

Sistema de Información Gerencial de Bedelía

Informe Principal

Taller V - 1999

Tutor: Dr. Ing. Raúl Ruggia

Integrantes: Luis Aguerrebere
Mariela De León
Freddy Hernandez
Carlos Zabaleta

1 - INTRODUCCION.....	3
EL PROBLEMA.....	4
OBJETIVOS DEL PROYECTO.....	4
INTERÉS ACADÉMICO DEL PROYECTO.....	5
CONTRIBUCIÓN DEL PROYECTO.....	5
ANTECEDENTES.....	5
EVOLUCIÓN DEL PROYECTO.....	6
VISION GENERAL DEL SISTEMA.....	7
2 - FUNCIONALIDAD.....	8
¿QUÉ PERMITE EL SISTEMA?.....	9
OTRAS CARACTERÍSTICAS DEL SISTEMA.....	12
<i>Resumen de las preguntas que le ayuda a contestar el sistema</i>	14
EXTENSIONES DEL PROYECTO.....	15
3 - DISEÑO.....	16
TECNOLOGÍAS USADAS.....	17
ARQUITECTURA DEL SISTEMA.....	18
DISEÑO RELACIONAL DEL DW.....	19
<i>Tablas del datawarehouse:</i>	20
DISEÑO MULTIDIMENSIONAL.....	21
<i>Diseño Conceptual.</i>	21
<i>Diseño Lógico.</i>	23
DISEÑO DE LA INTERFASE.....	25
<i>Diseño detallado.</i>	26
4 - IMPLEMENTACION.....	32
IMPLEMENTACIÓN.....	33
<i>Visión general</i>	33
<i>Interfase Web</i>	33
<i>Olap</i>	39
5 - CONCLUSIONES.....	42
CONCLUSIONES.....	43
MEJORAS.....	44
AGRADECIMIENTOS.....	45
 BIBLIOGRAFÍA.....	 46

1 – INTRODUCCION.

EL PROBLEMA.

Los sistemas de información gerencial tienen como objetivos principales ofrecer información de calidad para análisis gerencial y toma de decisiones. Para esto extraen datos de bases de datos fuentes, se transforman y se presentan al usuario a través de diferentes herramientas Front-End que hacen posible su análisis. En la actualidad estos sistemas se implementan siguiendo una arquitectura llamada de Sistema de Datawarehousing.

El Decanato de la Facultad de Ingeniería y las secretarías asociadas requieren de información gerencial relacionadas, por ejemplo, con las actividades de enseñanza. Estos requerimientos se aplican también a los institutos, en particular al Instituto de Computación (In.Co).

La información base para resolver estos requerimientos se encuentra almacenada en la BD de Bedelía a la cual no tienen acceso los docentes, asistentes, Unidad de Enseñanza, etc., interesa por ejemplo: tener una visión general de la aprobación de cursos por parte de una generación, o en cursos de un cierto año, saber cuantos estudiantes pueden potencialmente inscribirse en un curso, correcciones de exámenes de un Instituto.

Este proyecto de Taller 5 apunta al desarrollo de un Sistema de Información Gerencial que cubra requerimientos planteados en Decanato y en el In.Co (tomado como caso tipo de instituto).

Para el desarrollo del mismo se construirá un Sistema de Datawarehousing cuya fuente de datos sea la BD de Bedelía. Los problemas a resolver están relacionados con el acceso a bases de datos y su integración, la automatización de la carga, el control de calidad de datos, manejo de datos en herramientas OLAP, y el acceso a los resultados a través de visores html. Se trata de aplicar y experimentar técnicas y herramientas de apoyo al diseñador, algunas de ellas propuestas en el equipo In.Co-CSI (Concepción de Sistemas de Información).

Este proyecto parte de trabajos realizados en otros proyectos similares en 1997 y 1998, los cuales avanzaron en diferentes aspectos de estudio de la tecnología, análisis de requerimientos y de estudio de la integración de los datos.

OBJETIVOS DEL PROYECTO.

El objetivo del proyecto es el desarrollo de un Sistema de Datawarehousing para Decanato de la Facultad de Ingeniería y el InCo que integre información tal como: resultados de cursos, análisis de las actividades de un estudiante, etc.

Los principales aspectos técnicos a resolver son:

- Aplicación de tecnologías OLAP-Híbridas en la implementación de Data Marts. La implementación se realizó en OLAP-Services de MS-SqlServer 7 . Para esto se debió realizar una revisión del diseño multidimensional y del Data Warehouse aplicando técnicas y modelos propuestos por el Laboratorio CSI (In.Co).
- Diseño e implementación de interfase a usuario a través de visores html. En este aspecto se definió la tecnología y herramientas a utilizar.

INTERÉS ACADÉMICO DEL PROYECTO.

El área de Data Warehousing es una de las más activas actualmente tanto a nivel académico como industrial. Una de las características importante que presenta, es que se basa en la integración de soluciones técnicas más que en productos completos. Algunas de las soluciones técnicas son las herramientas OLAP, herramientas de Extracción, Integración y Control de Calidad de Datos. Un punto particularmente interesante es el uso de interfaces html para análisis gerencial.

El equipo de Concepción de Sistemas de Información (CSI) se encuentra trabajando en el estudio y propuesta de técnicas de diseño de Sistemas de Data Warehousing. Este proyecto de Taller 5 se enmarca en dichas actividades. Por lo tanto resulta de interés la experimentación de diferentes técnicas en contextos razonablemente realistas pero controlados.

CONTRIBUCIÓN DEL PROYECTO.

Los principales aportes de éste proyecto se resumen en:

- Construcción de un Sistema de DW de Bedelía.
- Construcción de un browser *genérico* de cubos a través de visores html. Es de destacar que el objetivo inicial fue construir un visualizador específico para nuestro datawarehouse, sin embargo logramos algo más allá de esto.
- Evaluación de las herramientas Microsoft disponibles para ello: Sql-Server 7.0, OLAP Services, servidor web Internet Information Server (IIS), programación de páginas ASP, Visual Basic.
- Análisis de los datos extraídos del RDBMS de Bedelía por integrantes del equipo de desarrollo del Sistema de Gestión de Bedelías, lo cual implica un conocimiento más exacto de los mismos, lográndose de esta forma que el producto obtenido este pronto para su instalación y uso.

ANTECEDENTES.

El proyecto continúa el camino comenzado en el año 1997 por el proyecto “Estudio de Técnicas y Software para la construcción de sistemas de DW” [1] y de los dos proyectos del año 1998: “Sistema de Datawarehousing, Carga y Control de Calidad”[2] y “Desarrollo de un Sistema de Data Warehousing para el INCO” [3].

Los puntos a destacar en dichos proyectos son, el desarrollo de un Sistema de DW para el In.Co y la metodología y herramientas utilizadas.

Principalmente nos basamos en los proyectos del año 1998 ya que en ellos se profundizó en el conocimiento de la realidad y de las preguntas que se querían contestar entrevistando a los interesados. También se destaca un estudio detallado de la carga y control de calidad de los datos que en la primera etapa de nuestro proyecto pretendimos re-utilizar.

No hay antecedentes en los talleres anteriores sobre el tema interfase ya que en ellos no se trató el tema porque se trabajaba con software que ya incluía una interfase apropiada (Power-Play de Cognos). En nuestro taller, con las herramientas disponibles en Facultad (NT Server, Sql-Server, VisualStudio, Office, etc), tenemos facilidades de Office 2000 para resolverlo, pero al tener en cuenta la diversidad de plataformas y browsers existentes, esta opción no es viable. Por este motivo buscamos otras alternativas que fueran menos "propietarias", y encontramos algunos ejemplos, que si bien solo funcionaban con MS Internet Explorer, de ellos se podían aprovechar algunas ideas. [4] y [12]

EVOLUCIÓN DEL PROYECTO.

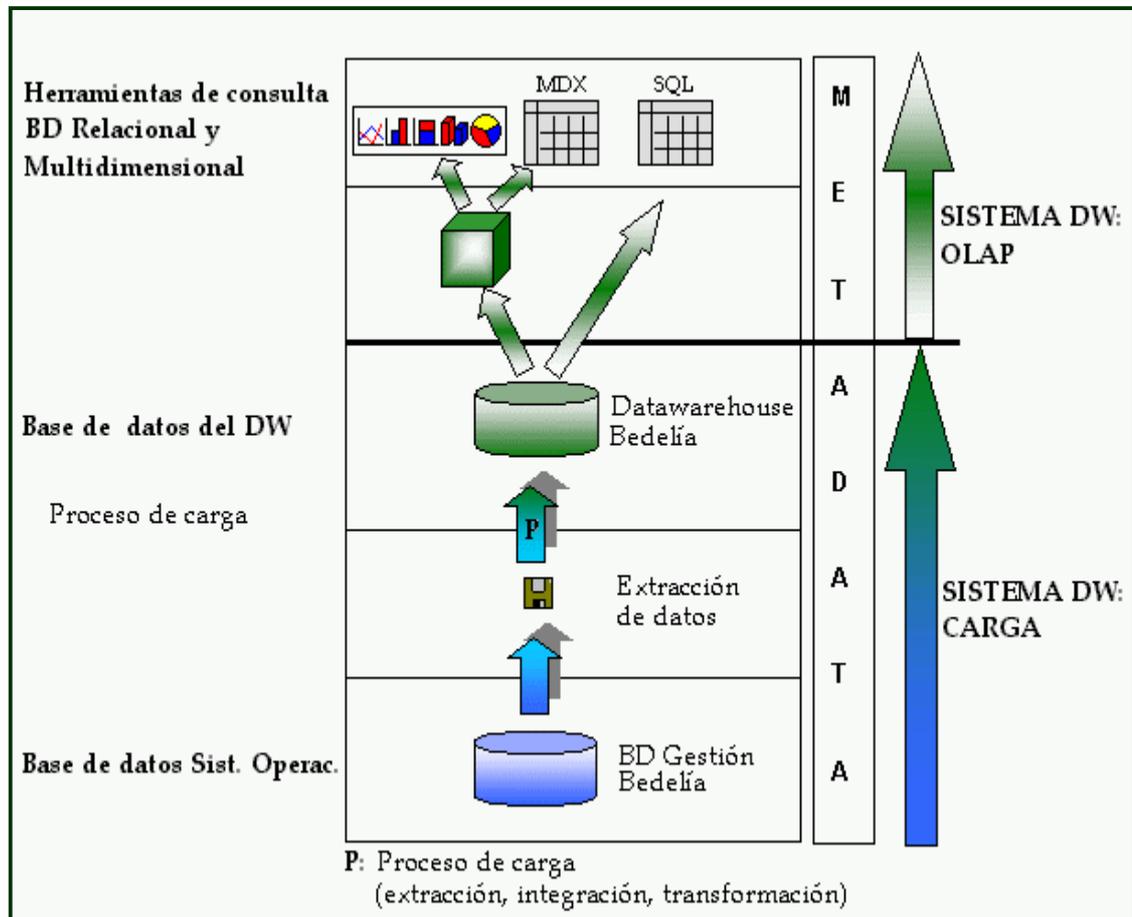
La idea era continuar los proyectos del año 1998 con pocos cambios y abocarnos al estudio de las herramientas Microsoft y a la construcción de una interfase web, tomando casi por completo la implementación de la carga del datawarehouse realizada con procedimientos PL/SQL de ORACLE y también la definición de los cubos. Es decir queríamos re-utilizar código para transportarlo al DTS (Data Transformation Server) de Microsoft y OLAP Services, lo cual luego de un estudio detallado concluimos que no era completamente automatizable, ni tampoco inmediato. Por otro lado analizamos la definición de los cubos y las preguntas que se querían contestar, los mismos no permitían el estudio por Materias e Institutos dentro de la Facultad sino que el estudio era por asignatura.

