particular cuya memoria se va a manipular. El llamante puede así manipular su propia memoria o (con los permisos adecuados) la de otros procesos. Por ejemplo, el proceso padre puede reservar, liberar, leer y escribir la memoria del proceso hijo si llama a los servicios de memoria virtual y pasa como argumento el manejador del proceso hijo.

La mayoría de estos servicios se exportan por medio de la API de Win32. La API de Win32 administra la memoria de las aplicaciones mediante tres grupos de funciones:

- Funciones de memoria virtual con granularidad de página (Virtualxxx)
- Funciones de archivos en memoria (CreateFileMapping,MapViewof File)
- Funciones del heap (Heapxxx y las interfaces antiguas Localxxx y Globalxxx)

El administrador de memoria también proporciona un cierto número de servicios para los controladores de dispositivos, como la reserva y liberación de memoria física y el bloqueo de páginas en memoria física para las transferencias de acceso directo a memoria (DMA).

5.2.6 Servicios

El sistema operativo Windows NT soporta distintos tipos de aplicaciones. Un tipo que resulta de particular interés son los servicios. Los servicios son programas que pueden ser ejecutados por el sistema operativo en forma automática al iniciar el sistema. Los servicios resultan de gran interés ya que pueden ser ejecutados sin que ningún usuario haya iniciado sesión en el sistema y por lo tanto, servirían para cumplir con los requerimientos planteados⁶.

El funcionamiento de los servicios se explica mediante tres componentes fundamentales: el Service Control Manager (SCM), el servicio y el Service Control Program (SCP).

El SCM comienza su ejecución en forma automática cuando el sistema operativo bootea, y finaliza su ejecución cuando el sistema es apagado. El SCM es responsable de comunicarse con los distintos servicios y por esto tiene privilegios de sistema. El sistema mantiene una base de datos de los servicios instalados a través del SCM. La API de Win32 provee de un conjunto de funciones que pueden ser usadas para extraer información asociada al SCM.

Un servicio debe contener tres funciones esenciales para poder interactuar con el SCM. La primera de esas funciones es *Main* que puede verse como el punto de entrada (entry point) de una aplicación del tipo consola (aplicaciones sin interfaz gráfica). El hilo principal ejecuta esta función, que inicializa a todos los servicios del proceso. Cuando todos los servicios del proceso han terminado su ejecución, el hilo principal es quien realiza las operaciones necesarias para que el proceso termine su ejecución en forma correcta. La segunda función es la función *ServiceMain*. Dicha función es invocada por el sistema operativo y ejecuta el código que implementa al servicio propiamente dicho. Se utiliza un hilo dedicado exclusivamente a ejecutar cada función *ServiceMain* de un servicio. La tercera y última función es *Handler*. El SCM invoca la función *Handler* de un servicio para provocar cambios en su estado. El hilo principal es el encargado de ejecutar todas las funciones *Handler*.

El SCP es una aplicación que tiene interfaz con el usuario y que le permite interactuar con el servicio, para realizar tareas tales como iniciarlo y detenerlo. Es por tanto, que debe interactuar con el SCM.

5.2.7 Performance en Sistemas Operativos Windows

Para realizar un estudio de performance es necesario seleccionar inicialmente cuales son los parámetros de interés para el estudio. Dichos parámetros deben ser elegidos de tal modo que sean característicos del estudio que se está realizando, y que a través del análisis de sus valores a lo largo del tiempo sea posible obtener la información buscada.

En el caso de un sistema operativo, existen muchos parámetros de estudio posibles. Los mismos van desde los clásicos estudios de tiempo de CPU o memoria utilizados hasta estudios particulares acerca de un protocolo de red específico. Aunque puede ser relativamente fácil obtener algunos de estos parámetros, resulta prácticamente imposible obtener algunos otros.

Si bien en teoría es posible obtener datos acerca de los parámetros que se desee, está tarea es realmente muy complicada y puede requerir re-programar y/o re-diseñar partes de sistemas operativos para tal propósito. En la práctica, los parámetros que son estudiados son aquellos que el diseñador del sistema operativo decidió registrar, y brinda una interfaz de acceso a los mismos.

En los sistemas operativos Windows 95, Windows 98, Windows NT 4 y Windows 2000 existen varias formas de acceder a los datos de performance. Debido a que el tipo de información al que se desea acceder

⁶ Ver la sección 3.4 “Objetivo” del presente documento.

es de muy bajo nivel para el sistema operativo, es natural que las estructuras de datos donde dicha información está almacenada sean diferentes en cada sistema Windows. Incluso, algunos datos de performance existen sólo para algunos sistemas operativos, ya que las entidades que reflejan no tienen sentido en los demás.

Es por esto, que resulta un tanto difícil encontrar un método genérico para obtener los datos de interés en cualquier sistema Windows. A continuación se presentan las diferentes interfaces existentes que permiten acceder a datos de performance, describiendo en cada caso su compatibilidad entre sistemas y nivel de información.

	Windows 95	Windows 98	Windows NT 4	Windows 2000
Registro	Alto	Alto	Alto	Alto
NTDLL	Alto	Alto	Alto	Alto
PDH	No soportado	No soportado	Medio	Medio
PSAPI	No soportado	No soportado	Medio	Medio
ToolHelp32 API	Medio	Medio	No soportado	Medio
API de Win32	Bajo	Bajo	Bajo	Bajo

Tabla 3: Clasificación de las distintas interfaces existentes según la cantidad de información que las mismas brindan para cada sistema operativo.

5.2.8 Performance utilizando SNMP

SNMP es un protocolo de administración orientado a las redes, utilizado en redes TCP/IP para monitorear y administrar gran diversidad de software y hardware conectado a la misma.

Para comprender mejor la estructura y funcionamiento de este protocolo, se mencionan a continuación sus cuatro partes más importantes.

