

**ESCUELA TÉCNICA SUPERIOR DE INGENIEROS
INDUSTRIALES Y DE TELECOMUNICACIÓN**

UNIVERSIDAD DE CANTABRIA



Trabajo Fin de Grado

**Generación de escenarios de planificación
de redes mediante técnicas OLAP
(Generation of planning scenarios of
networks with OLAP technics)**

Para acceder al Titulo de

***Graduado en
Ingeniería de Tecnologías de Telecomunicación***

Autor: Jesús Ramos García

10 - 2015

TÍTULO	Generación de escenarios de planificación de redes mediante técnicas OLAP		
AUTOR	Jesús Ramos García		
DIRECTOR	Alberto Eloy García Gutiérrez		
TITULACIÓN	GRADUADO EN INGENIERIA DE TECNOLOGIAS DE TELECOMUNICACIÓN	FECHA	10-15

Indice

INDICE DE FIGURAS	5
RESUMEN	7
ABSTRACT	8
1. Introducción	9
1.1 Introducción	9
1.2 Descripción general y objetivos del proyecto	10
1.3 Organización del documento	10
2. Conceptos Teóricos	13
2.1 Minería de datos/Big Data	13
2.2 Cubos/Almacenes de datos.....	19
3. Aspectos Prácticos.....	21
3.1 Herramientas para Big Data, Data Mining y OLAP	21
3.2 Escenario de aplicación y requerimientos	22
4. Desarrollo del proyecto.....	23
4.1 Especificación del problema.....	23
4.2 Organización de los datos	31
4.3 PowerPivot	35
4.4 Creación del cubo.....	39
4.5 Análisis del cubo.....	47
4.6 Manejo del cubo.....	49
4.7 Análisis de resultados.....	63
5. Otros Ejemplos	69
Conclusiones y líneas futuras	81
REFERENCIAS.....	83
ANEXO	86
TABLAS	87
Tabla Total LO (12AM):.....	89
Tabla filtrada LO (12AM y 1AM):.....	90
Tabla total PH (12AM):.....	91
Tabla DAX LO (ESW, ESW1a, LC10):	93
Tabla DAX PH (1, 70, 90):.....	94
Tabla de resultados LO:.....	95
Tabla de resultados PH:.....	96
Tabla technology:	98
PROGRAMAS VBA.....	99

RecData_LO:.....	100
RecData_PH:.....	106
calculos_LO:.....	107
calculos_DAX_LO:.....	111
calculos_PH:	113
calculos_LO_hora:	114
calculos_PH_hora:.....	115
calculos_PH_hora2:.....	121
calculos_PH_technology:	122
calculos_hibridos:.....	125
resultados_LO:	127
resultados_PH:	131
barra:	133
Códigos DAX	134
Suma de LC:.....	135
Suma de PIU:.....	135
Suma de LC 7PM:.....	135
Suma de PIU 7PM:.....	140
Consumo Watt/hr_LO:.....	141
Suma de L LC:	142
Suma de R LC:.....	142
Suma de PIUs:.....	142
Suma de repeaters:	142
Consumo L LC:	142
Consumo PIU:.....	142
Suma de L LC 7PM:.....	143
Suma de R LC 7PM:.....	170
Consumo Watt/hr_PH:.....	170

INDICE DE FIGURAS

Figura 1: Exceso de información [23]	10
Figura 2: Pasos del KDD (Knowledge Discovery in Databases) [24]	14
Figura 3: Las cuatro grandes Vs del Big Data [25]	15
Figura 4: Planes de proveedores de servicios para invertir en escenarios basados en Big Data [15]	17
Figura 5: Big Data en las telecomunicaciones [26]	18
Figura 6: Relación entre tipos de datos y gráfica de velocidad [12]	18
Figura 7: Grado de importancia de servicios [15] Eje x: Grado de importancia Eje y: Servicios	19
Figura 8: Relación entre Data Warehouse y OLAP [27]	20
Figura 9: Cubo OLAP [28] Eje x: Web Page, Eje y: Action, Eje z: Time	20
Figura 10: Representación geográfica de la red (nodos de nivel 0) [50]	24
Figura 11: Consumo Energético (en KWatts) [50]	25
Figura 12: Hoja con datos a las 12AM (LO)	26
Figura 13: Hoja con datos a las 12AM (PH) 1	28
Figura 14: Hoja con datos a las 12AM (PH) 2	29
Figura 15: Código RecData_LO 1	31
Figura 16: Código RecData_LO 2	32
Figura 17: Código RecData_LO 3	32
Figura 18: Código RecData_PH 1	33
Figura 19: Código RecData_PH 2	33
Figura 20: Diagrama de flujo RecData	34
Figura 21: Esquema del programa redata	34
Figura 22: Operaciones secundarias LO Excel	36
Figura 23: Operaciones secundarias LO DAX	37
Figura 24: Operaciones secundarias PH Excel	38
Figura 25: Operaciones secundarias PH DAX	38
Figura 26: Tabla dinámica vacía	40
Figura 27: Ejemplo de selección de campos de un cubo	40
Figura 28: Tabla dinámica LO	42
Figura 29: Cubo OLAP LO Eje x: Tipo de disp. Eje y: Valores Eje z: Hora	43
Figura 30: Tabla dinámica PH 1	44
Figura 31: Tabla dinámica PH 2	45
Figura 32: Cubo OLAP PH Eje x: Conf_Index Eje y: Valores Eje z: Hora	46
Figura 33: Ejemplo de Segmentación de datos	46
Figura 34: MDX	47
Figura 35: Search	48
Figura 36: Filter	49
Figura 37: Defaults	49
Figura 38: Expresión individual de VALORCUBO	50
Figura 39: Ejemplo de VALORCUBO (LO)	51
Figura 40: Código Calculos_LO 1	51
Figura 41: Código calculos_LO 2	52
Figura 42: Identificador OLAP 1	52
Figura 43: Ejemplo de código DAX "7PM" Parte LO	53

Figura 44: Ejemplo de operación DAX parte LO	53
Figura 45: Código Calculos_DAX_LO	54
Figura 46: Ejemplo dispositivos ESW (Consumo total)	55
Figura 47: Ejemplo dispositivos LER (Consumo total)	56
Figura 48: Ejemplo dispositivos LSR (Consumo total)	57
Figura 49: Ejemplo de VALORCUBO (PH)	58
Figura 50: Código calculos_PH 1	58
Figura 51: Código calculos_PH 2	59
Figura 52: Identificador OLAP 2	59
Figura 53: Fragmento del código DAX "7PM" Parte PH	60
Figura 54: Consumo PH DAX	60
Figura 55: Código calculos_DAX_PH	60
Figura 56: Gráfica del dispositivo 111	61
Figura 57: Gráfica del dispositivo 139	61
Figura 58: Diagrama de flujo calculos	62
Figura 59: Esquema programa calculos	62
Figura 60: Código resultados_PH	64
Figura 61: Diagrama de flujo resultados_calculos	65
Figura 62: Esquema del programa resultados	65
Figura 63: Esquema del desarrollo del proyecto	67
Figura 64: Cubo con filas technology Eje x: Technology Eje y: Valores Eje z: Horas	69
Figura 65: Código calculos_PH_technology	70
Figura 66: Identificador OLAP 3	70
Figura 67: Gráfica OTN	70
Figura 68: Gráfica OXC	71
Figura 69: Gráfica ROADM-OTN	71
Figura 70: Cubo Type-technolgy Eje x: Type Eje y: Valores Eje z: Technology	71
Figura 71: Código calculos_hibridos (Type-Technology)	72
Figura 72: Cubo LO_hora Eje x: Horas Eje y: Valores Eje z: Tipo de disp	73
Figura 73: Ejemplo de tabla dinámica LO "hora"	73
Figura 74: Código calculos_hora_LO 1	73
Figura 75: Código calculos_hora_LO 2	74
Figura 76: Identificador OLAP 4	74
Figura 77: Consumos de algunas LC	75
Figura 78: Cubo PH horas Eje x: Horas Eje y: Valores Eje z: Conf_Index	75
Figura 79: Ejemplo de tabla dinámica PH "hora"	76
Figura 80: Código calculos_hora_PH 1	76
Figura 81: Código calculos_hora_PH 2	76
Figura 82: Identificador OLAP 5	77
Figura 83: Consumos de algunos dispositivos	77
Figura 84: Diagrama de flujo calculos_hora	78
Figura 85: Esquema del programa calculos_hora	78
Figura 86: Fuente de datos cubo	79

RESUMEN

Debido a la gran cantidad de datos que a veces se tienen que manejar, es necesario el uso de herramientas que se encarguen de almacenar, reducir, extraer, analizar, procesar o modificar dichos datos de la forma más rápida y sencilla posible. Las bases de datos multidimensionales, con el nombre de cubo OLAP (On-Line Analytical Processing) se encargan de representar la información elegida de forma multidimensional, es decir, mediante cubos de varias dimensiones, haciendo que dicha información sea más fácil de visualizar y se puedan llegar a resultados que, sin el uso de las técnicas tipo “OLAP”, sean más difíciles de obtener.

El gran conjunto de información del que disponemos es acerca de dispositivos de telecomunicación. Antes de emplear las técnicas OLAP se realizarán una serie de pasos intermedios, como organizar la información en una única tabla y emplear una herramienta para realizar algunas operaciones, para posteriormente crearnos el cubo OLAP y poder analizar su información y extraer resultados, después se crearán distintos cubos con la información distribuida de forma diferente para ver la utilidad de tener almacenada la información de dicha manera (en cubo) y para obtener distintos resultados.

ABSTRACT

Because of the number of information we sometimes need to work, it is needed the uses of tools which stored, reduced, analyzed, processed and modified that information to the fastest and easiest way. Multidimensional databases, with the name OLAP cube (OnLine Analytical Processing), have the job to show the information to the multidimensional form with multidimensional cubes which make the information more easier to watch it and we can obtain results that are more difficult to obtain without those technics.