Todo esto llevó a que el trabajo se repartiera de la siguiente manera:

- Re-análisis de los cubos propuestos por los talleres anteriores y como consecuencia las tablas del datawarehouse.
- Planificación de consultas SQL para la extracción de los datos necesarios de la base de gestión de Bedelía.
- Un grupo de docentes del In.Co. realizó la carga del datawarehouse, a partir de nuestras especificaciones dado que la re-utilización de los procesos no era posible por cuestiones de incompatibilidad de las herramientas, además de los cambios propuestos para la estructura del datawarehouse.
- Definición y carga de los cubos. Optimización de los procesos.
- Construcción de una interfase html como herramienta Front-End.

VISION GENERAL DEL SISTEMA.

El sistema puede esquematizarse de la siguiente forma:



En la figura se puede observar el proceso para obtener como resultado un producto que permite consultar y analizar a través del Web datos acerca de las actividades cursos, exámenes, etc. de los estudiantes de cualquier carrera de la Facultad de Ingeniería, así como también avance y desempeño en las mismas, entre otras cosas.

La primera etapa del mismo consiste en extraer los datos de la base fuente de Bedelía a través de consultas SQL que Se.C.I.U. proporciona a través nuestro. Los mismos son copiados a diskette y cargados en el datawarehouse.

Para la carga de esos datos se debe usar el software implementado por el grupo de carga del In.Co. [5] el cual se realizó utilizando la herramienta DTS (Data Transformation Server) del Sql-Server7.0, de Microsoft.

Luego se procede a llenar las tablas facts [6] sobre las cuales se basará la definición de los cubos. Para ello se deberá utilizar un procedimiento almacenado definido especialmente para la carga (carga_facts). [Ver especificaciones técnicas – Implementación OLAP]

La siguiente etapa consiste en la construcción de cubos, almacenamiento y procesamiento de los mismos lo cual hicimos usando la herramienta OLAP-Services de Sql-Server 7.0 de Microsoft.

Por último para poder visualizar y analizar los cubos construimos un visualizador genérico de cubos a través de browsers html. Esto se realizó utilizando herramientas Microsoft tales como: Visual Basic 6.0, Visual Interdev.

2 - FUNCIONALIDAD

¿QUÉ PERMITE EL SISTEMA?

El sistema fue diseñado para el análisis de las actividades de los estudiantes y para poder cuantificar su situación en la Facultad así como para dar una idea global a los docentes, Asistentes Académicos e investigadores en el área de la educación con el fin de evaluar planes de estudio, cambios de plan, dictado de cursos, formas de evaluación de los mismos, etc.

Estudio de las actividades por Materia

Uno de los principales aportes de este Taller es la posibilidad de estudiar la situación de los estudiantes por Materia entendiéndose ésta como conjunto de asignaturas agrupadas por área del conocimiento. En los talleres precedentes este análisis solo era posible por asignatura. Esto lleva a poder realizar estudios por área, lo cual brinda una visión más global, más acorde al nuevo enfoque de los planes de estudio, más comprensible y significativa.

Se podrán obtener resultados del estilo: cantidad de estudiantes que tienen por ejemplo las Matemáticas aprobadas/reprobadas, promedio de notas de las Matemáticas, cantidad de estudiantes que exoneraron las Matemáticas, cantidad de estudiantes con un determinado porcentaje de avance en las mismas.

Estudio de las inscripciones por Materia

El sistema brinda la posibilidad de saber la cantidad de estudiantes inscriptos a los cursos de las asignaturas de una Materia, discriminando si es necesario por generación o por año lectivo; de esta forma se logra visualizar la evolución de las inscripciones en el correr de los años.

Estudio por Asignatura

El mismo estudio que se realiza por Materia se puede trasladar a cada Asignatura, obteniendo de esta forma un estudio más detallado. Es decir se podrán analizar inscripciones, exámenes y cursos reprobados, aprobados, promedio de notas, etc. de una asignatura en particular.

Aclaración: Cuando se habla de promedio de notas existen dos promedios, uno que tiene en cuenta solo las actividades aprobadas y otro que toma el total de actividades rendidas.

Estudio del avance por Materia

Permite el análisis de la situación de los *estudiantes dentro de una materia de una carrera*. El estudio brindará la cantidad de estudiantes que se encuentran en un rango de porcentaje de créditos obtenidos. Los rangos variarán de 0% a 10%, de 11% a 70%, 71% a 99% y el 100%. De esta manera se podrá evaluar en un plan de estudio de una carrera el comportamiento de los estudiantes de una determinada generación en las distintas materias, permitiendo determinar las áreas de mayor dificultad.

Por ejemplo, se podrá determinar el avance de los estudiantes de la generación 97 en el plan 97 de la carrera Ingeniería en Computación en las Matemáticas.

Estudio de los exámenes por período en su versión numérica

Se ha incorporado el estudio de los exámenes por período, lo que interesa es ver la información de los mismos desde dos ángulos. La visión numérica del período de examen es aquella que los interpreta como primer, segundo, tercer, etc. período de la asignatura. Entonces se podrá saber entre otras cosas cuántos estudiantes aprobaron/reprobaron el examen de una asignatura en su primer período, el promedio de las notas de los estudiantes que aprobaron el examen en el primer período, etc.

Estudio de los exámenes por período en su versión nombre

Se mantiene la visión del período en su forma de mes es decir aquella que los interpreta como febrero, marzo, julio, agosto, diciembre y extraordinarios. Entonces se podrá saber entre otras cosas cuántos estudiantes aprobaron/reprobaron el examen de una asignatura en diciembre, el promedio de las notas de los estudiantes que aprobaron el examen en marzo, etc.

Estudio de los cursos

Se incorpora un estudio de los cursos un poco más detallado ya que se incluyen no solo los *cursos normales* sino los *cursos invalidados* y *cursos caducados*.

Se entiende por curso invalidado aquel que por una acción de re-inscripción al mismo hace que se pierda la validez del anterior.

Se entiende por curso caducado aquel que una vez aprobado ha perdido su vigencia ya que cumplió con el plazo establecido y el estudiante no ha aprobado el examen.

Las medidas que se pueden obtener son cantidad de estudiantes que aprobaron/reprobaron un curso y promedio de notas de los mismos.

Los cursos invalidados dan una primera aproximación de los estudiantes que están recursando una asignatura, ya que éstos son los que no alcanzaron la nota de exoneración del examen y han optado por recurrar la asignatura. A estos estudiantes habrá que sumarles aquellos que perdieron el curso. De esta forma se obtiene una mejor estimación de los estudiantes que estarían recursando una asignatura.

Si la asignatura tiene una validez de curso al número anterior habría que anexarle los estudiantes con el curso caducado, ya que son potenciales estudiantes destinados a recurrar la asignatura.

Por ejemplo, si queremos tener una idea de cuántos estudiantes están recursando la asignatura "Cálculo 1" debemos sumar los estudiantes que tienen un curso invalidado más los que tienen el curso perdido más los que tienen el curso caducado, cuidando la fecha en la cual se realiza el estudio.

Estudio por Instituto

Se agrega también la posibilidad de efectuar estudios por Instituto dentro de la Facultad, lo que permitirá cuantificar el esfuerzo y la cantidad de recursos que deberá asignar cada uno de ellos destinados a atender un curso o una carrera.

Esto se deducirá a partir de la cantidad de inscriptos a los cursos de las asignaturas que el mismo dicta, también se podrá saber cuántos exámenes ha corregido un Instituto.

El análisis también se podrá filtrar por Instituto y Materia lográndose combinaciones interesantes como ser: cantidad de inscriptos a Electivas que deberá atender el IN.CO o el Instituto de Ing. Eléctrica o el Instituto de Matemática por ejemplo.

Es decir, permitirá medir la variación de la población estudiantil en los cursos a través de los años, se podrá ver si se está atendiendo mayor o menor cantidad de estudiantes con los mismos recursos que antes se usaban para atender menos o más

estudiantes, de esta forma se podrán fundamentar reclamos o pérdida de calidad de los cursos, re-asignación de recursos, etc.

Estudio del avance por Carrera, Plan y Perfil

Incluimos resultados sobre porcentajes de avance de los estudiantes en una carrera determinando quienes están Comenzando (entre el 0 y el 10% de los créditos totales), Medio (entre el 11 y el 70% de los créditos totales), Terminando (entre el 71 y el 99% de los créditos totales) o Egresado (100% de los créditos totales). Este estudio podrá analizarse según plan de la carrera y perfil dentro del mismo.

Con esta facilidad se tiene una idea de cómo avanza el plan de estudio definido y de cuántos estudiantes posiblemente lleguen a recibirse.

Estudio del desempeño por Carrera, Plan y Perfil

Muestra el desempeño de los estudiantes en una carrera de la siguiente manera: estudiantes que abandonaron, son poco activos, activos o muy activos según la cantidad de actividades rendidas en los últimos años. Se podrá obtener entonces cuántos estudiantes están en cada una de estas situaciones. Este estudio podrá analizarse según plan de la carrera y perfil dentro del mismo.

Estos datos permiten saber cuántos estudiantes “activos” o “muy activos” está atendiendo la facultad y permite hacer comparaciones con años anteriores. Ayuda a tener una idea de cómo es el desempeño de los estudiantes de una generación en el transcurso de una carrera, es decir, cuántos estudiantes eran “muy activos” en el comienzo de la carrera y cuántos son “muy activos” al finalizar la misma.

Estudio por Generación

Todo el análisis antes mencionado se podrá discriminar por generación permitiendo de esta forma evaluar el comportamiento de las distintas generaciones en los distintos planes de estudio.

Estudiantes que aprobaron/reprobaron una asignatura y/o reprobaron/aprobaron otras.

Se ha constatado la necesidad de los docentes de saber este tipo de cosas, lo cual no es resoluble por un funcionario de bedelía. Este sistema permite contestar esta clase de preguntas pudiendo variar las asignaturas y limitando hasta cuatro and's/or's, por problemas de visualización.

El resultado de este tipo de consultas a diferencia de las anteriores no es un número únicamente sino que despliega una lista de cédulas de estudiantes que cumplen la condición solicitada.

Esta opción es la que permitirá definir las previas de una asignatura que ya existe, pudiéndose conocer la cantidad de estudiantes que están habilitados para realizar el curso o dar el examen de dicha asignatura. De la misma forma ayuda a determinar las previas de una asignatura nueva en el plan de estudio de una determinada carrera.

Por ejemplo, supongamos que queremos agregar al plan de estudios de una determinada carrera una asignatura “Asig I” y que queremos determinar sus previas para cubrir un cupo de alumnos que puede atender el Instituto, entonces basta con comenzar consultando quienes han aprobado la “Previa I”, la “Previa II” y la “Previa III”, etc e irla modificando hasta lograr lo deseado. Cabe aclarar que el resultado de éste tipo de consulta no incluye a los egresados debido al objetivo de la consulta.

Estudiantes que reprobaron una asignatura una cantidad de veces X.

Así como indica el título se podrán obtener las cédulas de los estudiantes que han reprobado una cierta cantidad de veces una asignatura determinada contando a partir de un año dado.

Este tipo de consulta es útil cuando lo que interesa es medir cuánto cuesta aprobar una asignatura, cual es la cantidad de veces promedio, que los estudiantes rinden un examen pudiéndose comparar los diferentes planes de estudio de una carrera, así como también los distintos docentes que dictan el curso, etc.

Estudiantes que aprobaron todas las asignaturas de un año de una carrera.

Se ha incorporado también este estudio lo que implica un esfuerzo de recolección de datos extra además de los proporcionados por Bedelía. Esta consulta estaría pensada para planes anteriores al 1997 ya para este son más apropiadas las consultas por créditos.

A la fecha de entrega de este taller dichos datos no están disponibles, pero una vez incorporados el sistema está en condiciones de resolver el requerimiento.

Estudiantes inscriptos/egresados en una carrera C y/o inscriptos/egresados en otras.

A partir del relevamiento de talleres anteriores parece interesante tener armadas este tipo de consultas.