- **Management Information Base (MIB):** es una base de datos que contiene la descripción e identificación de todos los objetos que pueden ser administrados.
- **Identificación de Objetos (OID):** cada objeto es identificado a través de un único identificador.
- **Programas Agente:** programas instalados en cada dispositivo que pueden ser monitoreados a través de SNMP.
- **Programas Administradores:** son utilizados por los administradores de la red para obtener la información.

Escribir un programa de administración de SNMP, que corra en Windows NT es una tarea relativamente sencilla. Basta con comprender la interfaz brindada por el sistema operativo y hacer un correcto uso de la misma.

5.2.9 Performance en los CPU Intel

La información obtenible es sumamente dependiente de la arquitectura del procesador, su fabricante, e incluso su modelo. Es este el motivo por el cual para monitorear un CPU no se utiliza una herramienta genérica que funcione en todos los CPU (de más está decir que no existe), sino que se aborda cada procesador por separado, y se construye una solución para cada caso particular.

Dentro de los procesadores de Intel, existe una gran variedad de modelos. Lamentablemente, cada uno de estos modelos tiene características diferentes, y brinda interfaces distintas para obtener información de

performance. Es por esto que, dentro de los procesadores, se estudiarán únicamente los procesadores más comunes del mercado al momento de escribir este informe: Pentium, Pentium MMX, Pentium Pro, Pentium Celeron, Pentium II y Pentium III.

Evidentemente, escapa completamente al espíritu de este capítulo abordar las diferencias o características tecnológicas que presenta cada uno de estos modelos, únicamente se pretende estudiar la información de performance que está disponible y la interfaz para obtenerla.

En realidad en cuanto a este estudio se refiere, únicamente interesará determinar la familia y el modelo del procesador, de esta forma se definen 3 grupos:

- Familia P5: Pentium y Pentium MMX
- Familia P6: Pentium Pro, Pentium Celeron, Pentium II y Pentium III
- Otros

De este modo, se realizará un tratamiento diferenciado para cada grupo, permitiendo que la herramienta funcione en cualquier procesador y realice las mediciones correctamente para cualquier procesador Intel de las familias P5 y P6.

El sistema operativo Windows NT se divide (en cuanto a seguridad se refiere) en dos partes: el código que se ejecuta en modo kernel y el código que se ejecuta en modo usuario. A su vez, existen determinadas operaciones que el CPU puede ejecutar en modo kernel y en modo usuario, y otras que solo puede ejecutar en modo kernel. El problema aparece cuando se desea escribir un programa (como una herramienta de monitoreo) que corre en modo usuario, y se desea hacer uso de determinadas instrucciones que solo pueden ejecutarse en modo kernel (como las instrucciones para obtener información de performance). Este tipo de problemas se resuelve a través de un driver.

De este modo, construyendo un driver que contenga en su código rutinas con las instrucciones para medir performance del CPU, se logrará realizar las mediciones sin que la seguridad de Windows NT lo impida. De ningún modo esto implica que haya un hueco en la seguridad de Windows NT que permita a cualquier aplicación realizar cualquier tipo de operación, ya que para lograr invocar a rutinas contenidas en un driver es necesario que este instalado, y no todos los usuarios del sistema tienen permiso para hacerlo.

5.3 Resultados y Conclusiones

Se llega a la conclusión de que para construir la herramienta con las características pedidas es necesario que la aplicación se ejecute en la forma de un servicio de Windows NT. Esto se basa principalmente en el hecho de que es posible ejecutar un servicio aunque ningún usuario haya iniciado sesión en el sistema operativo.

Se encuentran seis opciones diferentes de realizar las mediciones sobre los contadores de Windows NT y se determina bajo que circunstancias es conveniente utilizar cada una de ellas.

También se evalúa como implementar las mediciones de SNMP y los contadores del procesador Pentium, indicando en cada caso los métodos para realizarlo.

5.4 Evaluación

La investigación es una de las etapas en las cuales el grupo se mostró más animado. El interés del grupo se debía fundamentalmente a estar investigando temas por completo desconocidos.

Si bien en una primera instancia no surgía información significativa que diera frutos, posteriormente surgieron elementos suficientes para completar el objetivo del proyecto.

Se lograron conocimientos sólidos en lo concerniente al sistema operativo Windows NT, así como comprender cabalmente el problema planteado y comenzar a conjeturar posibles formas de resolverlo.

También se adquirieron los conocimientos necesarios como para poder implementar las mediciones de SNMP y los contadores del procesador Pentium.

Es por tanto, que se realiza una evaluación positiva de la etapa, ya que todos los ítems de la investigación fueron debidamente cubiertos, incluso algunos sobre los cuales no se tenía mucha esperanza.

6 Segunda Etapa – Prototipo

Se describe a continuación la segunda etapa, el Prototipo. Se detallan los objetivos de la etapa, destacándose los entregables que la misma genera. Se comenta el cronograma de trabajo, así como también se describe el desarrollo de la etapa, incluyendo los detalles de mayor relevancia. Finalmente, se comentan los resultados obtenidos y se realiza una evaluación general.

6.1 Objetivos

El objetivo de esta etapa consiste en la selección de una herramienta de desarrollo, familiarizarse con la misma y generar un primer programa ejecutable que cubra un mínimo de las funcionalidades requeridas.

De este modo se busca evaluar la dificultad de llevar a la práctica los estudios realizados previamente, e ir delimitando el alcance del proyecto.

6.1.1 Entregables

Se considera un único entregable, que consiste en un programa ejecutable que liste el nombre de todos los procesos activos en el sistema operativo, y el porcentaje de tiempo de CPU que los mismos están utilizando. Este programa debe poder ser ejecutado en la mayor cantidad de sistemas operativos Windows que sea posible.