The great number of information that we have is about telecommunications devices. We will do some task before the use of OLAP technics, some of that task are the agrupation of the information in one table or the use of a tool to do operations, next we will create the OLAP cube to we can analyze its information and we can extract results, in the end we will create another different cubes to see the useful of that way to store the information and to obtain another results.

1. Introducción

En este apartado se dará una descripción general del proyecto. Al principio se comentará el concepto de información, el proyecto está muy relacionado con ese término. También se describirán la motivación y objetivos del mismo así como la organización de este documento.

1.1 Introducción

En el mundo actual estamos rodeados de información de todo tipo, el término información es un concepto universal y de una interpretación casi ilimitada que es aplicable a cada aspecto de nuestro día a día en la actual sociedad de la información TIC (Tecnologías de la Información y las Comunicación) y, en función de ella, tomamos decisiones que afectan directamente a nuestro bienestar.

Ejemplos de información sencillos y cotidianos son la temperatura de la calle, la cantidad de dinero de la que disponemos, los alimentos que tenemos en nuestra nevera... Estos datos pueden ser cuantificables o no: igual tenemos pocos alimentos en nuestra nevera pero los tenemos de un “tipo” el cuál únicamente necesitamos. En nuestro ejemplo “el número” y el “tipo” podríamos decir que representan ambos información útil y manejable.

Por otro lado hay muchas formas de representar la información aparte de la numérica: el color de un objeto, la dificultad de una actividad, el estado anímico de una persona...

Cuanto más pasa el tiempo y más evolucionamos, más información podemos extraer del entorno que nos rodea y más acciones podemos realizar en base a esa información, sobre todo gracias a la aparición y desarrollo de internet y su implementación en los teléfonos móviles, que permiten que la información llegue a cada persona, ¿Pero qué pasa cuando dicha información es tan grande que un ser humano no puede manejarla adecuadamente, como podemos ver en la **figura 1**?



Figura 1: Exceso de información [23]

Lo mismo que una persona no necesita manejar dicha cantidad de información en su vida cotidiana, este problema se traslada a muchas empresas que, debido a su tamaño, necesitan de mecanismos que gestionen adecuadamente la gran cantidad de información que poseen para poder aplicarla de forma conveniente a sus correspondientes ámbitos y poder realizar una gestión de recursos óptima y satisfactoria.

Dichos mecanismos vienen representados por sistemas computacionales que disponen de herramientas para procesar gran cantidad de información y también tienen en cuenta su complejidad ya que, como hemos mencionado anteriormente, la información no solo viene representada mediante “números” si no de una variada multitud de formas.

Al conjunto de técnicas que nos permite trabajar con un conjunto de información se las conoce como técnicas de “Big Data”. Entre ellas, la encargada de extraer datos y obtener resultados a partir de ellos se la conoce como minería de datos.

1.2 Descripción general y objetivos del proyecto

En el proyecto disponemos de un gran conjunto de información relacionada con dispositivos de telecomunicación, organizada y dividida en numerosas tablas y hojas.

El objetivo del proyecto es organizar la información en cubos OLAP, analizarla y sacar la información del consumo energético de cada dispositivo a lo largo de un día en función de dicha información mediante las operaciones y técnicas necesarias, de esta forma podemos estudiar la evolución energética de un dispositivo concreto a lo largo de un día. Para lograr dicho objetivo habrá que manejar la información existente de forma adecuada.

En resumen, tenemos dos objetivos: Utilizar los cubos OLAP para estudiar distintas maneras de trabajar con la información y analizar los resultados obtenidos.

Podríamos decir que el proyecto consiste en un “Big Data aplicado a las telecomunicaciones” o, siendo más concreto, “Big Data aplicado a redes de telecomunicación energéticamente eficientes”.

1.3 Organización del documento

En la primera parte del documento haremos un desarrollo teórico de los conceptos “Minería de Datos” y “Big Data”, bases fundamentales en las que se apoya el proyecto; Se definirán los

conceptos, se especificarán sus utilidades, elementos derivados y se expondrán ejemplos; Habrá una parte que se dedicará al Big Data enfocado al sector de las telecomunicaciones. Luego se comentarán los conceptos de “cubos” y “almacenes de datos”, elementos usados por el propio proyecto. En la segunda parte se dará una información general sobre el proyecto, que nos permitirá tener una visión de conjunto sobre él y determinar sus objetivos. También se comentarán aspectos relacionados sobre el escenario mediante el cual se desarrollará el trabajo y los requerimientos que necesita dicho escenario. Por último en la tercera parte se realizará el desarrollo del proyecto en detalle, pasando por cada una de las fases más importantes, cada una determinada por un objetivo concreto, hasta llegar a alcanzar el objetivo final.

2. Conceptos Teóricos

Los apartados siguientes extenderán los conceptos de Big Data y minería de datos, se darán ejemplos de aplicación y se profundizará en como se aplican dichos conceptos a las telecomunicaciones. Después se hablará de los almacenes de datos y los cubos (un tipo de almacén de datos), elementos importantes cuándo hablamos de trabajar con la información.

2.1 Minería de datos/Big Data

Desarrollo teórico:

La minería de datos, mas conocida por el término inglés Data Mining, es un conjunto de técnicas de análisis de información en bases de datos de forma que a partir de patrones de información adecuados se pueda llegar a conclusiones y resultados satisfactorios. Podríamos decir que forma parte de un proceso completo de análisis de información, formado por almacenamiento, preprocesado, transformación, minería de datos e interpretación, también conocido como KDD (Knowledge Discovery in DataBases) siendo la minería de datos uno de los pasos más

importantes ya que es el que da más significado y relevancia al significado y posterior aplicación de los datos [1].

En la **figura 2** podemos ver los pasos del KDD para entender mejor lo explicado anteriormente.

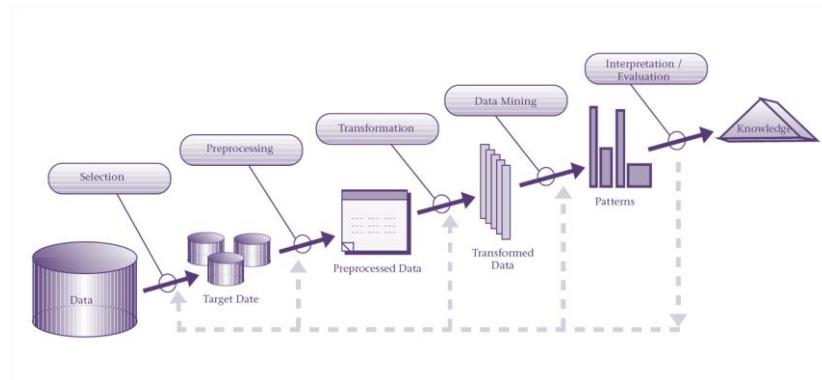


Figura 2: Pasos del KDD (Knowledge Discovery in Databases) [24]

Si extendemos los conceptos de KDD y Data Mining nos topamos con el término “Big Data”, muy extendido hoy en día, que engloba todas las disciplinas anteriores añadiendo la particularidad de que se manejan grandes, complejos y variados volúmenes de información [2]. A grandes rasgos, podríamos decir que “Big Data” es un conjunto de técnicas que se encargan de trabajar con la información.

Sin embargo algunas fuentes opinan que “Data Mining” y “Big Data” son conceptos totalmente “separados”, al considerar a la primera como el proceso de extracción de información útil y a la segunda el proceso de organizar la información para ser usada de forma rápida [3].

En la sociedad TIC en la que vivimos la cantidad de datos que se generan van desde los 1700 billones de bytes por minuto [4] hasta los 2,5 quintillones de bytes al día [5]. Obviamente, no toda la información generada es útil para su estudio, ni mucho menos. Las técnicas de “Big Data” se encargan de tratarla adecuadamente atendiendo a las tres Vs: Volumen, variedad y velocidad [6], aunque se puede incluir una cuarta V, la veracidad de los datos.

En la **figura 3** se muestra las definiciones de las 4 Vs y su relación con el Big Data.