Lo interesante de este tipo de resultados es determinar la gente que egresó de más de una carrera, la gente que egresó de una carrera y continúa estudiando, la gente que se inscribió a más de una carrera, etc.

Estudio individual de un estudiante en una asignatura.

El sistema permite un análisis más particularizado, tomando un estudiante y una asignatura se podrá saber generación del estudiante, si aprobó la asignatura, forma de aprobación de la misma (reválida, cambio plan, normal o automática a partir de un curso), nota y fecha de aprobación y en caso de haberla reprobado la cantidad de veces, notas y fechas de reprobación.

OTRAS CARACTERÍSTICAS DEL SISTEMA

- El sistema brinda la posibilidad de visualizar desde todos los ángulos la información disponible en los cubos del datawarehouse con simples “clicks” y por otro lado consultar las tablas del datawarehouse, minimizando el ingreso de información (desde el teclado) por el usuario para satisfacer sus consultas, desplegando las opciones disponibles en cada selección.
- La visualización de los cubos permite el acceso en forma dinámica a la información pudiendo variar la consulta en todo momento agregando o eliminando tanto dimensiones como medidas, filtrar por dimensiones que no se encuentren en la consulta, así como navegar a través de los diferentes niveles jerárquicos de la información.
- La posibilidad de efectuar consultas a las tablas del datawarehouse permitirá escapar del esquema que proponemos para ingresar al mundo del análisis de las

actividades de los estudiantes y deja un espacio para la imaginación de cada consultante el cual con un mínimo conocimiento del lenguaje SQL podrá obtener sus respuestas.

Existen 2 modalidades: consultas prearmadas con cierta flexibilidad en la selección de los datos a visualizar, y por otro lado el ingreso de la consulta en SQL puro, con la facilidad de conocer la estructura de las tablas disponibles.

- También se permite graficar el resultado de las consultas en diferentes formas: 2 Dimensiones barras paralelas, 2 Dimensiones multilínea, 2 Dimensiones barras acumuladas, barras de 3 Dimensiones y gráfica estilo torta. Esta alternativa brinda una forma de visualización de los resultados aún más global y comprensible.
- Se incluye la posibilidad de grabar a disco la página que se está consultando, con el fin de reanudar el análisis más adelante y continuar el estudio a partir del punto desde cual se almacenó.
- En todo momento el usuario dispone de una ayuda en línea de fácil comprensión, diseñada manteniendo la estructura de la página de trabajo, con la diferencia que al clickear sobre cada elemento se despliega su explicación. Por ejemplo se podrá consultar sobre lo que se entiende por dimensión, medida, filtrado, etc.; lenguaje tal vez desconocido para un usuario que no está en el área del OLAP y que creemos son los que más necesitan los datos, pero que rápidamente y sin tener que leer demasiado comprenderá y podrá investigar sin dificultad.
- El sistema permite una carga semi-automática de los datos fuente al datawarehouse a través de un programa desarrollado como dijimos antes por docentes del In.Co. [5].
- Permite un fácil mantenimiento de los cubos cuando los datos fuente varían, usando simplemente un modo Refresh en el momento del procesamiento.
- Si a lo largo del uso del sistema se viera que el modo de almacenamiento de los cubos elegido por este Taller no fue el apropiado, fácilmente en el mantenimiento de los cubos se podrá cambiar el modo sin tener que modificar la definición de los mismos. Asimismo podrá actualizarse la optimización basada en uso realizada por este Taller con la facilidad que provee el OLAP Services y de esta manera lograr una mejor performance en las consultas.

Resumen de las preguntas que le ayuda a contestar el sistema

Entre ellas tenemos:

- C.I., nota y forma de aprobación de los estudiantes que aprobaron todas las asignaturas del año X, en una carrera dada.
- Estudiantes que cumplieron X créditos de una materia y/o Y créditos en otra y/o
- Lista de asignaturas de un Instituto Y que fueron dictadas en los últimos N años.
- Estudiantes que rindieron más de N veces el examen de una asignatura desde el año X.
- ¿Cuántos estudiantes aprobaron el curso de una asignatura en un año X?
- ¿Cuál es el promedio de notas de aprobación del curso de una asignatura en una fecha, para una generación?
- ¿Cuántos estudiantes exoneraron el examen de una asignatura en el período de Julio?
- ¿Cuántos exámenes corrigió un Instituto de una asignatura en un período X?
- ¿Cuántos estudiantes aprobaron el examen de una asignatura en el primer período?
- ¿Cuál es el promedio de notas de aprobación de los exámenes de una asignatura en un período X?
- ¿Cuántos estudiantes tienen entre el 11% y el 70% de las matemáticas aprobadas en una carrera X?
- ¿Cuántos estudiantes han abandonado una carrera X en un año Y?
- ¿Cuántos estudiantes son Muy Activos en una carrera X en un año Y?
- ¿Cuántos estudiantes están inscriptos al curso de una asignatura X en un año Y?
- ¿Cuántos estudiantes debe atender un Instituto X en un año Y?

EXTENSIONES DEL PROYECTO.

Sobre la base del resultado de este proyecto -sumándole las mejoras que más adelante proponemos, las cuales requieren un relativo esfuerzo-, el “Sistema Gerencial para Bedelía” podría ser instalado en cualquier Facultad de la Universidad de la República aprovechando el hecho de que el datawarehouse se basa en la BD de Gestión de Bedelía que tiene la misma estructura para todos los centros educativos.

Es de destacar también que la carga y definición del datawarehouse, la estructura de los cubos y el browser están implementados en forma genérica lo cual permite su re-utilización.

De esta forma se lograría que cualquier docente con un PC conectado a la red de su Facultad con un navegador estaría en condiciones de analizar: cursos, exámenes, egresos, correcciones por instituto, etc. de su Facultad.

Otro de las extensiones del proyecto es que si se continúa el estudio de los talleres previos que analizan la realidad de Presupuesto de la Facultad habría que ocuparse de la definición del datawarehouse y cubos y no habría que dedicar esfuerzo a la construcción de un visualizador para los datos de Presupuesto, se podría utilizar el que surgió de este proyecto.

3 - DISEÑO

TECNOLOGÍAS USADAS.

El proyecto estuvo basado en tecnologías Microsoft en casi su totalidad.

Para la extracción de los datos de la base de Gestión de Bedelía la cual se encuentra sobre una plataforma cliente-servidor conformada por:

sistema operativo (servidor): Solaris Intel 5.1.

manejador de BD: Oracle 8

clientes con Windows NT Workstation 4.0.

se usaron consultas SQL [Ver Especificaciones Técnicas – Archivos para la recolección de los datos desde Bedelía] que se corren desde un cliente cualquiera de la bedelía, usando un usuario habilitado por el sistema. El resultado de las mismas son archivos ascii los cuales son copiados a diskette y trasladados al servidor del datawarehouse.

Plataforma del proyecto:

- Servidor Windows NT Server 4.0 Enterprise Edition con Service Pack 4.0
- Internet Information Server (IIS)
- Sql-Server 7.0 Enterprise Edition.

En el proyecto se usaron de diferente forma las herramientas anteriores:

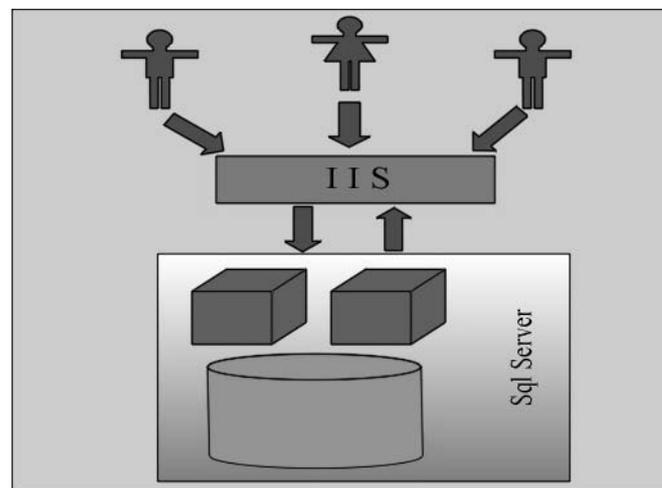
- Para la carga del datawarehouse se utilizó el Data Transformation Server, herramienta que viene incluida con el Sql-Server 7.0. Por más detalles sobre la misma referirse a [5].
- El datawarehouse está implementado como una base de datos relacional construida sobre Sql-Server 7.0. [Ver especificaciones técnicas – Tablas del DataWarehouse]
- La definición y procesamiento de los cubos se realizó con el OLAP Services de Sql-Server 7.0.
- La carga de las tablas fact la implementamos mediante la ejecución de un procedimiento Transac-SQL de Sql-Server 7.0., en las que se basan los cubos definidos.[Ver especificaciones técnicas – Implementación OLAP]
- El visualizador html de cubos fue desarrollado como una aplicación ASP (Active Server Page) con Microsoft Visual Interdev y Visual Basic 6.0. El código del servidor se programó en Visual Basic Script junto con objetos ADO (ActiveX Data Object) para el acceso a los datos, mientras que para el cliente se utilizó JavaScript y HTML standard. Una breve justificación del motivo por el cual se decidió usar JavaScript para la secuencia de comandos del cliente se encuentra en el siguiente cuadro:

		Javascript	Aplic. IIS	DHTML	Pivot Table
W I N D O W S	Netscape 3.x	Sí	No	No	No
	Netscape 4.x	Sí	No	No	No
	Netscape 4.7	Sí	Sí	Sí	No
	I Explorer 3.0	Sí	No	No	No
	I Explorer 4.x	Sí	No	No	No
	I Explorer 5.x	Sí	Sí	Sí	Sí
U N I X	Netscape 3.x	Sí	No	No	No
	Netscape 4.x	Sí	No	No	No
	Netscape 4.7	Sí	No	No	No

- La facilidad de visualizar en diferentes formas gráficas los datos presentados en la grilla se programó usando el objeto MSChart de Visual Basic 6.0 aprovechando su interacción con las páginas ASP.

ARQUITECTURA DEL SISTEMA.

La siguiente figura muestra los diferentes componentes del sistema:



- **Capa inferior:** que está formada por la base de datos relacional del datawarehouse y la definición multidimensional.
- **Capa intermedia:** conformada por el servidor web que estará atendiendo las solicitudes de los usuarios y que será el nexo entre el datawarehouse y el browser.
- **Capa superior:** formada por la interfase html del sistema, esta es la capa más cercana al usuario final.

DISEÑO RELACIONAL DEL DW.

A partir de las preguntas que queríamos contestar definimos 6 cubos cuya definición se basa en la siguiente base relacional:



Tablas del datawarehouse:

BD_Estudiantes: Contiene todos los estudiantes de la facultad, estén egresados o no. Allí se guardan datos personales de los mismos.

¿Qué es un estudiante?

Es toda persona que se inscribió a una/s carrera, podrá estar egresado, activo, suspendido.

BD_Carreras: Tiene la definición de las distintas carreras que dicta o dictó la facultad, incluye plan y perfil.

¿Qué es una carrera?

Es la rama de conocimiento sobre un mismo tema y que puede diferir en los distintos planes y perfiles.

BD_Materias: Guarda todas las materias que existen en Facultad.

¿Qué es una materia?

Es una agrupación de asignaturas que tratan sobre un mismo tema.

BD_Asignaturas: Contiene todas las asignaturas de la Facultad, las que dictan y las que se dictaron alguna vez.

BD_Institutos: Mantiene la definición de los distintos Institutos dentro de la Facultad y fuera también.

BD_Actividades: Resume las actividades de los estudiantes en la facultad, agrupadas por materia.

¿Qué es una actividad?

Se entiende por actividad: un curso, un examen, un curso caducado, un curso inválido. Es todo lo que ha hecho un estudiante dentro de la facultad en cualquier año, en cualquier carrera.

BD_Est_Carr: Tiene las inscripciones a carrera que cada estudiante a realizado en la Facultad.