Debido a que este entregable sirve principalmente para cumplir con los objetivos puntuales de esta etapa, y no se utilizará como base para las siguientes, no se pone especial énfasis en su documentación. Es por esto que se incluirán los fuentes del mismo en la entrega final del proyecto, pero no contará con ningún tipo de documentación adicional.

6.1.2 Cronograma

El tiempo asignado para esta segunda etapa es de aproximadamente un mes. El comienzo se ubica en Julio de 1999, ya que de hacerlo antes no se tendría suficientemente avanzado el estudio, y de hacerlo después dejaría muy poco margen a la etapa de desarrollo.

6.2 Desarrollo

A continuación se describe el desarrollo de esta etapa, donde se presentan varias dificultades que serán detalladas más adelante.

En primer lugar, resulta de gran importancia la selección de una herramienta de desarrollo. Si bien, no es necesario utilizar la misma herramienta en las etapas posteriores, se lo considera deseable debido a que permite una única etapa de familiarización con la misma en caso de tratarse de una herramienta desconocida para los integrantes del grupo. De este modo, previendo una importante necesidad de interactuar con elementos de bajo nivel de los sistemas operativos y la necesidad de eficiencia, se opta por Microsoft Visual C++ 6.0⁷.

Si bien se contaba con experiencia en programación en el lenguaje C++ bajo Unix, no se tenía ningún tipo de experiencia con el entorno de desarrollo Visual C++. En principio se desestimó este problema, creyendo que

⁷ De aquí en más se hará referencia a esta herramienta como Visual C++.

ambos lenguajes deberían ser sumamente parecidos, y que la diferencia de entornos no introduciría mayores dificultades.

Al comenzar con la implementación se observó la gran cantidad de diferencias existentes entre los entornos de desarrollo, principalmente a la hora de desarrollar aplicaciones con interfaz gráfica y orientación a eventos (jerarquía de clases de Windows), y en la interacción con el sistema operativo (Win32 API). Frente a estas diferencias, resultó claro que no sería trivial el aprendizaje. Es por esto que se opta por desarrollar este prototipo como una aplicación de consola⁸, minimizando así las dificultades mencionadas.

Luego, ya que se plantea como objetivo realizar un programa que pueda ser ejecutado en varios sistemas Windows, se opta por implementar uno capaz de funcionar correctamente en Windows 95, 98 y NT. Debido a que una computadora de oficina (objetivo principal de estudio) no ejecuta habitualmente Windows CE, y que no se cuenta con una versión de Windows 2000, no se consideran dichos sistemas operativos.

De este modo, se construye un programa que utiliza instrucciones que sean válidas en todos los sistemas mencionados. Esto se logra haciendo uso del subconjunto de posibilidades que brinda la Win32 API que resulta compatible con los sistemas operativos de interés. Para los casos puntuales en los que esto no es posible, se detecta bajo que sistema operativo se está ejecutando y se utiliza en cada caso las instrucciones adecuadas. En particular, para obtener la lista de todos los procesos activos sobre Windows 95 y 98 se utiliza la API ToolHelp32 que es la única herramienta con la que se cuenta, mientras que para Windows NT se opta por la API PSAPI debido a su simplicidad.

El uso de la PSAPI genera problemas sobre Windows 95 y 98, ya que está implementada como una DLL y dichos sistemas operativos no son capaces de cargarla en tiempo de ejecución debido a que la misma no está diseñada para ellos. Este problema puede ser resuelto utilizando la invocación explícita de la DLL cuando se lo considere necesario. Para profundizar en este aspecto, así como en las interfaces de aplicación mencionadas anteriormente, es posible recurrir al capítulo "Performance en Sistemas Operativos Windows" de [PRW].

6.3 Resultados

Se obtiene una familiarización básica con la herramienta de desarrollo Visual C++, que permite un manejo sumamente elemental de la misma.

Además, se logra construir un programa ejecutable con las características planteadas como objetivo.

6.4 Evaluación

Si bien se logra cumplir con los objetivos planteados para la etapa en tiempo y forma, se genera una cierta preocupación por la complejidad que presenta la herramienta de desarrollo.

De todas formas, no resulta clara la necesidad de hacer uso del potencial gráfico o la orientación a eventos de Visual C++, por lo que el nivel de conocimiento alcanzado podría ser suficiente.

⁸ Referirse a la documentación [VC++] por mayor información.

7 Tercera Etapa – Desarrollo

Se describe a continuación la tercera etapa, correspondiente al Desarrollo. Se detallan sus objetivos y los entregables que la misma genera. Se presenta el cronograma de trabajo, y se describe el desarrollo del mismo, incluyendo los detalles de mayor relevancia. Finalmente, se comentan los resultados obtenidos y se realiza una evaluación general.

7.1 Objetivos

A partir de los conocimientos adquiridos en la etapa de investigación, y los estudios de viabilidad realizados en la etapa del prototipo, se pretende realizar el desarrollo completo del producto final. A este producto se le llamará MRW.

En primer lugar, se procede a especificar claramente el alcance del producto, definiendo todas las funcionalidades del mismo.

Luego, se define la arquitectura general con la que se trabajará, la cuál se bajará de nivel progresivamente hasta completar el diseño.

Finalmente, se realiza la programación de la solución especificada, mientras que se genera un plan de pruebas y un manual de usuario.

7.1.1 Entregables

A medida que se avanza en el desarrollo de la etapa, se generarán diferentes entregables, los cuales serán ajustados a lo largo de todo el desarrollo, en la medida que resulte necesario.

Una vez que las funcionalidades, así como el alcance del producto estén claramente especificados, se procede a incluir dicho detalle en el entregable “Análisis de Requerimientos”, al que se hace referencia como [ANRQ].

La especificación completa del diseño del producto será descrita en el entregable “Especificación de Diseño”. A este entregable se hace referencia como [ESDN].