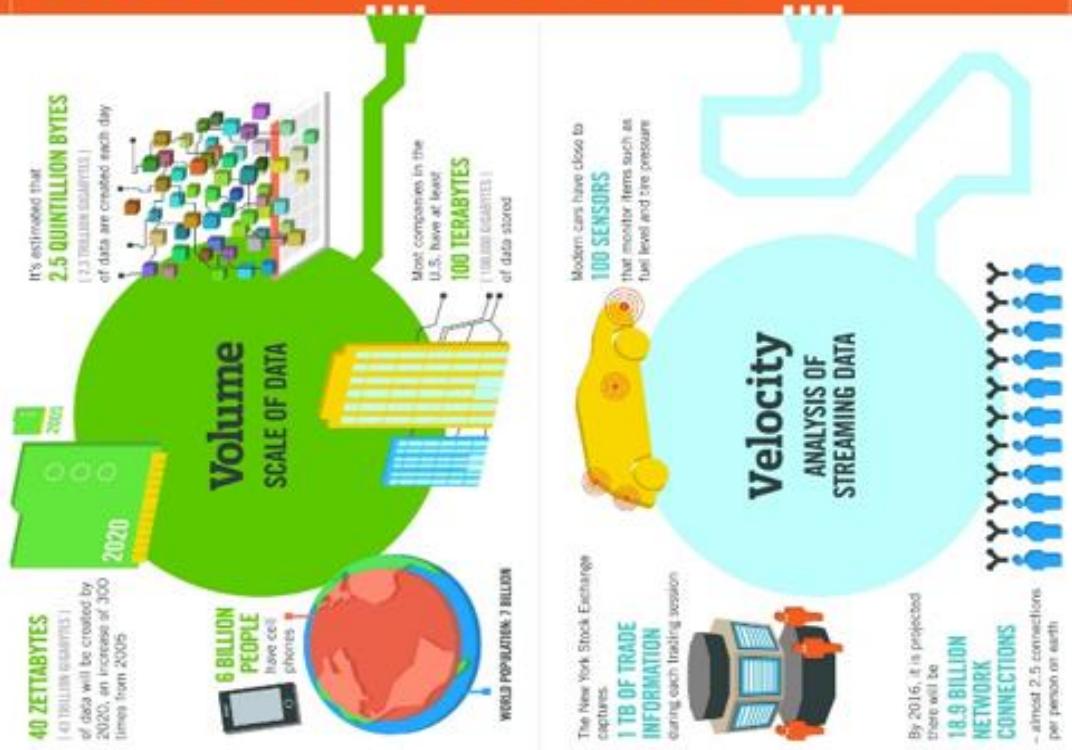
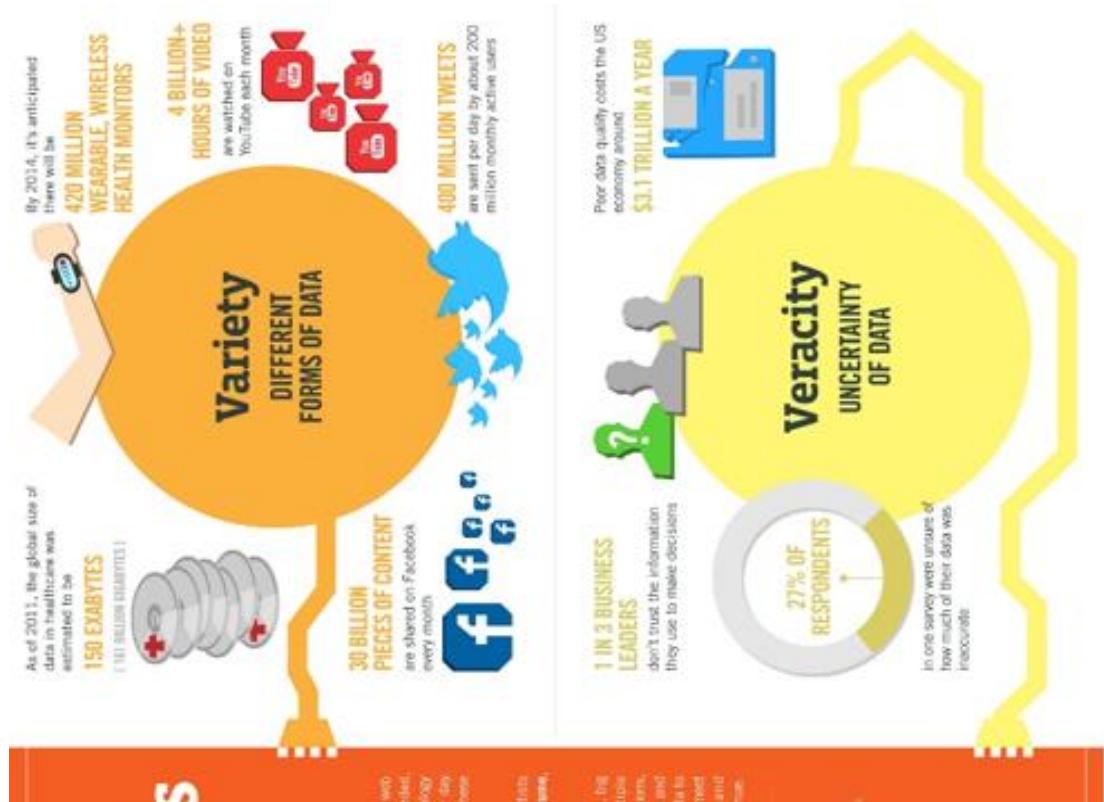


Figura 3: Las cuatro arañas Vs del Big Data [25]

Sin embargo, la saturación de información y su necesidad de procesamiento no solo es un problema exclusivo de nuestros días si no que debemos remontarnos a finales del siglo XIX, poniendo como ejemplo el Censo de los Estados Unidos. En 1881 es el nacimiento de las máquinas de procesamiento de la información de forma rápida y que han evolucionado hasta llegar a las tecnologías de hoy en día [7]. Realmente desde siempre ha habido necesidad de gestionar grandes cantidades de datos, pero es actualmente cuando disponemos de las herramientas útiles y necesarias para su gestión: Bases de datos, volúmenes de memoria, velocidades de procesamiento, software especializado y sobretodo, fácil y rápido acceso a la información mediante Internet...

Ejemplos:

Ejemplos actuales de aplicación de Big Data son: La generación de un mapa marino a partir de algoritmos y técnicas de filtrado aplicadas a 200.000 muestras de sedimentos, dichas muestras se obtuvieron desde los años 50, que se quedaron en 15.000 tras eliminar las no válidas, después se aplicaron un conjunto de algoritmos que permitieron crear un modelo que predice a que clase pertenecen dichas muestras y en que zonas del fondo marino están [8]; Otro ejemplo es la identificación de “cuásares” en la galaxia a través de técnicas de agrupación y decisión aplicadas a terabytes de imágenes obtenidas por el Second Palomar Observatory Sky Survey, una vez recogidas las imágenes el sistema Sky Image Cataloguing and Analysis Tool (SKYCAT) se encargó de clasificar dichas imágenes en estrellas, planetas, galaxias...; El último ejemplo es el desarrollo de sistemas de prevención de lesiones estudiando las características de un jugador como su rendimiento y alimentación aplicando redes neuronales [9], al jugador se le conecta una serie de 24 sensores mientras realiza una serie de ejercicios, dichos sensores envían información por radio a una base de datos y se analizan cada 15 días, este sistema fué creado por Computer Associates International y lo usa el club de fútbol AC Milan. A partir de ahí, las aplicaciones se disparan y encontramos ejemplos en biometría, inteligencia artificial, marketing, seguridad en la red, medicina...Tantos ejemplos como la gran variedad de datos que pueden llegar a existir.

Cuando el Big Data se extiende al ámbito de los negocios, se habla del concepto de Business Intelligence (BI), que se encarga de analizar los datos existentes para resolver problemas relacionados con un negocio, como por ejemplo la rentabilidad de un producto o la planificación de la producción [10].

Big Data en las telecomunicaciones:

Como no podría ser de otro forma, dentro de la infinidad de campos en los que el Big Data se puede aplicar, el ámbito de las telecomunicaciones presenta claros ejemplos como:

-Descubrir los motivos por los que unos determinados clientes abandonaron una compañía, descubriendo que como aspecto común ellos habían recibido pocas promociones y tenido muchas incidencias [9].

-El análisis de la información proveniente de redes de sensores, en la que ya no interviene el ser humano al ser una comunicación del tipo M-2-M (Machine-To-Machine) [11].

-La optimización de rutas y la mejora de la calidad de servicio mediante el análisis de los datos que circulan por la red [12].

-Detección de fraude mediante el análisis de datos relacionados con los datos económicos de fuentes como informes de crédito, como su localización y uso.

Como podemos ver en la **figura 4**, hay muchos servicios relacionados con el uso de Big Data los cuales las empresas de telecomunicaciones están invirtiendo ahora mismo en un alto porcentaje, dicha inversión subirá en los próximos tres años.

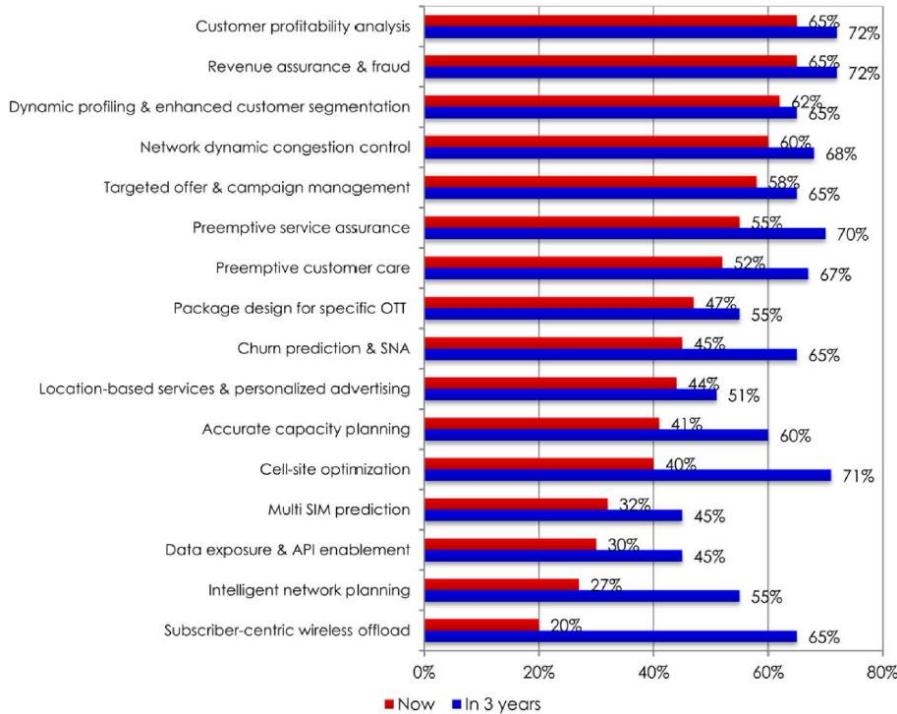


Figura 4: Planes de proveedores de servicios para invertir en escenarios basados en Big Data [15]

Eje x: Porcentaje de inversión Eje y: Servicios que utilizan Big Data

El sector de las telecomunicaciones tiene una gran ventaja con respecto a otros sectores y es que puede extraer datos referidos a sus subscriptores, con esos datos puede crear estrategias que mejoren el servicio que ofrecen a sus clientes [13].

En la **figura 5** podemos ver algunos ejemplos relacionados con el uso que una compañía hace a la información relacionada con un cliente como por ejemplo su geolocalización o las páginas web donde realiza compras. Dicha información es usada para crear campañas personalizadas o para mejorar la lealtad del cliente con la compañía.