BD_Mat_Carr: Almacena la información relacionada con las materias que se dictan en cada carrera.

BD_Egresos_Títulos: Aquí se guardan los egresos por estudiante y carrera de toda la Facultad.

BD_Desempeño: Esta tabla contiene el cálculo de la situación de un estudiante dentro de la carrera en el año en que se realizó el análisis.

BD_Avance: Mantiene el cálculo del avance de un estudiante dentro de una materia en una carrera a la fecha en que se realizó el análisis.

BD_Año_Asignatura: Indica en que año se dicta una asignatura en una carrera.

BD_Año_Dictada: Indica desde qué año calendario se dicta una asignatura en la Facultad.

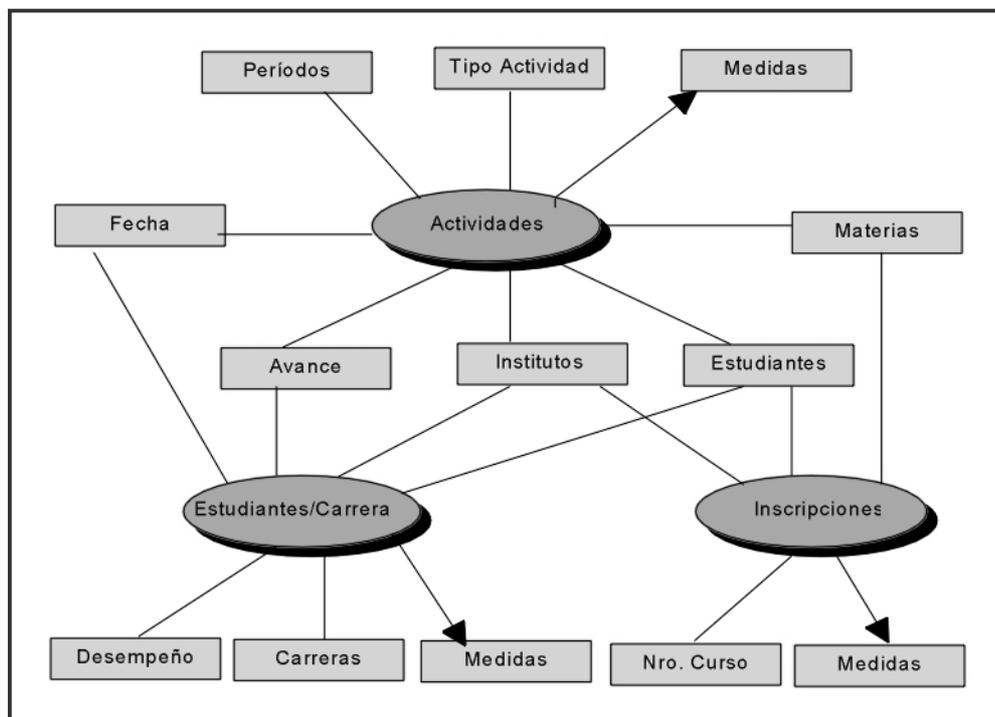
BD_Año_Cantasi: Indica la cantidad de asignaturas que se deben aprobar en un año de una carrera.

Por información más detallada sobre las tablas remitirse a las especificaciones técnicas – Tablas del DataWarehouse.

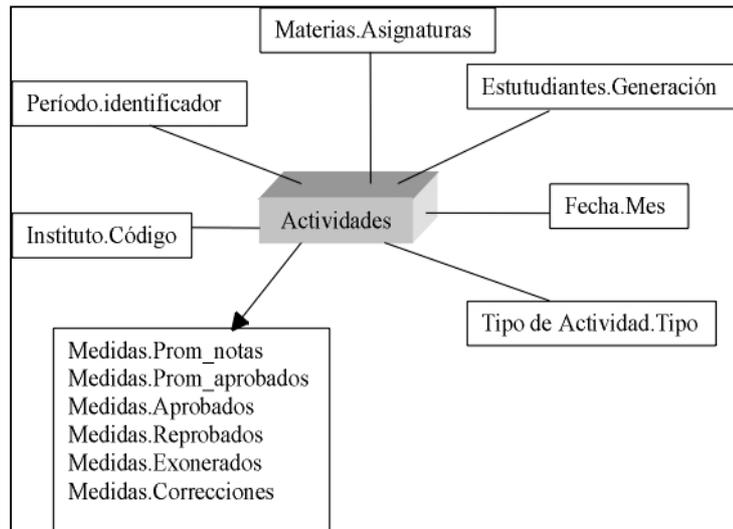
DISEÑO MULTIDIMENSIONAL.

Diseño Conceptual.

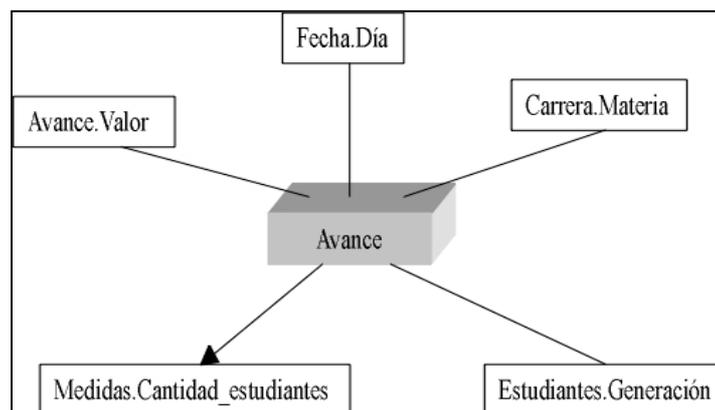
A partir del análisis de la realidad y utilizando el lenguaje de especificación propuesto por Fernando Carpani [7] el esquema multidimensional del sistema es:



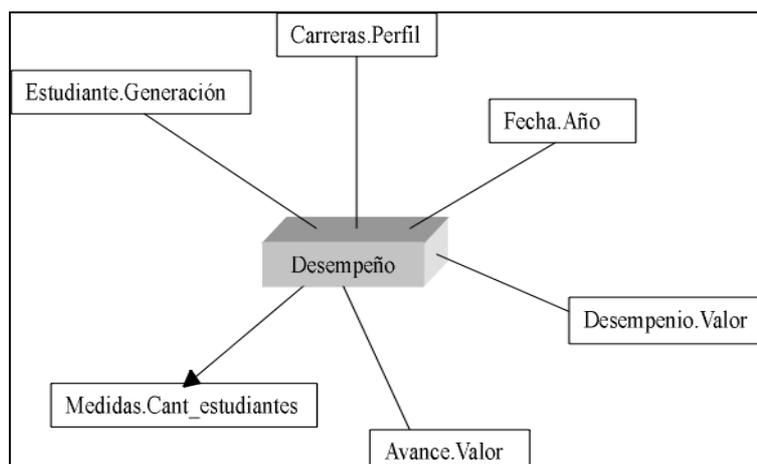
De este esquema nos interesan los siguiente cubos:



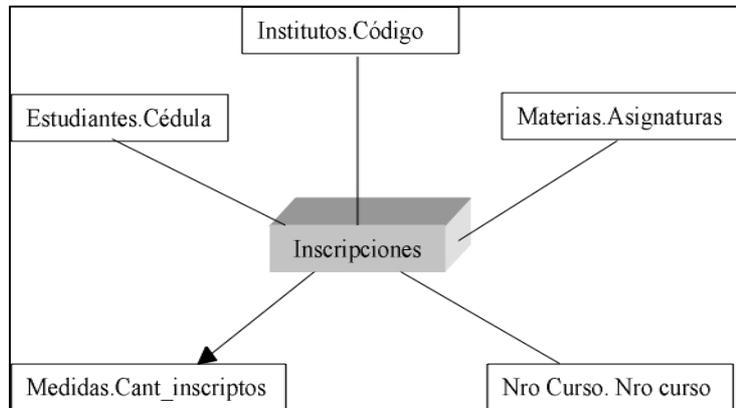
Este cubo permite el análisis de las actividades de los estudiantes dentro de las distintas materias/asignaturas desde diferentes perspectivas, dadas por los cruces de las dimensiones que interesan para el estudio. Las medidas que se permite analizar resultaron del relevamiento realizado por el Taller 1998.



El cubo Avance permite el análisis de la situación de los *estudiantes dentro de una materia de una carrera*. El estudio brindará la cantidad de estudiantes que se encuentran en un rango de porcentaje de créditos obtenidos. Los rangos variarán de 0% a 10%, de 11% a 70%, 71% a 99% y el 100%.



El cubo Desempeño permite el estudio de la situación de los *estudiantes dentro de una carrera* de la Facultad. El análisis brindará la cantidad de estudiantes que son Muy Activos, Activos, Poco Activos o Abandonaron. También permite el estudio de la situación de los estudiantes que están Comenzando, Medio, Terminando o Egresado.



El cubo Incripciones permite obtener la cantidad de estudiantes que se inscribieron al/los curso/s de una materia/asignatura en un año dado. El análisis también se podrá realizar por Instituto que dicta la asignatura.

Diseño Lógico.

Tablas Fact del sistema:

Para crear los cubos anteriores fueron necesarias crear 4 tablas fact auxiliares dado que resultaron 6 cubos a implementar:

- Activ_cursos
- Activ_periodo_numerico
- Activ_periodo_nombre
- Avance
- Desempenio
- Incripciones

FACT_ACTIVIDADES	FACT_ACTIVIDADES1
ES_CI	ES_CI
MA_CODMAT	MA_CODMAT
AS_CODAS	AS_CODAS
IN_CODINST	IN_CODINST
ES_GENERACION	ES_GENERACION
AC_PERIODO	AC_NOMPERIODO
AC_ANIO	AC_ANIO
AC_FECHA	AC_FECHA
ME_PROM_NOTA	ME_PROM_NOTA
ME_PROM_APROB	ME_PROM_APROB
ME_CANT_APROB	ME_CANT_APROB
ME_CANT_PROM	ME_CANT_PROM
ME_CANT_EXON	ME_CANT_EXON
ME_CANT_CORREC	ME_CANT_CORREC

FACT_ACTIVIDADES2	FACT_DESEMPEÑO
ES_CI	CC_CODCARR
MA_CODMAT	CC_PLAN
AS_CODAS	CC_PERFIL
IN_CODINST	EC_FECHAINGR
ES_GENERACION	ES_GENERACION
AC_TIPOACTIVIDAD	DE_ANIO
AC_ANIO	DE_DESEMPEÑO
AC_FECHA	DE_AVANCE
ME_PROM_NOTA	ME_CANT_EST
ME_PROM_APROB	
ME_CANT_APROB	
ME_CANT_PROM	
ME_CANT_CORREC	

Dado que la herramienta (OLAP Services), como dijimos antes, se basa en un esquema estrella cada cubo debe ser construido a partir de una tabla fact asociada por lo tanto son necesarias las siguientes tablas fact:

Fact_Actividades1: Resume la información necesaria para la definición del cubo Activ_Período_Nombre. Contiene datos fuente más pre-cálculos sobre las actividades examen que ayudarán en la definición de las medidas del cubo.

Fact_Actividades2: Resume la información necesaria para la definición del cubo Activ_Cursos. Contiene datos fuente más pre-cálculos sobre las actividades curso que ayudarán en la definición de las medidas del cubo.

Fact_actividades: Resume la información necesaria para la definición del cubo Activ_Período_Numerico. Contiene datos fuente más pre-cálculos sobre las actividades examen que ayudarán en la definición de las medidas del cubo.

Fact_Desempenio: Resume la información necesaria para la definición del cubo Desempenio. Contiene datos fuente más pre-cálculos sobre la actuación de los estudiantes en una carrera que ayudarán en la definición de la medida del cubo.

Por más información referirse a [Especificaciones Técnicas – Implementación OLAP]

DISEÑO DE LA INTERFASE.