Se considerará un entregable que será el resultado de la programación de toda la aplicación, incluyendo cada uno de sus eventuales componentes, así como los elementos auxiliares que fueran necesarios para su correcta instalación.

Por último, se consideran los documentos “Plan de Pruebas” y “Manual de Usuario” a los que se hace referencia como [PLPR] y [MUSR] respectivamente.

Cabe destacar que para escribir cada uno de los documentos mencionados se utiliza el estándar de la IEEE correspondiente.

7.1.2 Cronograma

El tiempo asignado para el desarrollo de la aplicación es de cinco meses, y comienza en agosto de 1999. De este modo, luego de finalizada la etapa del prototipo, se comienza inmediatamente con el desarrollo, mientras aún continúa la investigación. Esto puede ser riesgoso, ya que nuevos descubrimientos de la etapa de investigación pueden provocar modificaciones relativamente importantes en el desarrollo, pero se

decide asumirlo para avanzar ágilmente en esta etapa, poniendo especial énfasis en los componentes de los que ya se dispone de suficiente conocimiento.

7.2 Desarrollo

En esta sección se cubre toda la etapa, realizando comentarios generales de cada uno de los entregables, sobre los que se puede profundizar recurriendo a cada uno de ellos en particular.

Cabe destacar que se comentará el proceso de cómo se llega a cada una de las soluciones planteadas, fundamentando la toma de decisiones cuando corresponda. Esta información sólo aparece en este documento y resulta muy valiosa para comprender las decisiones tomadas, así como para no tener la necesidad de re-evaluar los mismos problemas y soluciones en múltiples oportunidades.

7.2.1 Análisis de Requerimientos

En primer lugar se establecen las características que tendrá el producto a ser desarrollado durante el proyecto, describiendo sus alcances y limitaciones.

Cabe destacar que al igual que ocurre habitualmente en los procesos de desarrollo de software, los requerimientos van aumentando y/o modificándose con el transcurso del tiempo. Debido a estos cambios, y la falta de conocimiento sobre la complejidad que podría implicar la introducción de algunos de ellos se decide organizar el análisis de requerimientos en tres grupos: Mínimos, Deseables y Opcionales⁹.

En la categoría Mínimos se encuentran los requerimientos mínimos específicos que se considerarán para completar la aplicación. En la categoría Deseables se encuentran los requerimientos que se considera importante que tenga la aplicación pero que no son imprescindibles. Finalmente en la categoría Opcionales se encuentran los requerimientos que no fueron solicitados, pero que vistas las virtudes y carencias de los productos similares existentes podría resultar interesante incluir.

En términos generales, la herramienta permitirá la extracción en tiempo real de información referente a un sistema Windows NT, y eventualmente extender su alcance a otros tipos de mediciones tales como SNMP y contadores de performance de procesadores Pentium. Finalmente, se ejecutará sobre Windows NT aunque ningún usuario haya iniciado sesión y grabará esta información en un DBMS, permitiendo una configuración amigable de parámetros.

7.2.2 Especificación de Diseño

Se establecen las bases para la etapa de implementación y verificación del sistema, así como una guía para su desarrollo.

El sistema se compone de dos módulos principales: Configurador y Servicio.

El Configurador será el encargado de especificar todo el comportamiento que tendrá el sistema y será la cara visible del mismo para el usuario, mientras que el Servicio realizará toda la tarea de medición y almacenamiento según se haya especificado.

Esta separación se ve motivada por el hecho de que la herramienta debe ser capaz de realizar mediciones aunque ningún usuario haya iniciado sesión en el sistema operativo. El modo habitual de lograr este

⁹ Las categorías Mínimos y Deseables son dadas por los tutores, mientras que la de Opcionales es propuesta por el grupo de trabajo.

comportamiento en Windows NT es programar la aplicación para que funcione como un servicio, pero esto introduce un nuevo problema, ya que los servicios no son capaces de interactuar con el usuario como el resto de los programas de Windows, al no tener ventanas asociadas. Es por esto, que para facilitar esta interacción, se decide incluir un nuevo componente que realice el nexo entre el usuario y el servicio.

Dada la separación del sistema en dos módulos independientes, resulta de gran importancia la interacción entre los mismos. Esta interacción se da a través de un archivo de configuración que es generado por el configurador, y procesado por el servicio para obtener los datos necesarios. Si bien resulta claro que esta información debe ser almacenada, dada la tendencia actual de la programación bajo Windows hubiera resultado más razonable almacenar esta información en el registro del sistema (Registry). Se opta por el archivo de configuración, debido a la flexibilidad que el mismo brinda, tanto para posibles modificaciones manuales, como para permitir la configuración de múltiples computadoras de una misma red, simplemente copiando un archivo.

También existirá una interacción directa entre ambos módulos, que permitirá controlar las características básicas del módulo Servicio (instalar, desinstalar, comenzar a ejecutar y detener la ejecución) desde el módulo Configurador. Si bien estas características pueden ser controladas con las herramientas estándar de Windows NT, el hecho de incluirlas en el módulo Configurador permite un control del sistema desde un lugar único, lo que facilita su uso.

Los resultados obtenidos a partir de las mediciones se almacenarán en una Base de Datos a través de ODBC. Si bien el uso de ODBC introduce una complejidad mayor en el código, permitirá una gran independencia sobre el DBMS a utilizar, lo que producirá una herramienta mucho más flexible.

Cabe destacar que los servicios en Windows 95/98 se comportan de un modo diferente, por lo que habría que abordar un enfoque completamente distinto para estos sistemas. Es por esto que se opta por realizar un desarrollo para Windows NT y un esfuerzo extra por mantener la compatibilidad con Windows 2000.

7.2.3 Programación

En esta sección se comentarán los problemas y dificultades enfrentados a lo largo de la programación de toda la aplicación, incluyendo cada uno de sus componentes, así como los productos auxiliares que fueran necesarios para su correcta instalación.