Figura 5: Big Data en las telecomunicaciones [26]

Gracias a toda la información una compañía puede ofrecer complementos individualizados a sus clientes, por ejemplo de tipo de deportivo o la incorporación de “audiobooks” a su “servicio”, presentando una mejora en marketing y servicio.

Las compañías de telecomunicaciones disponen de dos tipos de datos relacionados con sus subscriptores [12]:

- Datos internos o estructurados, obtenidos mediante la monitorización del cliente, por ejemplo como y donde usa su teléfono móvil.
- Datos externos o desestructurados, que es información extraída de por ejemplo redes sociales y mensajes de texto.

Los datos estructurados y desestructurados se dividen a su vez en generados en tiempo real (mensajes de texto, mensajes en redes sociales) o estar agrupados (foros online, datos financieros).

En la **figura 6** podemos ver como se relacionan los datos estructurados y no estructurados con los agrupados y los de tiempo real, adicionalmente se nos muestra una gráfica que indica que a mayor variedad y volumen de datos mayor velocidad de generación de los mismos, dicha gráfica también muestra como los desestructurados se generan a mucha mayor velocidad que los estructurados debido a que son de tipos más variados y con más volumen.

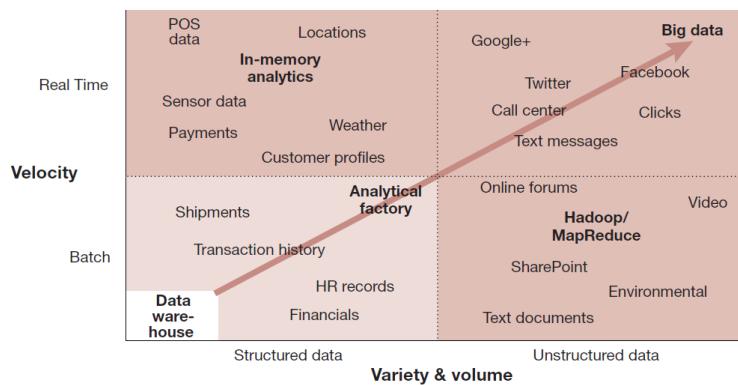


Figura 6: Relación entre tipos de datos y gráfica de velocidad [12]

La incorporación de las redes 4G/LTE, las direcciones tipo IPv6 [13], el uso cada vez más extendido de la “nube” para almacenar datos y el “Internet de las Cosas” [14] facilitan que el volumen de la información sea cada vez más grande por lo que las compañías de

telecomunicaciones necesitan disminuir el coste y mejorar los aspectos de procesado y gestión de los datos que disponen.

La **figura 7** nos da una idea del “grado de importancia” (1 significa que no es importante para nada y 5 que es lo más importante) que un proveedor de servicio da a ciertos aspectos relacionados con los servicios que ofrece en el año 2012 y en el 2014, esta información se recogió de una encuesta realizada a unos 60 proveedores de servicio y parece que simplificar sus operaciones financieras si tiene bastante importancia para ellos [15], esto incluye la mejora de la gestión de los datos de los que disponen.

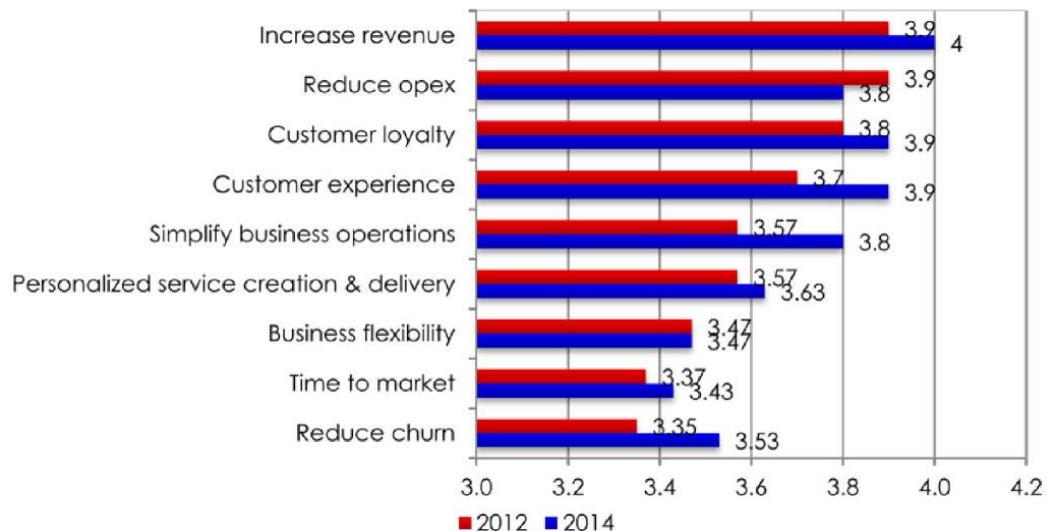


Figura 7: Grado de importancia de servicios [15] Eje x: Grado de importancia Eje y: Servicios

2.2 Cubos/Almacenes de datos

Los almacenes de datos, cuyo nombre en inglés es Datawarehouse, son una gran colección de datos agrupados y ordenados de forma que proporcionan una interfaz adecuada para el acceso a los mismos y, como consecuencia, facilitan la toma de decisiones [16].

Hay muchas formas de organizar la información: Desde el punto de vista del almacenamiento físico ésta puede estar en un servidor o en varios, desde el punto de vista de la información contenida en un datawarehouse, ésta puede separarse en “datamarts” [17], que son almacenes que contienen los datos de un área de negocio específica.

Hay que tener cuidado a la hora del almacenamiento para que la información sea representada de la forma más “ limpia” y ordenada posible, sin valores que puedan resultar confusos.

En un almacén de datos tipo OLAP (Online Analytical Processing) los datos se agrupan de forma multidimensional [18]. En general, los datos pueden presentar relaciones de hasta n dimensiones diferentes, el caso más sencillo, con $n=2$, es solucionado mediante las tradicionales hojas de cálculo y/o tablas. Para $n=3$, se definen los denominados cubos, con campos x, y, z relacionados entre sí. Los campos x e y tendrán una relación directa, se mostrarán valores como resultado de su intersección en una tabla y habrá varias tablas en función de varios valores. Son utilizados cuando la información a organizar es muy numerosa y pueden formar parte de un almacén de datos más grande, como podemos observar la **figura 8**.

En la **figura 8** también podemos ver como distintos tipos de información se organizan en el Data Warehouse y como éste se va dividiendo y subdividiendo en sucesivos Data Marts y Cubos OLAP.

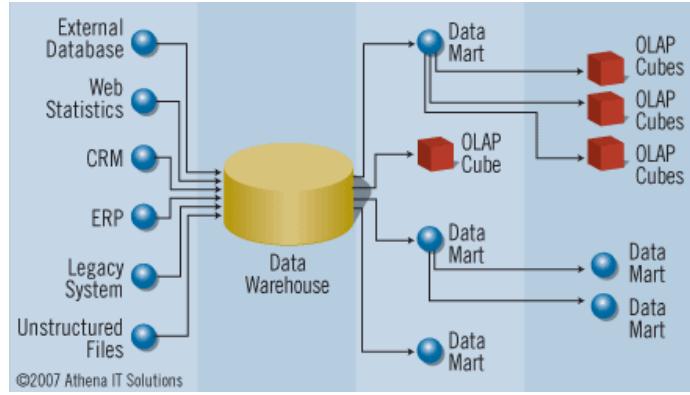


Figura 8: Relación entre Data Warehouse y OLAP [27]

De forma general, se nos mostrarán valores como resultado de la intersección de los campos x, y, z pero con la particularidad de que gráficamente veremos un conjunto de tablas organizadas las cuales llamaremos “láminas del cubo”, que, mediante el filtro z, iremos recorriendo a nuestra voluntad.

La potencialidad del cubo consiste en relacionar los distintos valores que tengamos representados en una tabla de datos externa, identificados cada uno por una columna correspondiente y con esa organización poder trabajar con el conjunto de los datos de forma adecuada.

Por ejemplo tenemos una tabla con los campos de “nombres de productos”, “tipo de producto”, “precio del producto” y “año del producto”. En un cubo podemos establecer “nombres de productos” como eje x (Filas), “tipo de producto” como eje y (Columnas), “precio del producto” como medida y “año del producto” como valor z (filtro). De esta manera tendremos tantas tablas como años se hayan establecido, en cada tabla veremos los productos existentes en cada año y para cada producto veremos los tipos a los que pertenecen, existiendo un precio para cada tipo, así se relacionaran “nombres de productos” y “tipo de producto” dándonos el precio por cada relación nombre/tipo en un año concreto, siendo la relación total año-nombre-tipo-precio.

En la **figura 9** vemos un ejemplo de un cubo OLAP con sus diferentes dimensiones.

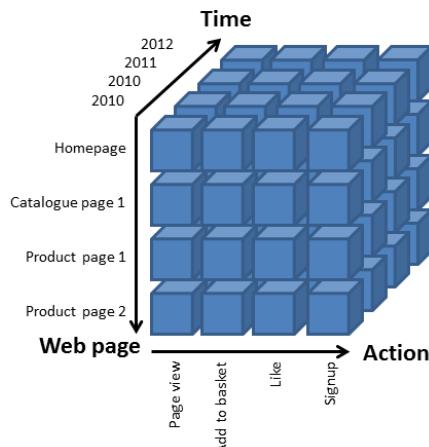


Figura 9: Cubo OLAP [28] Eje x: Web Page, Eje y: Action, Eje z: Time

3. Aspectos Prácticos

En este apartado se hablará de forma muy general de algunas de las herramientas que se emplean para el uso de técnicas de Big Data, Data Mining y para generar los cubos OLAP. Se detallará la herramienta utilizada en este proyecto, justificando el motivo de su uso.