La interfase es la capa superior del sistema, la más cercana al usuario final, (ver Arquitectura del Sistema) y esta formada por dos componentes:

Por un lado la Componente del Servidor que es la encargada de recibir las peticiones del cliente a través de IIS (Internet Information Server) para que nuestra aplicación proceda a generar el código de las consultas SQL o MDX según corresponda, conectarse a la base, ejecutar la consulta y finalmente crear las páginas que se envían al cliente.

Por otro lado la Componente del Cliente permite al usuario, a través de un browser, visualizar la información contenida en el código HTML junto con código Java Script que controla y copia a campos ocultos información que permiten las distintas funcionalidades del sistema. Mediante envío de formulario o pasaje de parámetros estos datos viajan al servidor, para que éste consulte la base y responda la petición.

En este contexto se diseñaron un conjunto de páginas ASP y HTML que llevan a cabo las tareas mencionadas anteriormente cuya organización y estructura trataremos a continuación.

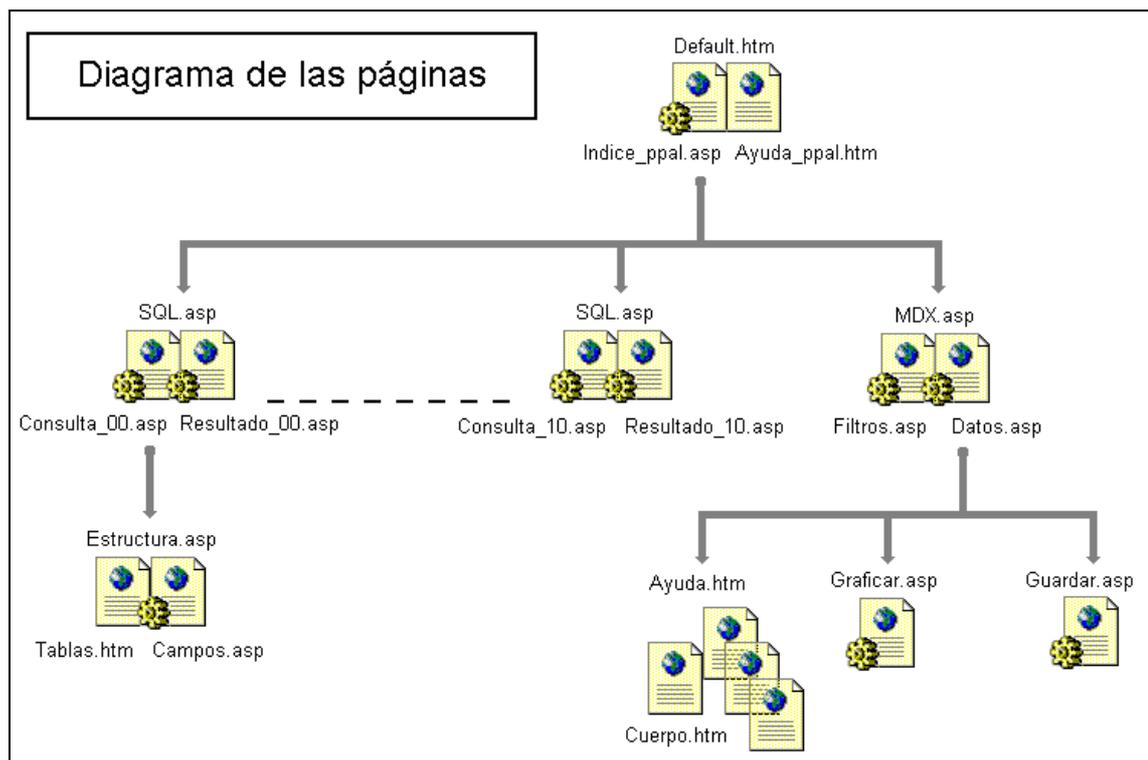


Figura - Diagrama de las páginas

Desde la página inicial se accede a dos subconjuntos de páginas:

- Consultas SQL – Es un conjunto de diez páginas (Consulta_nn.asp) con similar estructura que siguen un esqueleto de consultas pre-establecido, además de una página (Consulta_00.asp)-consulta libre- desde la que se puede acceder a la visualización del esquema de las tablas del datawarehouse(Estructura.asp).

- Análisis de cubos – A este grupo se accede a través de MDX.asp conformada por Filtros.asp y Datos.asp las cuales son refrescadas dinámicamente ante cualquier cambio en la consulta. Desde esta última se llama a Guardar.asp, Graficar.asp y Ayuda.htm.

Diseño detallado.

A continuación se detalla cada una de las páginas componentes del sistema:

Default.htm

Es la página de inicio del sistema. Esta compuesta por frames, uno de ellos es *Indice_ppal.asp* que contiene la consultas MDX iniciales (no configurables por el usuario) y despliega las opciones disponibles para acceder a los datos en sus dos modalidades, junto con links que acceden a una breve explicación de lo que hace la opción, lo cual se muestra en el marco inferior de esta página. Este último es *Ayuda_ppal.html*.

Interacción con las demás páginas: (envío de formulario) mediante links con *MDX.asp* y con *Consulta_nn.asp*

SQL_00.asp

Es la página que permite el ingreso libre de consultas SQL. Se presenta como un frame compuesto por: *Consulta_00.asp* y *Resultado_00.asp*.

Consulta_00.asp es donde se despliega el área de texto necesaria para escribir la consulta junto con links para ejecución, borrado e información de la base. Con éste último se abre una nueva ventana donde se puede acceder a los nombres de las tablas que constituyen el datawarehouse y la estructura de cada una de ellas. El motivo de abrir una nueva ventana, es dar la posibilidad al usuario de tener en pantalla tanto la consulta que escribe como las estructuras a consultar.

Tablas disponibles:		Estructura de BD_ACTIVIDADES		
BD_Actividades		Campo	Tipo	Nulo?
BD_Anio_Asignatura		ES_CI	int (4)	No
BD_Anio_Cantasig		MA_CODMAT	varchar (6)	No
BD_Anio_Dictada		AS_CODAS	varchar (5)	No
BD_Asignaturas		AC_FECHA	datetime (8)	No
BD_Avance		AC_CURRI	varchar (1)	No
BD_Carreras		AC_TIPOACT	varchar (1)	No
BD_Desempenio		ES_GENERACION	varchar (20)	No
BD_Egresos_Titulos		AC_PERIODO	varchar (10)	Sí
BD_Est_Carr		AC_NOMPERIODO	varchar (15)	Sí
BD_Estudiantes		AC_TIPOACTIVIDAD	varchar (20)	No
BD_Insecur		AC_TIPORESULTADO	varchar (10)	No
BD_Institutos		AC_NOTA	int (4)	No
BD_Mat_Carr		AC_CREDITOS	int (4)	Sí
BD_Materias		AC_REVALIDA	int (4)	Sí
		AC_ANIO	int (4)	No
		AC_APRUEBAAS	varchar (1)	No
		AC_TIPOGEN	varchar (1)	No
		IN_CODINST	int (4)	No
		BD_ACTIVIDADES tiene 18 campos.		

Resultado_00.asp es la otra página que compone el frame donde se despliega el resultado en un formato de tabla que se crea dinámicamente de acuerdo con la consulta. También se muestra la cantidad de registros seleccionados.

Interacción con las demás páginas: mediante un link vuelve a Default.htm.

SQL_nn.asp

Mostrar:
los estudiantes que aprobaron todas las asignaturas de <input type="text"/> año
de la carrera <input type="text" value="3-INGENIERIA ELECTRICA"/> del plan <input type="text" value="Plan87-Carr20-8"/> del perfil <input type="text" value="Carr. 2 Perfil 1 - PERFIL UNICO"/>
Ejecutar Consulta Borrar Criterio Página Principal
<p>Elija criterio de selección.</p>

La estructura de estas páginas sigue el mismo criterio utilizado en SQL_00.asp.

La diferencia con la anterior está en que Consulta_nn.asp hay una secuencia de listas de selección y campos de texto insertas en una sencilla oración donde se solicita los datos

a visualizar. Todas las listas de selección contienen la totalidad de los valores disponibles en la base, esto minimiza el error al momento de consultar.

Interacción con las demás páginas: idem a SQL_00.asp.

MDX.asp

The screenshot shows the MDX.asp interface with the following components and data:

Filters: Fecha (Sin Filtrar), Tipo Actividad (Sin Filtrar), Estudiantes (Sin Filtrar), Institutos (Sin Filtrar)

MEDIDAS: AGREGAR A: [Nueva Medida] | QUITAR DE: [Medida]

DIMENSIONES: Nueva Dimension | Dimension

Frame de Filtrado: Tipo Actividad > Curso

	[Reprobados]	[Aprobados]
Materias > COMPUTACION +	0,00	2
Materias > ELECTIVAS +	0,00	0,00
Materias > FISICAS +	0,00	0,00
Materias > MATEMATICAS +	0,00	1

Tabla de Resultado

Criterio de filtrado (Cubo: *activ_cursos*): NINGUNO

Esta página es el “motor” del sistema visualizador de cubos. También es un frame que contiene a Filtros.asp en la parte superior y Datos.asp en la inferior. En este caso se decidió la implementación de un frame por razones de eficiencia, ya que existe un costo elevado para mostrar las posibilidades de filtro al usuario. Veamos cada una de las páginas en detalle:

Filtros.asp

Contiene listas de selección, una para cada dimensión, que tienen todos los valores disponibles para el filtro. Éste se calcula únicamente la primer vez, por eficiencia, y los datos se obtienen de la Metadata, lo que da independencia de la base con la que se trabaja. El sistema controla que el filtrado se realice por dimensiones que no se encuentren en la consulta, también cuando se solicita el filtrado controla que el criterio elegido haya variado, para impedir la reiteración de la consulta inútilmente.

Interacción con las demás páginas: en forma directa y única con Datos.asp copiando en campos ocultos de ésta la información de filtrado seleccionada por el usuario. Por otro lado, Datos.asp actualiza las listas de selección de Filtros.asp en algunas circunstancias que veremos más adelante.

Datos.asp

En esta página el usuario dispone de todas las funcionalidades para el análisis de cubos y también se despliega la grilla resultado con los datos. En definitiva es quien “arma” y ejecuta las consultas MDX.

Para agregar o eliminar tanto dimensiones como medidas (operaciones que se hacen desde el tablero de control) existe un tablero “imitando” el aspecto de la tabla resultado. Dicha imitación colabora en la formación de la nueva página, la cual permitirá agregar o quitar las dimensiones y medidas que correspondan a la situación planteada.

También existe una barra de herramientas desde la cual se puede volver a la página inicial, graficar, ver texto de la consulta, cambiar modalidad, etc.

- ¿Qué significa cambiar modalidad?
Se implementaron dos modalidades para visualizar el drilldown: en profundidad y por nivel.
Cuando se hace un drilldown por nivel se verá el nivel siguiente para todos los miembros del nivel actual.
Cuando se hace un drilldown en profundidad se verá el siguiente nivel solamente para el miembro elegido del nivel actual.

		[Reprobados]	[Aprobados]
Materias > COMPUTACION -		2	3,00
Materias > COMPUTACION > C102-LOGICA		2	1,00
Materias > COMPUTACION > TAL1-TALLER 1		0	2,00
Materias > COMPUTACION > TAL2-TALLER 2		0	0,00

Ejemplo de visualización "en profundidad".

		[Reprobados]	[Aprobados]
Materias > COMPUTACION -		2	3,00
Materias > COMPUTACION > C102-LOGICA		2	1,00
Materias > COMPUTACION > TAL1-TALLER 1		0	2,00
Materias > COMPUTACION > TAL2-TALLER 2		0	0,00
Materias > ELECTIVAS -		0	1,00
Materias > ELECTIVAS > C501-TEC. AVANZADAS		0	0,00
Materias > ELECTIVAS > C503-PROG. FUNCIONAL		0	0,00
Materias > ELECTIVAS > C518-COMPUTACION GRAFICA		0	1,00
Materias > FISICAS -		0	0,00
Materias > FISICAS > FIS1-FISICA 1		0	0,00
Materias > FISICAS > MEC2-MECANICA 2		0	0,00
Materias > MATEMATICAS -		4	4,00
Materias > MATEMATICAS > MAT1-ANALISIS 1		4	3,00
Materias > MATEMATICAS > MAT2-ANALISIS 2		0	1,00

Ejemplo de visualización "por nivel".