Tal como se explica en la etapa prototipo, se opta por Visual C++ como herramienta de desarrollo, a pesar de las dificultades que esto presenta a la hora de desarrollar aplicaciones que interactúen con el usuario y/o sean orientadas a eventos. Resulta claro que esto no presentará mayores problemas para el desarrollo del módulo Servicio, pero que presentaría importantes problemas en el desarrollo del módulo Configurador. Es por esto, que el módulo configurador es desarrollado en Microsoft Visual Basic 6.0¹⁰ (herramienta de fácil uso ya conocida por los integrantes del grupo) en su mayoría, y se recurrirá a Visual C++ únicamente cuando sea inevitable.

7.2.3.1 Módulo Servicio

En este punto se detallarán las características principales del desarrollo del módulo Servicio. Para esto se describen las características generales y luego se presentan los principales submódulos.

La base principal del módulo Servicio es una clase de C++, que provee las principales características necesarias para permitir el desarrollo de un programa que cumpla con los requisitos necesarios para ser un servicio de Windows NT. Esta clase fue obtenida en [CLASE] y luego de algunas modificaciones

¹⁰ De aquí en más se hará referencia a esta herramienta como Visual Basic.

menores, se construye la clase del módulo Servicio a través de heredar la clase anteriormente descrita y la redefinición de las funciones de interés.

Debido a las dificultades para realizar el debug de un servicio y las restricciones de interfaz que el mismo presenta, se optó por mantener el desarrollo de este módulo como un programa habitual de Windows hasta que el mismo estuvo en una etapa avanzada. En el momento en que se decide convertir el módulo en un servicio se presentan algunos problemas debido a diferencias que el entorno Windows introduce, tales como la pérdida de visibilidad de las fuentes de información (DSN) de ODBC a nivel de usuario¹¹, y la modificación del directorio de origen del ejecutable.

7.2.3.1.1 Windows NT

Este submódulo es el encargado de realizar todas las mediciones de los contadores existentes en el sistema Windows NT.

Tal como se presenta en el capítulo Performance de [PRW], existen muchas formas diferentes de realizar este tipo de mediciones. Debido a que el sistema sobre el que se realizarán las mediciones es Windows NT y eventualmente Windows 2000, y la gran cantidad de contadores que se desean medir, se opta por seguir las recomendaciones que allí aparecen y utilizar la herramienta PDH.

Cabe destacar, que siguiendo el principio de generalidad¹², no se limitará la herramienta a los contadores solicitados explícitamente en el análisis de requerimientos, sino que se incluirán todos los que el sistema operativo permita medir.

7.2.3.1.2 SNMP

Este submódulo es el encargado de realizar todas las mediciones a través de SNMP a los dispositivos pertenecientes a la red donde se encuentra la computadora que está ejecutando la aplicación. Esta podrá realizar este tipo de mediciones únicamente si los componentes de SNMP de Windows NT están correctamente instalados.

Para el desarrollo de este submódulo se toma como base la utilidad [SNMPUTIL].

A lo largo del desarrollo se presentan varias dificultades debido a los tipos de datos particularmente complejos que maneja SNMP, y el manejo de memoria que necesita la implementación de SNMP que realiza Microsoft.

7.2.3.1.3 Pentium

Este submódulo es el encargado de realizar todas las mediciones sobre el procesador perteneciente a la computadora que está ejecutando la aplicación. Esta podrá realizar este tipo de mediciones únicamente si el procesador es Pentium.

Tal como se explica en el capítulo “Performance en los CPU Intel” del [PRW], la única forma de lograr este tipo de mediciones sobre Windows NT es a través de un driver. La introducción de un

¹¹ Este problema puede ser resuelto definiendo el inicio del servicio a través de una sesión de usuario en particular, que tenga definido el DSN. De todas maneras, esto no sería una buena solución, ya que no es el método que se utiliza habitualmente.

¹² Principio recomendado en la Ingeniería de Software para el desarrollo de productos.

driver al producto presenta dos dificultades importantes: la construcción del mismo y su posterior uso.

Para construir el driver fue necesario consultar gran cantidad de artículos, los que se comentan en detalle en el capítulo mencionado, a partir de los cuales se determina un posible proceso de construcción. En primer lugar se construye, con la ayuda de uno de los artículos, un programa wizard¹³ para el desarrollo de drivers, que se logra incluir en el entorno de desarrollo Visual C++. Luego, a través de la ejecución de dicho programa, se genera un esqueleto básico del código fuente de un driver. Y finalmente, sobre dicho código se realizan modificaciones hasta lograr la funcionalidad buscada. Cabe destacar que debido a la naturaleza de la funcionalidad, las partes claves del código deben ser programadas en Assembler. Una vez programado el driver, es necesario conseguir herramientas especiales para lograr compilarlo, tales como [DDK].

El driver debe estar cargado e iniciado para poder utilizarlo. Debido a que dejar este aspecto en manos del usuario final podría complicar bastante el uso de la herramienta, se opta por cargar en forma dinámica el driver cuando se lo necesite. Además, la comunicación entre una aplicación y un driver tiene diversas restricciones que dificultan la programación.

A las dificultades mencionadas, cabe agregar que es sumamente complicado realizar el debug de un driver, ya que cualquier procedimiento requiere software especial y al menos dos computadoras. Y a esto debe sumarse el hecho de que cualquier error que se produzca dentro del driver no podrá ser interceptado por Windows NT como sucede con las aplicaciones habituales, provocando así consecuencias impredecibles.

7.2.3.1.4 ODBC

Este submódulo es el encargado de la comunicación entre la aplicación y el DBMS. Para esto se necesita que exista un DSN declarado a nivel de sistema en ODBC, y que el mismo este correctamente configurado.