3.1 Herramientas para Big Data, Data Mining y OLAP

La mayoría de estas herramientas son software especializado para trabajar con diferentes tipos de información, sobre todo orientadas al BI (Business Intelligence).

Por ejemplo Pentaho es una plataforma que ofrece un conjunto de programas Open Source orientados a la inteligencia de negocios para, por ejemplo, crear informes o hacer minería de datos [45]. Con las herramientas que ofrece se pueden hacer, por ejemplo, una integración, un análisis visual y un análisis predictivo de los datos [46].

Hadoop es un framework de software para aplicaciones distribuidas bajo una licencia libre, se inspiró en las herramientas de Google: Map Reduce y GFS (Google File System) [47]. Hadoop permite desarrollar tareas intensivas de computación masiva, dividiéndolas en pequeñas piezas y distribuyéndolas en “máquinas” [48].

Otro ejemplo es SSAS (SQL Server Analysis Services), software de Microsoft, que permite crear estructuras multidimensionales desde otros orígenes de datos, como bases de datos relacionales, creadas por ejemplo en Microsoft Access. También permite diseñar, crear y visualizar modelos de minería de datos mediante una gran variedad de algoritmos [49].

Microsoft Excel también ofrece herramientas para la creación de estructuras multidimensionales en base a orígenes de datos externos y su posterior análisis. Mediante alguno de sus complementos se puede, por ejemplo, crear modelos de datos basados en múltiples relaciones entre diversas tablas o crear jerarquías entre los campos de alguna tabla.

3.2 Escenario de aplicación y requerimientos

El escenario de aplicación a utilizar será el software “Microsoft Excel 2013”, en dicho programa estarán representadas las hojas en las que se encuentran las tablas a analizar. Mediante las herramientas y complementos que ofrece organizaremos la información, la analizaremos y trabajaremos con ella.

“Excel” será fuente de información y herramienta de obtención de información, de análisis de información, de manejo de información y de análisis de resultados. Cada cubo generado vendrá representado en una de las hojas en forma de “Tabla dinámica”.

También necesitaremos algunos complementos que utiliza el programa y extienden sus funciones, algunos de ellos ya están incluidos como “PowerPivot” pero para otros es necesario una obtención externa, como es el caso de “OLAP PivotTable Extensions”.

Uno de los motivos del uso de ese escenario es que ofrece las herramientas necesarias para trabajar con grandes cantidades de datos, ya que permite su organización y análisis de forma eficiente.

Se empleará el lenguaje de programación VBA (Visual Basic For Applications) para extraer, mover y modificar los datos en los distintos libros y hojas, siendo la “herramienta” más potente debido a que es la que más opciones nos ofrece, será la más utilizada.

Normalmente para un trabajo de este tipo “Excel” no suele funcionar como origen de datos, dicha función suele estar a cargo de programas como “Microsoft Access” debido a que son orígenes de datos de tipo OLAP y permiten que “Excel” pueda trabajar con los datos mediante técnicas OLAP; Sin embargo, desde el propio Excel podemos conseguir que los datos que se encuentren en él sean considerados de tipo OLAP mediante algunos de los complementos descritos anteriormente, esto permitirá que se pueda trabajar directamente con las tablas con la información “inicial”, representadas en “Excel”, sin necesidad de usar ningún otro programa intermedio, simplificando el desarrollo del proyecto.

Para el desarrollo del proyecto se ha usado Excel en vez de otros programas debido a una mayor costumbre de trabajar con él.

Los requerimientos necesarios para el escenario de aplicación son un ordenador con un sistema “Microsoft Windows” como sistema operativo para que “Microsoft Excel” sea compatible; Es aconsejable que se disponga de unas prestaciones lo más elevadas posible debido a la necesidad del procesamiento y manejo de gran cantidad de datos, eso incluye un procesador rápido y una memoria extensa. En concreto para este proyecto se ha usado un Windows 8, un procesador i7 y una memoria RAM de 12 GB.

4. Desarrollo del proyecto

En este apartado se explicarán los distintos pasos de la realización del proyecto, cada paso estará compuesto por una serie de tareas a realizar, se irán superando objetivos hasta conseguir los objetivos finales, que son la obtención y análisis de diversas tablas de resultados mediante distintos métodos.

4.1 Especificación del problema

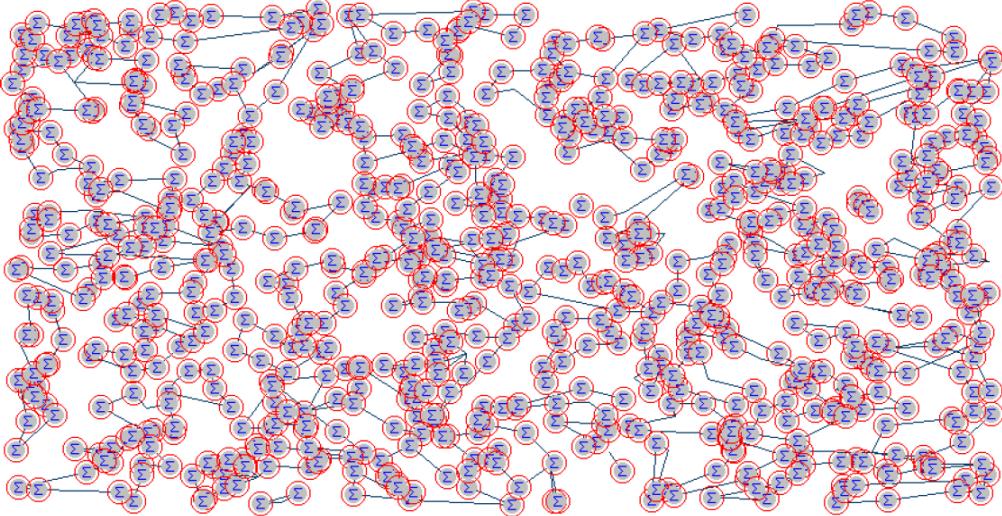
Descripción del problema:

Nuestro problema consiste en que tenemos una gran cantidad de información “desordenada” de dispositivos de telecomunicación y tenemos que hallar la información de su consumo, para ello tenemos que organizar la información de la forma más adecuada posible.

Nuestra información está dividida en dos partes, una formada por dispositivos de tipo lógico (LO) y la otra por dispositivos tipo físico (PH). Cada parte estará compuesta por 23 hojas, una por cada hora del día exceptuando las 8 PM. En cada hoja habrá una tabla representando la información correspondiente a esa hora, en total, disponemos de 46 hojas (23 parte LO + 23 parte PH). Cada parte deberá ser analizada por separado ya que la información que contiene es diferente.

Dicha información procede de la simulación del comportamiento de una red de envergadura nacional bajo distintos escenarios, modificando la cantidad de dispositivos conectados con el objetivo de medir la variación de su consumo energético y determinando el efecto que tiene la

cantidad de usuarios conectados con el uso de energía. En particular, la red utilizada como base corresponde a un anillo de 5 niveles con una envergadura aproximada de 40000 Km² [50], la **figura 10** nos da una visión general de dicha red.



La **figura 11** nos da idea del consumo energético de la red:

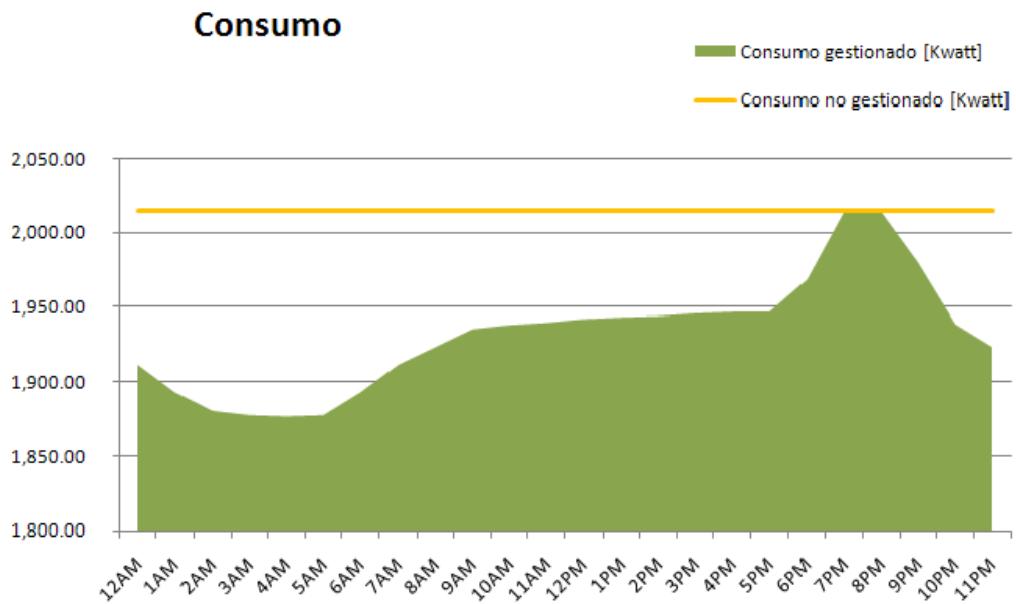


Figura 11: Consumo Energético (en KWatts) [50]

Posteriormente, en la realización del proyecto, podremos obtener el consumo de uno solo de los dispositivos.

Parte LO:

En la siguiente imagen (**figura 12**) vemos una hoja que muestra los datos a las 12AM.

Figura 12: Hoja con datos a las 12AM (LO)

Como podemos observar en la imagen, la primera columna representa la información del nombre del dispositivo en concreto, teniendo en total 100 filas, sin embargo, algunos vienen representados por valores 0 o en blanco, tendremos que tener cuidado con estos valores más adelante.

En detalle, los dispositivos existentes son:

Tipo de dispositivo			
ESW (Ethernet Switch)	LC (Lineal Card)	LER (Label Edge Router)	LSR (Label Switching Router)
ESW1, ESW1a, ESW2, ESW3, ESW4.	LC7-LC18, LC33-LC43, LC58-LC68.	LER1, LER2.	LSR1, LSR2, LSR3.