Los datos a analizar se muestran en una grilla que puede tener una o más dimensiones en cada eje y en el cruce de ellas se encuentran los valores almacenados en el datawarehouse. Siempre debe existir por lo menos una medida, las medidas deberán estar ubicadas en un mismo eje de la grilla. A través de los nombres de las dimensiones se puede hacer drilldown o drillup según se muestre “+” o “-“ respectivamente, por otro lado si el nombre no se ve como link indica la imposibilidad de realizar estas operaciones.

Interacción con otras páginas:

- actualiza “combos” de filtrado cuando se agrega a la consulta alguna dimensión por la cual se tenía filtro
- tiene link a la página inicial
- abre nueva ventana para Ayuda.html. La idea de nueva ventana es “tenerla a mano” sin necesidad de pedirla cada vez al servidor.
- abre nueva ventana para Graficar.asp. La razón de otra ventana es poder visualizar de una forma más clara tanto datos como gráfico. En este caso envía como parámetros el tipo de gráfica y el texto de la consulta.
- abre nueva ventana para Guardar.asp. Aquí la explicación está en que mediante la funcionalidad del navegador de guardar páginas, no se pueden guardar frames, por lo que transformamos la información disponible en Datos.asp en una página HTML que sí puede guardarse con el browser. En este caso también hay pasaje de parámetros.

Ayuda.html

Es un frame que contiene a Cuerpo.html (imita el ambiente de análisis de cubos) y un conjunto de páginas HTML con una breve explicación de cada funcionalidad que se van colocando en el otro marco. Dicha explicación se obtiene clickeando en el signo de interrogación que se encuentra al costado de cada elemento de la página ejemplo o en cualquier otro elemento de la misma.

Ayuda para la Visualización de Cubos

Institutos (Sin Filtrar) ▾
Carreras (Sin Filtrar) ▾

MEDIDAS ¿?		DIMENSIONES ¿?				
AGREGAR A:	[Nueva Medida] ▾	Columnas	[Nueva Dimension] ▾	Columnas	[Nueva Dimension] ▾	Filas
QUITAR DE:	[Medida] ▾	Columnas	[Dimension] ▾	Columnas	[Dimension] ▾	Filas

		Estudiantes>Gen 99 -		Estudiantes>Gen 99>123 ¿?	
		[Reprobados] ¿?	[Aprobados] ¿?	[Reprobados] ¿?	[Aprobados] ¿?
Materia>TALLER+	Periodo>1992-	0,00	0,00	0,00	0,00
	Periodo>1992>Julio+	0,00	0,00	0,00	0,00
Materia>ELECT.+	Periodo>1992-	0,00	0,00	0,00	0,00
	Periodo>1992>Julio+	0,00	0,00	0,00	0,00

Usted verá aquí lo que temenos para explicarle. Todos los botones, listas desplegables, hipervínculos y "¿?" le darán información.

Graficar.asp

Es una página que despliega en otra ventana el gráfico con los datos de la grilla en el estilo seleccionado (2D barras paralelas, torta, etc.).

Dicha página recibe como parámetros la consulta y el estilo de gráfica seleccionado.

Luego se invoca una dll (graficar.dll) que se encarga de generar en un directorio determinado el archivo .gif que contiene el gráfico el cual será mostrado por la página.

Guardar.asp

Es una página con la grilla resultado (HTML) para poder guardar el resultado del análisis y retomarlo desde este punto cuando se desee. Para ello hay un link a MDX.asp pasándole el texto de la consulta para recalcularla y volver a trabajar.

Pagina generada el 30/03/00 a las 21:32:30.

Hoja de Trabajo sin conexión

(Para Guardar esta página utilice su navegador: Archivo/Guardar como... (Tipo HTML))

Criterio de filtrado (Cubo: *activ_cursos*): NINGUNO

[Retomar Conexión Página Principal](#)

	Tipo Actividad > Curso	
	[Reprobados]	[Aprobados]
Materias > COMPUTACION +	0,00	2
Materias > ELECTIVAS +	0,00	0,00
Materias > FISICAS +	0,00	0,00
Materias > MATEMATICAS +	0,00	1

4 - IMPLEMENTACION

IMPLEMENTACIÓN.

Visión general

Nuestra idea en el transcurso del proyecto fue tratar de usar la tecnología Microsoft integrada en sus diferentes formas para implementar nuestras soluciones.

Es así que usamos, desde el sistema operativo Windows NT Server 4.0 Enterprise Edition y el Sql-Server 7.0 Enterprise Edition hasta el Visual Basic6.0 para programar una dll, Visual Interdev para programar las páginas de la interfase, utilizando Visual Script para desarrollar código de servidor, HTML y JavaScript en el cliente. Además en algún momento estuvimos investigando la conexión del Sql-Server7.0 con Microsoft Excel 2000, elementos que luego no pudimos utilizar por la diversidad de plataformas existente en Facultad.

Cabe destacar que las sentencias de los lenguajes utilizados en el cliente fueron elegidas de forma tal que pueda ser interpretada correctamente por cualquier navegador que soporte JavaScript.

Interfase Web

Para la implementación tomamos como guía un ejemplo existente en la Ayuda Online de Microsoft (MSDN) [12], conjuntamente con otro encontrado en Internet [4]. Mientras que en el primero de los casos se trata de un área de texto donde el usuario ingresa el código de la consulta MDX que desea y se despliega el resultado, en el segundo se permite seleccionar sólo dos dimensiones y despliega los datos resultantes de la consulta, permitiendo visualizar solamente de a una medida a la vez.

A partir de estos elementos y de características observadas en otros productos existentes en esta materia, buscamos la manera de aumentar la cantidad de dimensiones presentes en el cuadro, incluir medidas, implementar drillup/drilldown, implementar la doble modalidad de visualización, filtros, etc. Se buscó generalizar todas las funcionalidades basándonos en los metadatos para darle mayor independencia a la interfase.

En nuestro sistema, además, partimos de una consulta inicial fija para cada cubo, donde las partes componentes del texto de la misma están almacenadas en IndicePpal.Asp, para facilitar el acceso del usuario a cada cubo.

Acceso a datos

Aquí el problema no se limitaba solamente a obtener y desplegar información de los cubos, sino que también debíamos integrar a los datos algunas funcionalidades. Tal es el caso del "drill", el cual debía realizarse con el mínimo esfuerzo posible para el usuario, es decir que él, con un click sobre el cabezal de una fila o columna (donde se encuentran los nombres de los elementos pertenecientes a la dimensión) llevara a cabo dicha operación.

Para implementar esto, nuestro primer paso estudiar la estructura y propiedades de los datos que se obtienen como respuesta de las consultas multidimensionales. Una vez analizado este aspecto buscamos la forma general que tendrían las consultas para cumplir con nuestro objetivo

Para acceder a los datos utilizamos objetos ADOMD (ActiveX Data Object MultiDimensional). Con estos objetos la forma de obtener el resultado de una consulta multidimensional sobre una base determinada, es definiendo un objeto del tipo *CellSet* al cual, por lo menos, se le debe asignar valor a dos propiedades que son: *StrSource* que es el texto de la consulta y *ActiveConnection* en el que se determina la conexión, definiendo previamente el objeto *Catalog*. A tales efectos se utiliza nombre de servidor, nombre de la base y proveedor.

Para manipular la información almacenada en los cubos utilizamos el lenguaje de consulta MDX (Multidimensional Database eXpression), que tiene cierta semejanza con lo que es *SQL*. La estructura sintáctica genérica de una consulta MDX es [Especificaciones Técnicas - Consultas MDX]:

```
SELECT {...} ON COLUMNS,  
        {...} ON ROWS,  
FROM cubo  
WHERE {...}
```

De esta forma obtenemos como resultado el objeto *CellSet* que puede verse como una matriz donde se encuentran los miembros seleccionados para cada dimensión y/o medida ya sea en filas o columnas, y los valores (datos) correspondientes en el cruce de ambas.

Este objeto tiene múltiples propiedades como ser nombre de miembros, nombres de dimensiones, cantidad de hijos, nivel de jerarquía de un miembro, indica si un miembro fue expandido o no, etc. Información que fue de gran utilidad al momento de implementar.

También se utilizaron algunas funciones de MDX que facilitaron la tarea, como ser:

- “*Hierarchize*”: que conserva el orden jerárquico de los datos
- “*Union*”: operador clásico de unión de conjuntos. Notación: +
- “*Crossjoin*”: realiza el producto cartesiano de dimensiones y/o medidas.
Notación: *
- “*DrilldownLevel*”: Devuelve un conjunto de elementos, formado por todos los miembros de un nivel dado, de la jerarquía, y todos los "hijos" de cada uno de ellos

Más allá de la forma de visualización, la consulta fue pensada para mostrar como máximo dos niveles jerárquicos de los datos a la vez. Esto permite mayor claridad al usuario y manejo de menor volumen de información. Además, para implementar ambas modalidades el sistema genera simultáneamente las dos consultas pero al momento de ejecutar sólo lo hace con la correspondiente a la modalidad elegida.

Obteniendo los datos de la forma que mencionamos anteriormente, en el cellset encontramos los datos ordenados jerárquicamente, además de la información que necesitamos para implementar drillup y drilldown. [Especificaciones Técnicas - Consultas MDX]

Para el caso de las consultas *SQL* a las tablas del Datawarehouse, el resultado se obtiene mediante el uso del objeto *Recordset* (objeto ADODB), al cual se le debe asignar valor a dos propiedades: *Source* que es el texto de la consulta y *ActiveConnection* en el que se determina la conexión, definiendo previamente el objeto *Connection* al cual se le debe indicar el nombre y tipo de una conexión definida en el *Administrador de fuentes de datos de ODBC*.

Existen algunas diferencias entre el ambiente multidimensional y el acceso a la información de las tablas del Datawarehouse:

	Relacional	Multidimensional
Texto de la consulta	Estática	Dinámica
Armado de consulta	Solo valores de los parámetros del where	Toda la consulta
Creación de listas desplegadas	Acceso "puntual" a la información	Acceso a "toda" la información, a través de los metadatos
Manipulación de la tabla resultado	Ninguna	Permite drillup, drilldown, Intercambio de ejes, etc.
Visualización gráfica de la tabla resultado	No corresponde, casi siempre el resultado es un listado de estudiantes.	Si, se puede porque los datos son cuantificadores.

Criterios generales de implementación

El principal tema a resolver es el manejo del texto de las consultas de forma clara, dinámica y cómoda. Para ello tenemos las partes de la consulta que realmente nos interesan en campos invisibles para el usuario (los que llamamos campos ocultos), lo que facilita su identificación y manejo, que se mantienen en toda la aplicación viajando con las distintas modificaciones que vayan realizando los usuarios; a saber:

Nombre	Descripción
str_col_dim	Dimensiones de las columnas
str_col_med	Medidas de las columnas
str_fil_dim	Dimensiones de las filas
str_fil_med	Medidas de las filas
str_where	Condición de "where"
str_cubo	Nombre del cubo

Para el caso de la consulta en profundidad es análogo [Especificaciones Técnicas - Campos No Visibles].

Así mismo cuando el usuario selecciona alguna operación, en el cliente se dispara una función que copia al campo correspondiente el nombre de la dimensión a agregar, o quitar, o por la cual se quiere hacer drill, etc [Especificaciones Técnicas - Bibliotecas JavaScript]. De esta forma la página ASP recibe el formulario conteniendo los campos que componen el texto de la consulta y campos auxiliares para determinar la operación seleccionada y sobre quien se desea realizarla.