Las principales dificultades que se presentan al desarrollar este submódulo es la gran cantidad de consideraciones que deben realizarse al utilizar ODBC. Algunos ejemplos de las diferencias que presentan los manejadores son:

- al ser los tipos de datos denominados distinto por cada manejador y no todos los manejadores soportar los mismos tipos, se vuelve complicado crear las tablas necesarias para la aplicación
- los prefijos y sufijos necesarios para cada tipo de datos son diferentes en cada manejador
- las restricciones, tales como la necesidad que los campos definidos como “primary key” sean “not null” existen únicamente en algunos manejadores
- la sintaxis SQL es ligeramente diferente en cada manejador, ya sea en detalles tales como finalizar las sentencias con “;”, o diferencias más importantes como los comandos “for update” y “no wait” de Oracle.

También se presentan algunas dificultades al buscar la eficiencia a través de utilizar sentencias pre-compiladas.

Ya avanzada esta etapa, se encuentra una herramienta superior a ODBC llamada OLE DB, e incluso una superior a esta llamada ADO, las cuales encapsulan las principales dificultades mencionadas anteriormente. Lamentablemente ya no fue posible, debido a la escasez de tiempo, realizar las modificaciones necesarias para su utilización.

¹³ Programa que a través de consultas simples al usuario realiza una tarea compleja.

7.2.3.2 Generador

En este punto se detallarán las características principales del desarrollo del módulo Generador, así como sus principales componentes.

La estructura general del Generador es similar a los programas comúnmente desarrollados para el ambiente Windows. Este desarrollo se realiza con Visual Basic, utilizando las facilidades que brinda dicho lenguaje.

Según se explicó anteriormente, el generador interactúa de dos maneras diferentes con el módulo Servicio. Esta comunicación excede las posibilidades estándar de Visual Basic por lo que se utilizan diferentes extensiones. La interacción a través de un archivo de configuración utiliza la API que brinda Windows para tales efectos, mientras que para realizar la interacción directa con el servicio se desarrolla una DLL en Visual C++.

En particular, para realizar la selección de los contadores de Windows NT, se utiliza la herramienta PDH, que brinda funciones especialmente realizadas para Visual Basic, que incluyen hasta una interfaz gráfica que facilita la programación. Cabe destacar que al utilizar la versión de 1997 de la PDH aparecen varios problemas debido a diversos bugs, por lo que se opta por utilizar la versión de 1999, que se distribuye con las versiones beta de Windows 2000.

7.2.4 Manual de Usuario

Servirá como guía a los usuarios, ya sea para aprender a utilizar la aplicación, o como fuente de consulta para el usuario que ya conoce como utilizarla.

Este manual cubre gran diversidad de temas, entre los que se destacan los procedimientos de instalación de la aplicación, explicación de su uso (a nivel de usuario común o avanzado), y el detalle de los errores más comunes con sus respectivas soluciones.

7.2.5 Plan de Pruebas

Se busca cubrir un conjunto mínimo de pruebas que permitan descubrir los principales errores que eventualmente contenga el programa. Los resultados de las pruebas servirán para realizar correcciones sobre el producto.

Cabe destacar que este entregable no sólo es generado en esta etapa, sino que es utilizado hasta el final de la misma, donde se eliminan todos los errores detectados.

A través de las distintas corridas del plan de pruebas que se realizan, se detectan varios errores. Tal vez los más interesantes sean:

- La detección de diversos problemas de compatibilidad de la aplicación con las versiones de Windows en idiomas diferentes al inglés.
- Aumento de memoria al realizar mediciones a través de SNMP.
- Problemas de compatibilidad con procesadores que no son del fabricante Intel.
- Problemas de compatibilidad con los manejadores Interbase, SQL Server y Oracle.

Afortunadamente, todos los errores detectados han podido ser resueltos con éxito.

7.3 Resultados y Evaluación

Se logra especificar e implementar la herramienta propuesta, con una muy buena calidad destacando la baja cantidad de errores detectados y su rápida corrección.

La herramienta cumple con todos los requerimientos especificados, incluso con aquellos que se calificaron de opcionales o deseables, a pesar de la gran complejidad de algunos de ellos.

Se generó toda la documentación necesaria para permitir la comprensión y la evolución de la herramienta a lo largo del proceso de desarrollo. Si se suma a esto la claridad lograda en el código fuente a través de criterios de programación prolijos y comentarios explicativos, resulta clara la facilidad de continuar trabajando sobre la herramienta para extenderla.

A pesar de haber desarrollado una herramienta dirigida a usuarios con gran experiencia en el área, se puso especial hincapié y se logró que esta fuera muy fácil de usar, sin perder la flexibilidad que habitualmente brindan las herramientas más complejas. Esta facilidad se ve incrementada por contar con un manual de usuario, que permite comprender fácilmente todas las posibilidades provistas.

8 Cuarta Etapa – Informe Final y Presentación

Se describe a continuación la cuarta etapa, correspondiente al Informe Final y a la Presentación. Se detallan los objetivos de la misma, destacándose los entregables que genera. Luego se comenta el cronograma de trabajo, así como también el desarrollo, en donde se describe con todo detalle las situaciones originadas. Finalmente, se presentan los resultados obtenidos y se realiza una evaluación.

8.1 Objetivos

En la presente etapa se identifican tres objetivos de gran importancia.

El primero de ellos es construir un documento único que englobe a todos los obtenidos anteriormente.

El segundo es la construcción de una página Web, la cual brinde toda la información sobre el proyecto para que diferentes personas de todo el mundo puedan tener acceso a este trabajo.

El último de ellos consiste en la preparación de la presentación del proyecto.

8.1.1 Entregables

En primer lugar se considera el documento Informe Final (el presente documento). Dicho documento permitirá al lector tener una visión global y clara del proyecto, y servirá de nexo entre los diferentes documentos del mismo. Además, se realiza la compaginación de toda la documentación obteniendo la carpeta correspondiente a la documentación final del proyecto.