ESW, LER y LSR representan el “chasis”, cada uno tiene una serie de “LCs” asignadas.

Cada uno tiene asignada una información referida a los anchos de banda, los tipos de consumos, el número de puertos, precios por puerto, número de tarjetas lógicas o número de interfaces físicas, representadas mediante las columnas de la tabla, en total tenemos 24 columnas, hay hasta cinco niveles en los cuales tenemos Number of LC y Number of PIUs, siendo los niveles grupos de nodos distintos.

En la página representada disponemos de campos con cálculos, el Total de Number of LC y Number of PIU y Consumo Watt/hr. Dichos campos contienen fórmulas que operan con valores de distintas columnas de la tabla, dichas fórmulas serán empleadas cuando manejemos el cubo.

Parte PH:

En las siguientes imágenes (**figura 13** y **figura 14**) vemos un ejemplo de información a las 12AM.

Figura 13: Hoja con datos a las 12AM (PH) 1

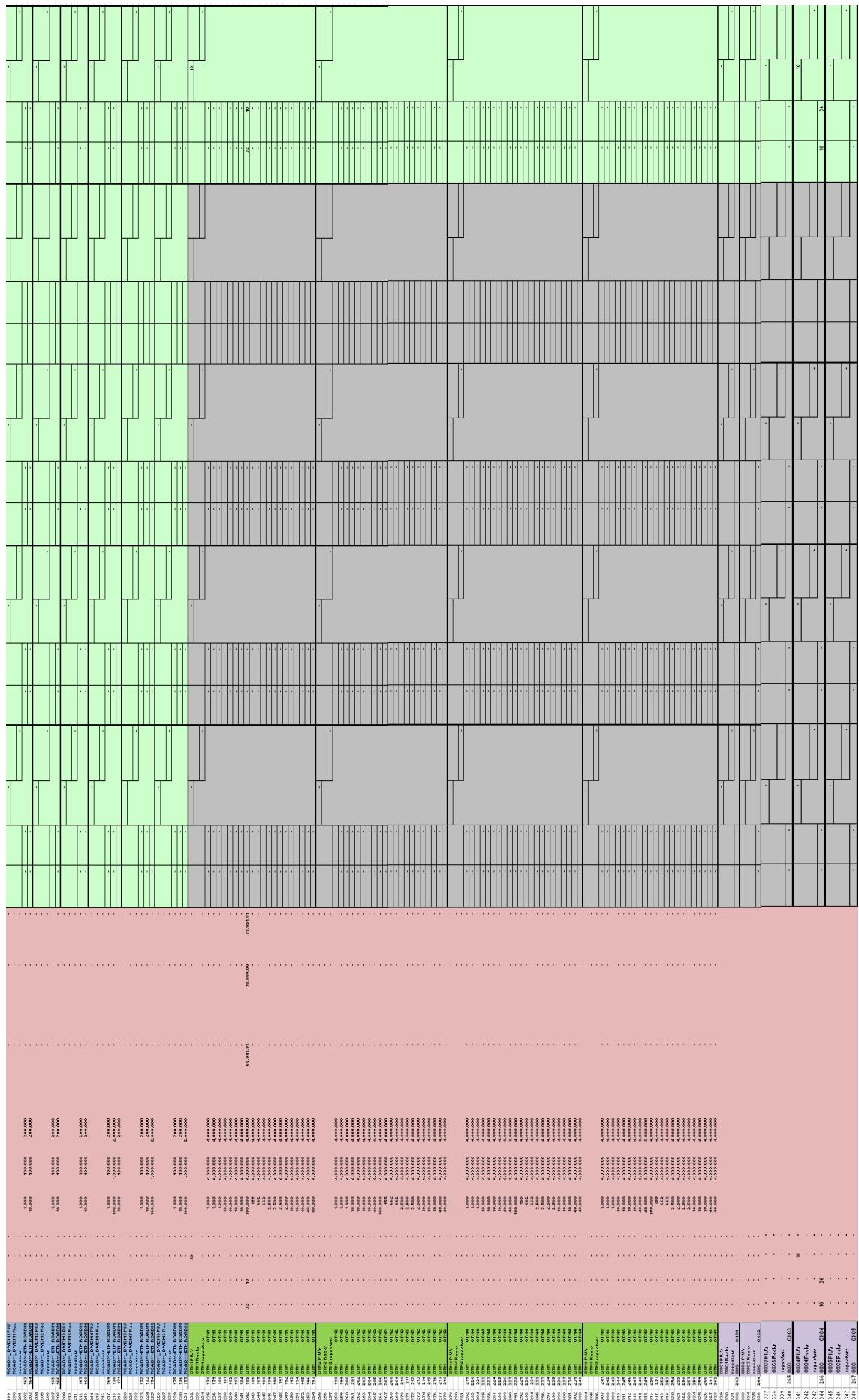


Figura 14: Hoja con datos a las 12AM (PH) 2

Esta parte, a diferencia de la LO, contiene muchas más filas y columnas.

Existen 344 filas con distintos elementos, cada elemento estará no solo identificado por una columna, si no por 4 columnas diferentes referidas a cuatro aspectos diferentes: Conf_Index, Technology, PIU_Index y Type.

Los identificadores son:

Conf_Index	Technology	PIU_Index	Type
"en blanco", 1-267.	ADM, OADM, ROADM-OTN, ROADM-ETH, OTN, OXC, ROADM-ETH, ROADM-OTN.	1-26.	OTN1, OTN1_OTU_2, OTN2, OTN2_OTU_3, OTN4, OTN4_OTU_3, OTN8_, OTN8_OTU_4, OXC1, OXC2, OXC3, OXC4, OXC5, ROADM_CWDM, ROADM_DWDM1, ROADM_DWDM2, ROADM_DWDM3, ROADM_DWDM4, ROADM_DWDM5, ROADM_DWDM6, STM1, STM16, STM256, STM4, STM64.
	<u>PIUs, Racks,</u> <u>repeaters:</u> OTN1, OTN1_OTU_2, OTN2, OTN2_OTU_3, OTN4, OTN4_OTU_3, OTN8, OTN8_OTU_4, STM1, STM4, STM16, STM64, STM256. <u>PIUs, Racks:</u> OXC1, OXC2, OXC3, OXC4, OXC5, ROADM_CWDM, ROADM_DWDM1, ROADM_DWDM2, ROADM_DWDM3, ROADM_DWDM4, ROADM_DWDM5, ROADM_DWDM6.		

Aunque existan valores en blanco en alguno de los identificadores, no significa que los elementos sean "no válidos" si algún otro de los tres identificadores restantes no lo están. Dichos valores aparecen en el campo Conf_Index y representan la información de PIUs, Racks y

repeaters que hay en la red de un Type determinado (Ejemplo: OTN1_OTU_2 PIUs tiene 656 Number of PIU a las 12AM).

La información relacionada con cada elemento viene en las columnas de la tabla, presentando algunas similitudes con la parte LO: Número de puertos, tipo de puertos, número de slots, precios, anchos de banda, consumos, número de tarjetas lógicas L, número de tarjetas lógicas R, número de interfaces físicas y número de repetidores, en total tenemos 50 columnas, como en el caso LO también tenemos 5 niveles para cada Number of....

Los cálculos vienen representados por las columnas Number of L LC, Number of R LC, Number of PIU's, Number of repeaters, Consumo L LC, Consumo R LC (sin fórmula), Consumo PIU, Consumo Watt/hr, al igual que en la parte LO, las fórmulas descritas nos serán útiles más adelante.

En resumen, tenemos una gran cantidad de información que procesar y analizar.

4.2 Organización de los datos

Una vez que hemos examinado los datos en detalle, pasamos a su organización, para ello se crearán una serie de tablas que contendrán el conjunto de los elementos. Se organizará la información referida a la parte LO y la parte PH en dos tablas distintas que agrupen su información. El motivo de hacer esto es poder recoger los datos de la forma más simplificada posible en apartados posteriores, es decir, realizar un pre-procesado de los datos.

Agrupar todos los datos de cada una de las tablas manualmente sería una tarea demasiado laboriosa por lo que se empleará un par de programas escritos en VBA para recoger los datos y almacenarlos correctamente, habrá un programa que se encargue de la parte LO y otro que se encargue de la parte PH. Dichas tablas estarán en un libro de Excel diferente al del conjunto de tablas iniciales.

Parte LO:

```
Workbooks.Open Filename:="C:\Users\Jesús\Desktop\resultados_detalle.xlsx"
Workbooks("resultados_detalle.xlsx").Activate 'Abrimos y activamos el libro donde se encuentran las hojas con información.
'Obtención de los nombres de las columnas y los dispositivos

Workbooks("resultados_detalle.xlsx").Worksheets(p(0)).Activate 'Activación de la primera hoja establecida en el array "12AM_LO".
s = recol_LO() 'Obtención del nombre de las columnas (no todas, solo las necesarias).
t = recnomb_LO() 'Obtención del nombre de los dispositivos.

ThisWorkbook.Worksheets("Tabla_LO").Activate 'Activación de la hoja la cual estará la tabla total.

Call barra(1) 'Barra de progreso.

Call intcol_LO(s) 'En la hoja destino introducimos el nombre de las columnas.
```

Figura 15: Código RecData_LO 1

Abrimos el libro con las hojas a extraer información y activamos la primera hoja, representada como el primer valor del vector p ("12AM_LO"). El vector s se encargará de extraer el nombre de las columnas de la tabla que nos interese, no todos los nombres de todas las columnas, el vector t se encargará de recorrer las filas extrayendo únicamente la información del nombre de los dispositivos, ubicados en la primera columna.

Las columnas se han elegido en función de las que necesitamos para las operaciones de nuestros cálculos, no obteniendo las que no forman parte de las operaciones, esto se ha hecho así para evitar un tamaño de las tablas demasiado grande.