También hay campos ocultos que mantienen información de los combos de filtrado, que son actualizados ante cada operación de este tipo, para que al retorno de la página se refresque el frame correspondiente. Cuando se solicita filtrar los datos, disparamos una función JavaScript que controla la validez de la operación y genera el nuevo campo where. Para conocer en detalle todos los campos auxiliares remitirse a las especificaciones técnicas.

Algunos de los criterios de implementación que seguimos en el diseño de la interfase fueron:

- *Combo Box* – Estos elementos permiten mostrar todas las opciones disponibles y validar implícitamente la información que ingresa el usuario, eliminando errores en este sentido, ya que la información allí desplegada se toma de la metadata o de los datos según corresponda. Son de gran utilidad en el filtrado, en el panel de control y también en las consultas *SQL*.
- *Botones* - Fueron utilizados en el panel de control, llaman a funciones que como vimos copian la información necesaria a campos ocultos y luego envían el formulario al servidor.
- *Frames* – Se utilizaron por diferentes razones: en el ambiente multidimensional por eficiencia, dado que los combo box de filtrado tienen una carga importante por la cantidad de opciones, y de esta forma se calculan una sola vez al principio. En el ambiente relacional, además del tema de la eficiencia, por practicidad para el usuario, ya que puede ver conjuntamente la consulta realizada y los resultados obtenidos.
- *Abrir nuevas ventanas* - El motivo principal es permitir al usuario tener “a mano” ambas páginas. Se llevó a cabo en las gráficas, la ayuda en línea, salvar la página, y para ver la estructura de tablas del datawarehouse en el caso de la consulta libre.
- *Links* – Fundamentalmente utilizados en los cabezales de la tabla resultado para hacer *drill*, resultando la forma más clara y conveniente. También se usaron aplicándolos a imágenes en la barra de herramientas y en la tabla resultado (gráfica estilo torta), los que invocan funciones *JavaScript* que copian y validan información a campos ocultos o abren nuevas ventanas (guardar, graficar y ayuda).

Módulos

El código del sistema está organizado en páginas ASP de acuerdo a lo explicado en el punto de Diseño y las funciones utilizadas por ellas las distribuimos según correspondan al servidor o al cliente, y en cada uno de ellos las agrupamos de acuerdo a la o las páginas que las necesitan [Especificaciones Técnicas - Bibliotecas VisualScript/Javascript].

Criterios particulares de implementación.

◆ Frame de Filtrado

La construcción del frame de filtrado se realiza consultando los metadatos y datos del cubo, lo que hacemos es, recorrer por nivel las distintas jerarquías de la información generando el combo correspondiente, para cada una de las dimensiones. Dado el volumen de los datos de cada dimensión acotamos la cantidad de valores a desplegar asegurándonos siempre que un nivel de información se incluya completamente.

◆ Tablero de Control

Para la implementación del tablero de control accedemos a la metadata para obtener los nombres de las dimensiones y medidas, además de investigar las partes de la

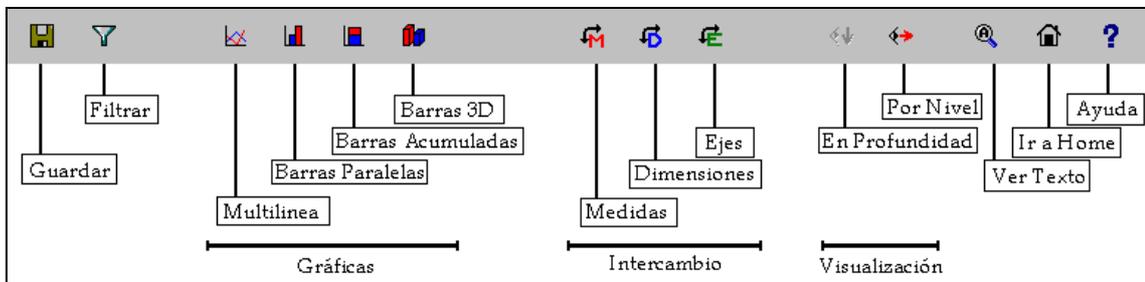
consulta actual, determinando si pertenecen o no a ella, para desplegarlas en el combo correspondiente.

Para evitar que el usuario efectúe alguna operación inválida, dejando la consulta en un estado inconsistente, le deshabilitamos los botones correspondientes a dichas opciones.

	MEDIDAS		DIMENSIONES			
AGREGAR A:	[Nueva Medida] ▾	Columnas	Nueva Dimension ▾	Columnas	Nueva Dimension ▾	Filas
QUITAR DE:	[Medida] ▾	Columnas	Dimension ▾	Columnas	Dimension ▾	Filas

◆ Barra de Herramientas

Implementamos una barra de herramientas para concentrar la mayor parte de funciones que ofrece nuestro visualizador, utilizando para ello imágenes insertas en links.



◆ Tabla Resultado

Aquí procesamos los datos de los cabezales de la tabla, para que el usuario solo al verla supiera si se trata de una medida o una dimensión, y en este caso si puede hacer "drill" o no.

Agregamos la posibilidad de ver la información, de cada una de los renglones de la tabla, en una gráfica Estilo Torta.

	Dimensiones	Tipo Actividad > Curso		
		Medidas	[Reprobados]	[Aprobados]
Materias > COMPUTACION +			0,00	2
Materias > ELECTIVAS +			0,00	0,00
Materias > FISICAS +			0,00	0,00
Materias > MATEMATICAS +			0,00	1

◆ Pasaje de parámetros

Hay casos particulares en que las páginas son invocadas mediante pasaje de parámetros, cuando abrimos una nueva ventana (*guardar.asp* o *graficar.asp*) o una página invoca a cada uno de sus frames y debe enviarles información (*mdx.asp*).

◆ Guardado de información

Guardar.asp genera una nueva página sin frames, que el browser permite almacenar tal cual se ve, sin problemas. A esta le enviamos los parámetros necesarios para la

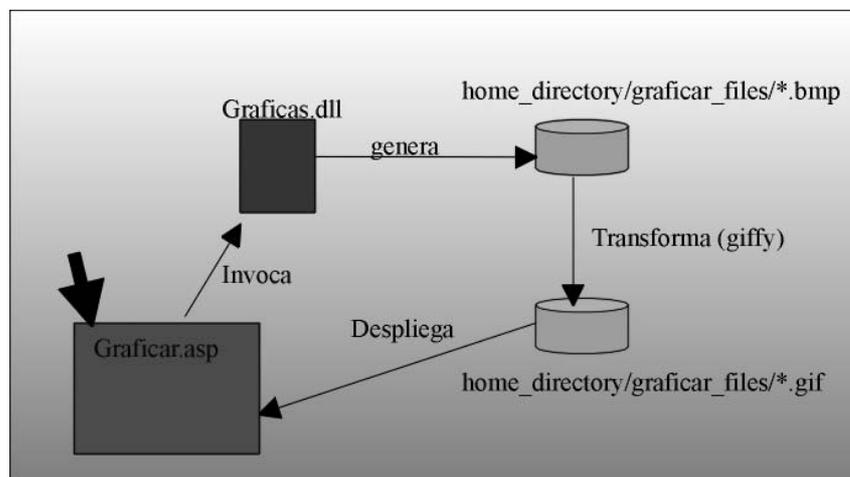
reconstrucción de la página original, excluyendo todas sus funcionalidades, dado que ella sólo muestra la tabla resultado sin permitir ningún tipo de modificaciones. Sólo está disponible el link para retomar la conexión.

◆ Consultas SQL

En el caso de las consultas *SQL* los criterios utilizados fueron diferentes, dado que se trata de consultas específicas. Los campos ocultos solo manejan valores para construir las condiciones a incluir en el *where* de la consulta, ya que el resto del texto de la misma es fijo. Aquí también se utilizan funciones *JavaScript* donde se copia y valida la información seleccionada o ingresada por el usuario.

Gráficas:

El proceso de construcción de una gráfica partiendo de la invocación de la página Graficar.asp desde el visualizador puede resumirse con el siguiente esquema:



La parte medular del graficado se encuentra en Graficas.dll, ésta biblioteca dinámica fue programada con Visual Basic 6.0 usando el objeto MSChart que tiene propiedades útiles para nuestros objetivos como ser: tipo de gráfica, ejes, rótulos de los ejes, referencias, etc.

El inconveniente de este objeto es que no posee un método o propiedad Imagen que almacene el objeto en formato gráfico, es decir archivos con formato bmp, gif, etc.

Esto nos llevó a tener que implementar una forma de obtener la imagen con ayuda del portapapeles (Clipboard) es decir copiamos el objeto conteniendo el gráfico en el portapapeles y lo pegamos como Imagen del objeto Form. A partir de esto obtenemos un archivo .bmp.

Nuestro objetivo es generar un archivo .gif para visualizar en el Web dado que ocupan menos espacio y por lo tanto requieren menor tiempo de transferencia; es así que usamos un software libre (giffy) [7] que transforma el .bmp en .gif. Una vez completada la conversión, borramos el .bmp. La dll le retorna a la página el nombre del archivo generado, el cual es utilizado por ésta para mostrar dicho archivo.

Sabiendo que muchas personas podrían estar generando el mismo tipo de gráfica con diferentes parámetros es indispensable obtener una identificación única por cada

solicitud de graficado. Para ello la dll genera el nombre del archivo concatenando el estilo de gráfico con un número randómico (función Random).

Otro detalle a tener en cuenta es el espacio del disco del servidor usado con archivos temporales como lo son estas imágenes, para ello programamos un cron NT que se dedica a limpiar los archivos .gif del directorio correspondiente generados el día anterior.

Por más detalles ver especificaciones técnicas.

Olap

Los aspectos relevantes de la implementación OLAP pueden resumirse en los siguientes puntos:

- Del análisis conceptual resultaron 4 cubos los cuales fueron implementados como los 6 cubos que se detallaron anteriormente, esto se debió a que del modelo conceptual de las Actividades se desprendieron 3 cubos: uno para análisis de las actividades curso, curso inválido y curso caducado en el cual la dimensión Período no tiene sentido (Activ_cursos); otro para el análisis de las actividades examen estudiadas según el período en su visión numérica, es decir se habla de primer, segundo, etc. período de una asignatura en un año (Activ_período_numerico) y por último un cubo para el análisis de las actividades examen según el período en una visión asociada al mes, es decir se habla de período de Julio, Agosto, etc. de una asignatura en un año (Activ_período_nombre).
- Para hacer más eficiente la implementación de la dimensión Materias creamos una vista Materias1 [Ver especificaciones técnicas – Implementación OLAP] que nos permite fácilmente obtener el conjunto de las materias con sus asignaturas y de esta forma nos evitamos usar un esquema snowflake -que es el que se desprende de nuestras tablas- menos eficiente. [Ver especificaciones técnicas - Optimización de la carga]
- El OLAP Services permite la creación de dimensiones compartidas (shared dimensions) o privadas a cada cubo. A partir del análisis del modelo de cubos implementamos como dimensiones compartidas:



La dimensión Carreras la hemos implementado como compartida aunque para nuestros cubos esto no es necesario lo hicimos con la idea de que quizás con el tiempo podrían surgir nuevas preguntas que se tradujeran en nuevos cubos y que posiblemente necesitaran la dimensión Carreras.

Hay que tener en cuenta que las dimensiones compartidas tienen una propiedad Member Property que sirve para agregar atributos específicos a un nivel de una dimensión, esta propiedad no la tienen las dimensiones privadas y muchas veces puede ser el elemento para decidir si una dimensión será compartida o no. Otra facilidad que ofrece una dimensión de este tipo es poder determinar hasta qué nivel se permite el drill-down.