Por otra parte, quedará publicado en Internet la página Web del proyecto que contendrá toda la información sobre el mismo, incluyendo documentación, fuentes y aplicaciones.

Por último, se construirán transparencias con la herramienta Microsoft PowerPoint 97, las cuales serán utilizadas en la presentación final del proyecto.

8.1.2 Cronograma

El tiempo asignado para el desarrollo de esta etapa es de tres meses, extendiéndose de noviembre a enero de 2000.

8.2 Desarrollo

A continuación se describe el procedimiento seguido para la obtención de cada uno de los entregables de esta etapa.

8.2.1 Informe Final

Cuando comenzó esta etapa se disponía de los diferentes documentos generados durante el proyecto, y se debía realizar un único documento que los englobara. Para lograrlo se redacta un Informe Final Ejecutivo, el cual da una visión global y clara del proyecto, dando en cada capítulo las referencias necesarias para una profundización sobre cada tema.

A pesar de ser un documento más corto que otros escritos anteriormente, sin duda es el más complejo de elaborar. Esto se debe principalmente a que se desea resumir todo lo acontecido a lo largo del proyecto en un número reducido de páginas. Por lo dicho anteriormente, se opta por dedicarle mucho más tiempo de trabajo en equipo para escribir este documento que el empleado en los documentos anteriores.

Luego de terminado el Informe Final, se compaginan todos los documentos generados durante el transcurso del proyecto obteniendo esta carpeta correspondiente a la documentación final del proyecto.

8.2.2 Página Web

Para realizar el diseño de la página Web se toma como base el Informe Final, para dar una introducción al proyecto. Se utiliza para realizar dicho diseño Microsoft Word y Microsoft FrontPage.

A su vez se incluyen links a todos los otros documentos generados durante el proyecto así como a los distintos artículos consultados que se encuentran disponibles en Internet.

Además se brindan los fuentes obtenidos durante el proyecto, los ejecutables finales y un link a una casilla de correo para realizar consultas.

Por ultimo se incluyen links a diversos sitios de interés que pueden ser útiles para profundizar en los diferentes temas mencionados a lo largo de la documentación.

8.2.3 Presentación

Para preparar la presentación, se comienza por la planificación de los temas a tratar a lo largo de la misma.

Una vez hecho esto, se determina el tiempo por el que se debe extender la presentación, y en base a esto se diseñan las transparencias que se utilizarán.

Para realizar dichas transparencias se utiliza la herramienta Microsoft PowerPoint 97.

8.3 Resultados y Evaluación

Se logra cumplir de modo satisfactorio con los objetivos propuestos, obteniendo la documentación final, así como la página Web del proyecto. Dicha página se encuentra accesible a través de Internet, en la dirección <http://www.pareno.fing.edu.uy:8080/mrw/>.

Al momento de finalizar de escribir este documento todavía no se había comenzado a preparar la presentación, por lo que no se esta en condiciones de evaluarla.

9 Resultados y Conclusiones

Transcurrido el proyecto, se informa sobre los resultados obtenidos. Luego, se comentan los puntos que se consideran importantes tener en cuenta para posibles extensiones del producto. Finalmente se hace una evaluación del proyecto.

9.1 Resultados obtenidos

Se ha logrado cumplir con todos los objetivos planteados durante el proyecto, ya que se llega a implementar una herramienta capaz de realizar las mediciones propuestas.

En particular los requerimientos mínimos fueron totalmente cumplidos, logrando medir todos los contadores de Windows NT solicitados específicamente y su resto. Cabe destacar que se graban todas las mediciones vía ODBC lo que permite abstraerse de los diferentes manejadores de bases de datos y que la aplicación se ejecuta como un servicio de Windows NT permitiendo así cumplir con la totalidad de los requerimientos inicialmente planteados.

Por otra parte, los requerimientos deseables también fueron satisfechos en su totalidad al extender los contadores de Windows NT permitiendo su discriminación por hilos, al agregar mediciones a través de SNMP y al monitorear los contadores de performance que proveen los procesadores de la línea Pentium de Intel.

Finalmente, se logra cumplir con los requerimientos opcionales a través del desarrollo del módulo Generador, que permite configurar y controlar de modo amigable toda la aplicación, y el agregado de la reconfiguración automática para el módulo Servicio.

Incluso, se extiende la herramienta por fuera de los requerimientos, para brindar la posibilidad de monitorear cualquier contador de Windows NT que este disponible en el sistema sobre el que se esta ejecutando la aplicación.

Cabe destacar que gracias al esfuerzo realizado en la búsqueda de compatibilidad con Windows 2000, se logrará que el sistema mantenga su vigencia luego del inminente lanzamiento de la nueva versión de Windows.

Al final del proyecto se logra obtener la documentación completa del mismo. Los diferentes documentos tratan los distintos temas y etapas del proyecto, los cuales se encuentran presentados en esta carpeta.

9.2 Problemas abiertos

En este punto se describen las principales áreas en las que no fue posible profundizar por diversos motivos, y que en base a la experiencia lograda en este estudio resulta interesante investigar, para así extender la herramienta.

En primer lugar, es posible incluir tres nuevas funcionalidades en enriquecerían de manera muy importante al sistema desarrollado.

La primera es un módulo que permita tomar los datos generados desde el DBMS, permitiendo su representación gráfica, comparación y eventual análisis. Incluso, estas gráficas podrían ser generadas bajo demanda y accedidas a través de un Web Browser en tiempo real. Podría ser interesante permitir realizar consultas simples para que todo el análisis pueda ser realizado usando la herramienta y no tener que recurrir a otros productos.

Sería interesante contar con un módulo que oficie de agenda, y permita definir varios tipos de mediciones. Cada una de estas mediciones podría programarse para diferentes horarios, con diversa periodicidad (por ejemplo: diaria o semanal).