Para extraer esta información se podría haber activado cualquier hoja del libro ya que la información de los dispositivos y los nombres de las columnas son iguales en todas ellas.

Después activamos la hoja destino e insertamos los valores de las columnas; El nombre de la primera columna, la del nombre de los dispositivos, se introducirá manualmente con el nombre de “tipo de disp.”, al igual que el nombre de las horas.

```

f1 = 2 'Índices que representan la primera fila y la última del bloque de datos en una sola hora.
f2 = 102

For y = 0 To 22 'Ciclo que irá recogiendo y depositando datos a medida hasta recorrer todas las horas.

    Workbooks("resultados_detalle.xlsx").Worksheets(p(y)).Activate 'Activamos la hoja correspondiente para extraer sus datos.

    x = recinf_LO() 'Recogemos la información más importante, la numérica referida al número de puertos, el consumo, número de tarjetas.

    ThisWorkbook.Worksheets("Tabla_LO").Activate 'Activación de hoja adecuada para inserción de datos, en el libro destino,
    'el cuál ejecutará este código.

    Call intinf_LO(f1, f2, t, x, q, y) 'Función más relevante, introducimos en nuestra tabla destino los valores de los dispositivos.
    'Los valores asignados a cada dispositivo y la hora correspondiente,
    'todo con los índices necesarios para establecer el lugar de la tabla y extraer la hora correcta.

    'Actualizar índices para ir recorriendo nuestra tabla, los índices van representando los valores límite de cada fila en cada bloque,
    'siendo las columnas constantes.
    f1 = f1 + 101
    f2 = f2 + 101

Next y

```

Figura 16: Código RecData_LO 2

En la siguiente parte del código iremos recorriendo todas las hojas origen de datos y extrayendo los valores numéricos de las columnas correspondientes para cada una de las filas, después iremos depositando dicha información en una tabla aparte junto con la información del nombre de los dispositivos, tantas veces como horas tenemos, de esta manera iremos almacenando las hojas “una encima de la otra”. La inserción de la hora que representa cada fila de nuestra tabla destino se introducirá manualmente mediante el vector q, vector que contiene la información de las horas, en la última columna de la tabla.

```

Call barra(2) 'Barra de progreso.

Call ceros_LO

'En esta parte crearemos una hoja nueva con la tabla original filtrada con valores no deseados eliminados.

Call filtrar(q, s) 'Necesitamos q y s ya que son valores que permanecen igual en la tabla sin filtrar y en la filtrada.

Call barra(3) 'Barra de progreso.

Workbooks("resultados_detalle.xlsx").Close 'Cerramos el libro que contiene tablas.

```

Figura 17: Código RecData_LO 3

Los valores que estén en blanco se les dará un valor 0 para evitar errores en los cálculos posteriores.

Una vez tenemos la tabla ya completa, pasamos a eliminar aquellos valores que no nos va a interesar procesar en apartados posteriores, es decir, los valores con nombre 0 y “en blanco”, para ello usamos una función que nos filtre la tabla anteriormente creada y nos muestre otra con los valores no deseados eliminados, esta tabla sustituirá a la anterior como tabla de datos de la parte LO.

(Para ver la tabla de resultados sin filtrar ver anexo sección Tabla total LO y para ver la tabla de resultados filtrada ver anexo sección Tabla filtrada LO).

Parte PH:

```
Workbooks.Open Filename:="C:\Users\Jesús\Desktop\resultados_detalle.xlsx"
Workbooks("resultados_detalle.xlsx").Activate 'Activamos el libro el cuál contiene las hojas origen de las tablas.
Workbooks("resultados_detalle.xlsx").Worksheets(p(0)).Activate

s = recol_PH() 'Obtención de los nombres de las columnas.
t = recnomb_PH() 'Obtención del "nombre" de los dispositivos, que en este caso vienen identificado por cuatro columnas,
'es decir, por dispositivo disponemos de cuatro identificadores.

'Introducción de dichos nombres en las hojas que tenemos, esta vez no usaremos una subrutina como en el anterior ejemplo.
Call intcol_PH(s)
```

Figura 18: Código RecData_PH 1

Esta parte realiza lo mismo que la parte LO, con la diferencia de que extraeremos valores de cuatro columnas diferentes como identificadores de los elementos para el vector t, también obtenemos el nombre de las columnas de los identificadores, cosa que no hicimos en la parte LO al estar en blanco y tuvimos que introducir manualmente.

```
f1 = 2 'Rangos de filas de la hoja nueva de datos, son mayores que las tablas LO.
f2 = 346

Call barra(1)

For y = 0 To 22 'Iremos recorriendo todas las horas.

    Workbooks("resultados_detalle.xlsx").Worksheets(p(y)).Activate 'Activamos la hoja correspondiente.

    x = recinf_PH() 'Recoger información numérica.

    ThisWorkbook.Worksheets("Tabla_PH").Activate 'Activación de hoja adecuada para inserción de datos, en el libro adecuado.

    Call intinf_PH(f1, f2, t, x, q, y) 'Introducir la información en cada bloque de la tabla, dicha información incluye los identificadores
    'y los valores numéricos.

    f1 = f1 + 345 'Actualización de indices.
    f2 = f2 + 345

    Next y

    Call barra(2)

    Call ceros_PH

    Call barra(3)

Workbooks("resultados_detalle.xlsx").Close 'Cerramos el libro que contiene tablas.
```

Figura 19: Código RecData_PH 2

En la siguiente parte obtenemos los valores y los insertamos en la hoja destino, juntando cada una de las tablas en una sola, esta vez no necesitamos filtrar la tabla ya que no hay elementos no válidos; Aunque haya algún campo de los identificadores en blanco, alguno de los otros tres no lo están, como lo explicado anteriormente.

(Para ver la tabla de resultados PH consultar anexo sección Tabla total PH).

Diagrama de flujo y esquema:

Para entender mejor los programas utilizados se muestran un diagrama de flujo y un esquema (**figura 20** y **figura 21**), el diagrama da más idea de los pasos de desarrollo del programa mientras que el esquema da más idea de cómo se relacionan las diferentes funciones del programa.

(Para consultar los códigos completos de los programas consultar anexo sección RecData_LO y RecData_PH).

Diagrama de flujo:

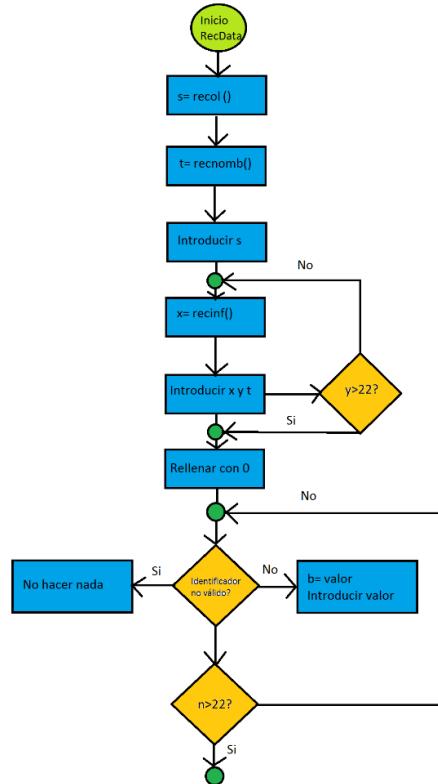


Figura 20: Diagrama de flujo RecData

Esquema del programa:

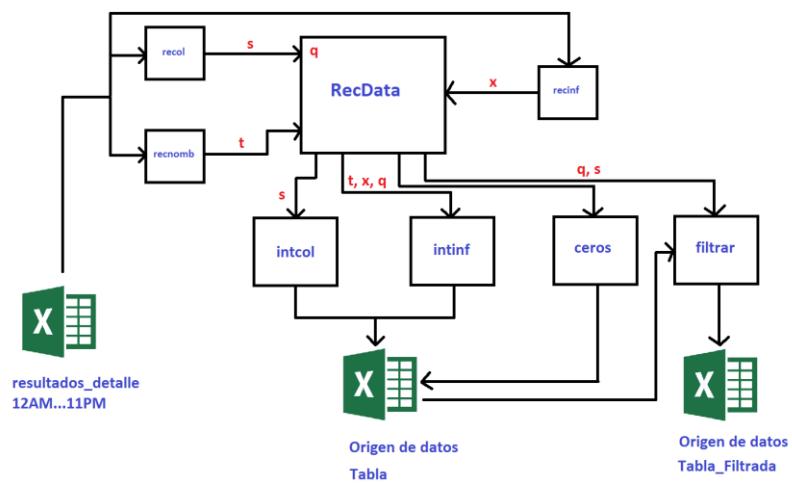


Figura 21: Esquema del programa recdata

Una vez realizados estos pasos nuestros datos ya están organizados y preparados para ser procesados, dicho libro funcionará como origen de datos aunque los datos contenidos se hayan extraído de otra fuente.

4.3 PowerPivot

PowerPivot es un complemento de Excel que permite un análisis de datos eficaz y crear modelos de datos sofisticados [19]. Lo usaremos únicamente para crear los cubos OLAP y realizar alguna operación (como por ejemplo la suma de LCs).

Insertar Tablas:

Abrimos otro libro de Excel y abrimos el complemento PowerPivot, este complemento nos permitirá importar las tablas LO y PH para que estén contenidas en él. Este paso es necesario como paso “intermedio” para que nuestro origen de datos sea considerado de tipo OLAP, esto permitirá aplicar ciertas técnicas que de otra forma no sería posible. Despu s crearemos el cubo con dichas tablas.

En la **tabla 2** y la **tabla 3** vemos un fragmento de ellas.

Tabla 2: Ejemplo de tabla PowerPivot LO

Tabla 3: Ejemplo de tabla PowerPivot PH

Una de las características de PowerPivot es que permite la creación de nuevas columnas, llamadas columnas calculadas, que permite representar valores como resultado de las operaciones entre otras columnas de la tabla, utilizando expresiones de tipo DAX (Data Analytical eXpressions).