El definir dimensiones compartidas tiene como ventaja tener la dimensión disponible para cualquier cubo que la necesite, pero tiene un pequeño inconveniente que se manifiesta en el momento de procesar un cubo. Si por alguna razón cambia la estructura de una dimensión compartida habrá que procesar todos los cubos que la utilizan, lo que obliga a tener muy claro el alcance de una dimensión. Nosotros hemos realizado un esquema de dependencia de dimensiones que será útil a la hora del proceso de los cubos. [Ver especificaciones técnicas – Implementación OLAP]

Definimos dimensiones privadas para aquellos cubos cuyas dimensiones tenían sentido en ese cubo y nada más. [Ver especificaciones técnicas – Implementación de los cubos en OLAP Services]

- El nivel All Levels se encuentra habilitado en todas las dimensiones.
- Siempre quisimos evitar el uso del esquema snowflake debido a que baja la performance de carga de los cubos, pero la dimensión Carreras debió definirse con este esquema ya que queríamos incluir el nivel Materias (con su nombre) en la dimensión y este dato se encuentra en la tabla BD_Materias.
- Encontramos que al hacer drill up nuestras medidas de cálculo de promedios de notas no se podían calcular fácilmente como las medidas que cuentan o suman. Esto se debe a que el OLAP Services no incluye en su gama de operaciones a efectuar en un drill up la operación AVG (average). Lo que usamos fueron los llamados Miembros Calculados que son medidas que no se almacenan en el cubo sino que se calculan cada vez que son necesarios, lo cual implica un retardo en el tiempo de consulta.

Es así que definimos en los cubos: Activ_cursos, Activ_periodo_nombre y Activ_periodo_numerico miembros calculados que se encargan de hacer el promedio “a mano” un miembro calcula la suma de las notas y otro la cantidad de actividades, luego la medida es el cociente de ambos.

- El modo de almacenamiento de cada cubo es el siguiente:

Cubo	Modo de Almacenamiento	Por qué?
Activ_cursos	MOLAP	Se necesita alto nivel de respuesta, se llega al mayor nivel de detalle a la hora de consultar.
Activ_periodo_nombre	MOLAP	Idem anterior
Activ_periodo_numerico	MOLAP	Idem anterior
Avance	HOLAP	Se necesita alto nivel de respuesta, y no necesita llegar al mayor nivel de detalle.
Desempenio	MOLAP	Idem Activ_cursos
Inscripciones	HOLAP	Idem Avance

- Siguiendo las 10 recomendaciones para optimización de la carga de los cubos que se mencionan en las Especificaciones Técnicas - Optimización de la carga, seteamos algunas propiedades como ser:
 - Seteamos la propiedad Member Key a los niveles que correspondiese en todas las dimensiones.
 - Seteamos la propiedad Unique Member de los niveles que correspondiese en todas las dimensiones.
 - Corrimos la función Optimización del esquema para todos los cubos (en tiempo de diseño).
 - Aumentamos el tamaño del buffer de procesos a 20 MB.
 - Se trato de utilizar el tipo de datos más eficiente en la definición de las medidas.
- Para favorecer el tiempo de respuesta de las consultas a los cubos desde el web usamos la facilidad del OLAP Services: Optimización basada en uso, que permite tener un cache con las consultas de uso más frecuentes y las que demoran más. Esta optimización deberá ser monitoreada cuando el sistema esté en funcionamiento para hacerle las mejoras que corresponda.
- El modo de mantenimiento de los datos de cada cubo que seleccionamos fue el modo REFRESH luego de que la primera vez se haya realizado un FULL PROCESS.

5 – CONCLUSIONES.

CONCLUSIONES.

- Hemos implementado nuestra solución combinando diferentes herramientas Microsoft, pero a la hora de utilizar el servicio, el usuario no se ve limitado a tener instalado en su PC productos Microsoft.
- En cuanto a la programación Microsoft, el OLAP Services nos resultó más rígido en la implementación de los cubos comparado con lo que vimos en otras herramientas del mercado; resolvíamos lo mismo pero con mayor esfuerzo. Su browser no está muy desarrollado.
- Tiene la ventaja de ser práctico a la hora de mantener la actualización de los datos en los cubos una vez puesto en marcha el sistema, así como de ofrecer herramientas para mejorar la performance. (Optimización Basada en Uso).
- Microsoft está continuamente en cambios, las herramientas con las que empezamos el taller han sufrido cambios y hemos tenido que ir agregando parches en el correr del mismo. Esta es una de las dificultades de trabajar con Microsoft.
- El modelo multidimensional de Fernando Carpani ha resultado una forma clara y estándar para entender los talleres anteriores y el nuestro continúa usándolo.
- Dado que nuestro principal interés fue resolver las consultas cuidando que el resultado fuese el correcto y la respuesta fuese dada en un tiempo razonable -el mejor que pudiésemos lograr teniendo en cuenta limitaciones de hardware y que trabajamos con productos Microsoft-, decidimos en la mayoría de los cubos un almacenamiento de tipo MOLAP sabiendo que el recurso disco no es un problema y asumiendo una pérdida en transparencia de los datos que quedan en formato multidimensional; descartamos el ROLAP y en dos oportunidades usamos HOLAP. Destacamos como ventajoso el tener la tercera opción HOLAP ya que si el volumen de datos fuese muy grande (cosa que para nuestra realidad no ocurre) y el recurso disco fuese un recurso escaso podríamos cambiar a este modo sin pérdida considerable de performance.
En resumen, el crecimiento del volumen de datos y el cambio en el tipo de consultas que se piden sobre ellos puede ser acompañado por un cambio gradual en el modo de almacenamiento, que seguramente se dará en este orden: de MOLAP se pasará a HOLAP y como último recurso se usará ROLAP, todo esto dado que nuestro principal objetivo es: aceptable tiempo de respuesta.
- Logramos implementar un visualizador que no depende del datawarehouse al que se está conectando. Sirve tanto para Bedelía como para cualquier otro.
- Microsoft Interdev proporciona un ambiente cómodo y sumamente integrado con todas las herramientas Microsoft, desde aquí se puede crear o modificar elementos de cualquier tipo que se deseen incluir en el proyecto. Diferencia claramente el ámbito de desarrollo del de producción, lo que ayudó en la programación.
- La mayor dificultad para el desarrollo del visualizador estuvo en encontrar el criterio a seguir para la construcción de las consultas MDX de una forma relativamente sencilla, pero logrando brindar todas las funcionalidades requeridas.
- Se requirió más esfuerzo del imaginado inicialmente, para conseguir que el código del cliente sea independiente del navegador utilizado. Cabe destacar que en la programación nunca se diferencia código según el navegador, todos ejecutan las mismas sentencias.
- El tema enseñanza ha resultado un tema complicado a la hora de la carga de los datos del datawarehouse y es lo que muchas veces retrasa un taller de este tipo.

- A la hora de decidir como implementar las páginas, la variada información existente sumado a nuestra poca experiencia en el tema y lo innovador del mismo nos dificultó la toma de decisiones.
- Logramos lo que queríamos y más.

MEJORAS.

Proponemos algunas mejoras:

- Sería bueno que la carga del datawarehouse no partiese de unos datos recolectados vía consultas sql y posterior traslado vía diskette desde bedelía, sino que se podría estudiar la posibilidad de conexión segura entre el sistema de Bedelías y este sistema.
- Sería bueno que se implementara otra forma para poblar las tablas fact del sistema, se nos ocurre que se podría usar el DTS para seguir en la misma línea de Microsoft.
- Mejoraríamos la interfase de la página principal del navegador.
Proponemos re-pensar lo que nosotros llamamos el tablero de control que tal vez no quedó del todo intuitivo, buscar la manera de quitar el nombre de la dimensión de los cabezales de la tabla resultado, colocándolo en otro lugar, que las gráficas incluyan como título algo sobre los datos que se están graficando.
- Podría estudiarse la forma de brindarle al usuario la posibilidad de visualizar una medida "inventada" por él, calculada en base a las existentes en la tabla resultado (por ejemplo: suma de dos de ellas).
- Se puede cambiar la forma de iniciar la sesión de análisis multidimensional, en lugar de setear en una página la consulta inicial para cada cubo, teniendo sólo el nombre de éste, acceder a la metadata y mediante algún criterio construir la consulta.
- Si solo se decide usar el navegador para Bedelía tal vez habría que agregarle algunos detalles que ayudarían a la comprensión del mismo pero haría perder su generalidad como ser: que la ayuda esté más relacionada con la realidad de Bedelía, no usar la terminología dimensión o medida eligiendo términos significativos acordes al tema enseñanza, etc.
- Habría que profundizar en el tema seguridad del datawarehouse y del servidor NT, cosa que no se hizo por estar fuera del alcance del taller.
- Estudiar la performance del sistema e identificar los puntos críticos de la misma, temas que también están fuera del alcance del proyecto.

AGRADECIMIENTOS

Queremos agradecer la colaboración brindada en el transcurso del proyecto al profesor Raúl Ruggia. Sus aportes e inquietudes fueron de gran utilidad para nosotros. También agradecemos a los docentes Sandro Moscatelli, Pablo Garbussi y Gerardo Ares por realizar la carga del datawarehouse. También al docente Daniel Perovich por su apoyo en los temas sobre Visual Basic.

Agradecemos también a nuestro compañero de trabajo Julio Cardozo egresado de la Facultad de Bellas Artes por ayudarnos con el diseño de la tapa del informe.

Por último, queremos agradecer a nuestros familiares, su apoyo incondicional y su fuerza nos ayudaron en los momentos más difíciles. Muchas Gracias.

BIBLIOGRAFÍA

- [1] – Proyecto de Taller V- “Estudio de Técnicas y Software para la construcción de sistemas de DW” año 1997. Veronika Peralta, Alvaro Ilizalde.
- [2] – Proyecto de Taller V - “Sistema de Datawarehousing, Carga y Control de Calidad” año 1998. Lucía Coppola, Diego Olave, Raquel Avella.
- [3] – Proyecto de Taller V – “Desarrollo de un Sistema de Data Warehousing para el INCO” año 1998. Andrea Pereyra, Osvaldo Varallo.
- [4] – Manipulate and Query Olap Data using ADOMD and MDX Part II: Writting the App. – Carl Nolan.
- [5] – Trabajo de carga para Taller V año 1999. Pablo Garbusi, Gerardo Ares, Sandro Moscatelli.
- [6] – P100 OLAP Services solution Bootcamp. www.olaptrain.com
- [7] –Trabajo en el marco de Tesis de Maestría: “Ejemplos de Modelado Multidimensional con el modelo INCO” año 1999. Fernando Carpani.
- [8] – Software shareware (giffy) extraído de: <http://flybum.hypermart.net>
- [9] – “Designing and Implementing a Data Warehouse using Microsoft SQL Server 7.0”. Microsoft Workbook.
- [10] – “Data Warehousing: Building the Corporate Knowledge Base” Ralph Kimball.
- [11] – www.microsoft.com/sql/70 Sitio Web de Microsoft para Sql Server 7.0.
- [12] – Microsoft Development Network Library Visual Studio 6.0 (MSDN)
- [13] – Microsoft Development Network Library Sql Server 7.0. (MSDN)
- [14] – P200 Olap Services Programers Bootcamp – OLAP Train.
- [15] – Microsoft Olap Solutions – Erik Thomsen, George Spolford, Dick Chase.
- [16] – An Introduction to Multidimensional Database Technology – Kernan Technologies.
- [17] – Designing and Implementing a Data Warehouse using Microsoft SQL Server 7.0 – Microsoft Training.
- [18] – Manipulate an Query Olap Data using ADOMD and Multidimensional Expressions – Carl Nolan (Internet).
- [19] – Java Script Guide – Netscape Navigator Version 4.0.

[20] – Java Script Soluciones Instantaneas – Rick Darnell.

[21] – Sitio de programación de páginas "asp" <http://asp.superexpert.com>

[22] – Sitio de programación de páginas "asp" <http://www.programmersresource.com>

[23] – Sitio de utilidades para creación de páginas web <http://abiglime.com/webmaster/>