También podría incluirse un módulo que permita definir condiciones y eventos a disparar automáticamente si se cumple alguna de ellas. De este modo, frente a al comportamiento extraño de una aplicación o dispositivo podría estudiarse automáticamente en mayor profundidad, reconfigurarse, apagarse o cerrarse según corresponda. Usando herramientas de estas características podría lograrse automatizar algunas tareas del administrador de una red.

En segundo lugar, es posible extender las actuales características del sistema. El módulo encargado de realizar las mediciones sobre los contadores de performance de procesadores Intel Pentium podría ser extendido para realizar estas mediciones sobre procesadores de otros fabricantes que permitan facilidades similares.

Por último, existen algunas mejoras técnicas, que si bien no agregan nuevas facilidades al producto final podrían mejorar su rendimiento y facilitar su crecimiento para un futuro.

En este sentido, se podría modificar el código para hacer uso de Unicode¹⁴, lo que permitiría el correcto funcionamiento del sistema en las versiones de Windows en idiomas que requieran más de 8 bits para su juego de caracteres.

Además, el uso de la tecnología ADO - OLE DB permitiría simplificar gran parte del trabajo realizado con ODBC, y ampliar de forma muy importante las posibilidades de extender la herramienta.

Finalmente, sería interesante hacer un uso más intensivo de la facilidad de hilos que brinda Windows para realizar las mediciones de cada uno de los módulos de forma simultánea, y eventualmente optimizar la ejecución dentro de los módulos que así lo ameriten.

9.3 Evaluación

Es posible evaluar el proyecto desde varios puntos de vista.

Desde el punto de vista del tiempo en el proyecto, el calendario inicial se vió claramente alterado durante el desarrollo del proyecto, y las etapas no fueron parejas en tiempo.

Se detallan en la tabla siguiente algunos datos que se consideraron de interés para visualizar el desarrollo del proyecto.

Datos de Interés	Proyecto Original	Proyecto Real	Factor de variación
Horas totales por integrante durante el proyecto	510	795	+ 55,88 %
Semanas totales de duración	34	43	+ 26,47 %
Horas promedio por semana por integrante	15	18,5	+ 23,33 %
Horas totales por integrante (Prototipo + Desarrollo)	330	335	+ 1,52 %

Tabla 4: Datos Interesantes para evaluar el proyecto

El primer dato es la cantidad de horas destinadas por integrante durante el proyecto. Originalmente se considero que el proyecto podía ser realizado con 510 horas por integrante del equipo. La realidad fue que se necesitaron

¹⁴ Unicode es un juego de caracteres internacional que define valores de 16 bits únicos para la mayoría de los juegos de caracteres conocidos en el mundo. Se puede consultar [MSDN] por más información.

795 horas por integrante (un 55 % más de lo original). Esto se debe en parte, a que se fueron agregando requerimientos para obtener una herramienta más interesante y flexible.

El segundo dato es la duración total en semanas, que nos permite apreciar el desfase entre la propuesta original y la realización del proyecto. Este desfase es producto de que en el cronograma original, el proyecto terminaba en noviembre, plazo que resultó imposible de cumplir.

El tercer dato es la cantidad de horas promedio por semana por integrante. Dicho indicador nos muestra que si bien el proyecto tuvo una mayor duración no se debió a una baja en la dedicación, sino que la dedicación fue también mayor de la esperada originalmente.

El último dato es la cantidad de horas totales destinadas por integrante a las etapas de prototipo y desarrollo. Se advierte que no hubo grandes variaciones, por lo que podría conjeturarse que las tareas que se descartaron en la etapa de prototipo se realizaron durante el desarrollo (hecho que efectivamente sucedió).

A partir de estos datos se puede ver que el proyecto creció más allá de lo estipulado y que esto no fue a causa de la etapa de desarrollo. Como se vio durante este informe y como se puede apreciar al leer el documento [PRW] a la etapa de investigación se le destinó muchas más horas de las estipuladas (resulta interesante comparar las horas originales con las reales como se ve en las gráficas de la sección 4.3). A la luz de los resultados, parece acertado haber extendido dicha etapa, ya que los conocimientos allí adquiridos, al realizar una investigación profunda y detallada, han sido de suma utilidad para lograr un producto completo y de buena calidad.

En cuanto a los requerimientos pedidos todos fueron logrados, incluso se agregaron más funcionalidades que las solicitadas.

Con respecto a la documentación, se considera que cubre todos los temas de interés, y permite comprender claramente el producto construido, así como la evolución del proyecto y las principales dificultades enfrentadas. Por otra parte, cabe resaltar que sirve como una guía para procesos posteriores de extensión del producto.

Además, dado el resultado de la verificación y el proceso de desarrollo utilizado, es posible afirmar que se obtuvo una herramienta de muy buena calidad.

La experiencia vivida a lo largo del proyecto ha sido muy importante, se logra seguir la metodología de trabajo que se había planteado, manejando diferentes etapas, teniendo reuniones semanales, por momentos separando el trabajo y por otros trabajando juntos, y realizando reuniones con los tutores. Todo inmerso en un tema del cual no se había profundizado hasta ahora en la facultad.

Se aprende un poco más a trabajar en equipo, a interactuar con los usuarios (tutores), se ingresa en un tema del cual no se tenía mucho conocimiento pero que resulta muy interesante y del cual se logra aprender mucho en este año de trabajo.

Tal vez más importante que el conocimiento adquirido sea el haber aprendido a buscar información en diversas fuentes, abriendo puertas en el terreno del autoaprendizaje.

Luego de haber evaluado diversos aspectos del proyecto, tales como su desarrollo y resultados obtenidos, de modo muy positivo, se puede concluir que el mismo resulta sumamente satisfactorio.