DAX es una biblioteca de funciones, operadores y constantes que se pueden combinar para crear fórmulas y expresiones con PowerPivot; Ofrece una sintaxis para una serie de funciones de tipo matemático, lógico, de filtrado, estadístico, de texto y de fecha y hora y una sintaxis para representar los elementos de las tablas [20] como el nombre de las columnas o el nombre de la propia tabla en sus operaciones.

Operaciones secundarias:

Antes de crear el cubo directamente podemos realizar alguna de las operaciones que estaban definidas en las tablas iniciales, aprovechando la potencialidad del DAX, dichas operaciones deben de tener la característica de ser operaciones sencillas. Operaciones más complejas utilizarán el cubo para su realización y se servirán de las columnas originales y las nuevas calculadas, que formarán parte del cubo. Las siguientes figuras nos muestran las fórmulas en diferentes formatos.

a) Parte LO:

Fórmulas LO:

$$\text{Suma de LC} = \text{Number of LC} + \text{Number of LC1} + \text{Number of LC2} + \text{Number of LC3} + \text{Number of LC4} + \text{Number of LC5}$$

Suma de PIUs = Number of PIUs + Number of PIUs1 + Number of PIUs2 + Number of PIUs3 + Number of PIUs4 + Number of PIUs5

a) En Excel:

=SUMA(M3;O3;Q3;S3;U3;W3)

=SUMA(N3;P3;R3;T3;V3;X3)

Figura 22: Operaciones secundarias LO Excel

b) En DAX:

$=[\text{Number of LC}]+[\text{Number of LC1}]+[\text{Number of LC2}]+[\text{Number of LC3}]+[\text{Number of LC4}]+[\text{Number of LC5}]$
 $=[\text{Number of PIUs}]+[\text{Number of PIUs1}]+[\text{Number of PIUs2}]+[\text{Number of PIUs3}]+[\text{Number of PIUs4}]+[\text{Number of PIUs5}]$

Figura 23: Operaciones secundarias LO DAX

En las fórmulas tipo DAX especificamos el nombre de las columnas en vez de las celdas, como es el caso de Excel. La columna calculada se auto rellena sola sin necesidad de ir “fila por fila”.

En la **tabla 4** vemos un fragmento de un par de columnas calculadas

Tabla 4: Ejemplo de columnas calculadas LO

Suma de LCs	Suma de PIUs
0	989
0	991
0	993
0	993
0	993
0	993
0	993
0	993
0	993
0	989
0	989
36	0
2	0
14	0
4	0
4	0
2	0
12	0
12	0
12	0
4	0
4	0
4	0
4	0
12	0
36	0

b) Parte PH:

Veremos las fórmulas representadas en distintos formatos.

Fórmulas PH:

$$\begin{aligned} \text{Suma de L LC} = & \text{Number of L LC1} + \text{Number of L LC2} + \text{Number of L LC3} \\ & + \text{Number of L LC4} + \text{Number of L L5} \end{aligned}$$

$$\begin{aligned} \text{Suma de R LC} = & \text{Number of R LC1} + \text{Number of R LC2} + \text{Number of R LC3} \\ & + \text{Number of R LC4} + \text{Number of R L5} \end{aligned}$$

$$\begin{aligned} \text{Suma de PIUs} = & \text{Number of PIUs1} + \text{Number of PIUs2} + \text{Number of PIUs3} \\ & + \text{Number of PIUs4} + \text{Number of PIUs5} \end{aligned}$$

Suma de repeaters

$$\begin{aligned} &= \text{Number of repeaters1} + \text{Number of repeaters2} \\ &+ \text{Number of repeaters3} + \text{Number of repeaters4} \\ &+ \text{Number of repeaters5} \end{aligned}$$

Caso Consumo por tarjeta = 0

$$\text{Consumo L LC} = (\text{BW L LC})^{2/3} * \text{Nº of L Ports} * \text{Suma de L LC}$$

Caso Consumo por tarjeta distinto de 0:

$$\text{Consumo L LC} = \text{Consumo por tarjeta} * \text{Suma de L LC}$$

$$\text{Consumo PIU} = \text{Suma de R LC} * 1000$$

a) En Excel:

```
=SUMA(AS4;AO4;AK4;AG4;AC4)  
=SUMA(AT4;AP4;AL4;AH4;AD4)  
=SUMA(AU4;AQ4;AM4;AI4;AE4)  
=SUMA(AV4;AR4;AN4;AJ4;AF4)  
  
=POTENCIA(U7;2/3)*H7*Q7  
=X91*Q91  
  
=R4*1000
```

Figura 24: Operaciones secundarias PH Excel

```
=[Number of L LC]+[Number of L LC1]+[Number of L LC2]+[Number of L LC3]+[Number of L LC4]  
=[Number of R LC]+[Number of R LC1]+[Number of R LC2]+[Number of R LC3]+[Number of R LC4]  
=[Number of PIUs]+[Number of PIUs1]+[Number of PIUs2]+[Number of PIUs3]+[Number of PIUs4]  
=[Number of repeaters]+[Number of repeaters1]+[Number of repeaters2]+[Number of repeaters3]+[Number of repeaters4]  
  
=IF([Conf_Index]=70||[Conf_Index]=71||[Conf_Index]=72||[Conf_Index]=73||[Conf_Index]=74||[Conf_Index]=75;  
([Suma de L LC]*[Consumo por tarjeta]);[Nº of L ports]*[Suma de L LC]*POWER([BW L LC Mbps];2/3))
```

=[Suma de R LC]*1000

Figura 25: Operaciones secundarias PH DAX

Usamos una expresión tipo IF debido a que en una misma columna hay varias operaciones, especificamos las filas cuyas operaciones son diferentes y el resto lo asignamos a una sola operación.

En la **tabla 5** vemos algunas de las columnas calculadas mediante las fórmulas anteriores referidas a la parte PH.

Tabla 5: Ejemplo de columnas calculadas PH

Suma de L LC	Suma de R LC	Suma de PIUs	Suma de repeaters	Consumo L LC	Consumo PIU
44	10	0	0	94795,1263614029	10000
10	36	0	0	0	36000
44	10	0	0	94795,1263614029	10000
10	36	0	0	0	36000
44	10	0	0	94795,1263614029	10000
10	36	0	0	0	36000
44	10	0	0	94795,1263614029	10000
10	36	0	0	0	36000
44	10	0	0	94795,1263614029	10000
10	36	0	0	0	36000
44	10	0	0	94795,1263614029	10000
10	36	0	0	0	36000
44	10	0	0	94795,1263614029	10000
10	36	0	0	0	36000
52	10	0	0	112030,603881658	10000
10	36	0	0	0	36000
52	10	0	0	112030,603881658	10000
10	36	0	0	0	36000
44	10	0	0	94795,1263614029	10000
10	36	0	0	0	36000
40	10	0	0	86177,3876012753	10000
10	36	0	0	0	36000
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

Una vez creadas las columnas calculadas pasamos a la creación del cubo mediante la opción de “Tabla Dinámica”.

4.4 Creación del cubo

La tabla dinámica aparecerá vacía en una de las hojas del libro, cuando la seleccionamos nos aparece el nombre de las tablas contenidas en PowerPivot, tendremos que seleccionar únicamente la tabla LO para el cubo LO y la tabla PH para el cubo PH; Tendremos también cuatro campos que representan las dimensiones del cubo: “Filas”, “Columnas”, “Filtros” y “Valores”. En filas y columnas se establecen los identificadores/etiquetas de nuestra tabla, en z el filtro para ir recorriendo las tablas y en valores aquellos valores que queremos identificar en cada tabla para cada fila y cada columna, se realizará una operación automáticamente con el conjunto de los valores de una columna, la suma o la media (entre otros) según se decida.

Esta parte requerirá que la persona que diseña el cubo sepa relacionar adecuadamente sus campos. En la **figura 26** vemos la tabla dinámica vacía y las tablas que pueden formar parte de ella (parte superior derecha de la imagen).

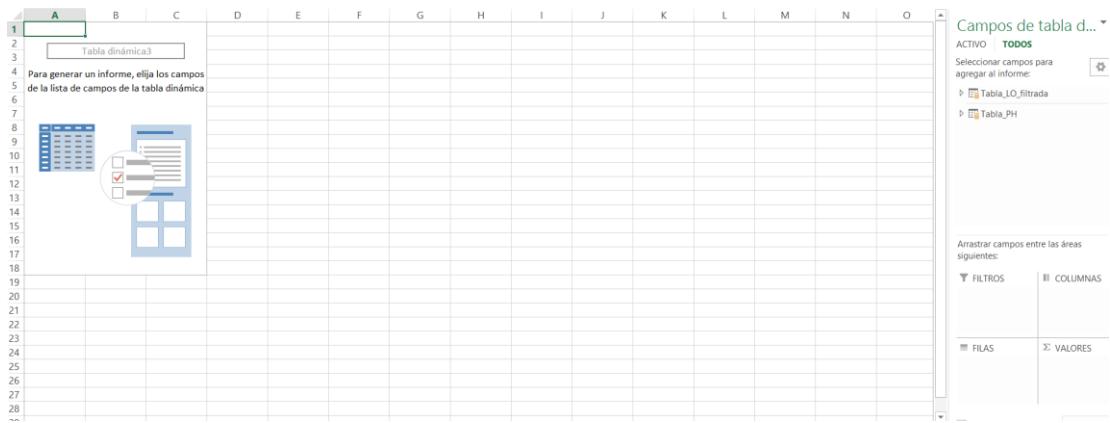


Figura 26: Tabla dinámica vacía

Los campos del cubo estarán formados por las columnas de las tablas anteriormente creadas (LO y PH). Por ejemplo en la tabla LO había una columna que se llamaba tipo de disp. y otra que se llamaba hora, pues tipo de disp. podemos colocarla como la fila de nuestro cubo y hora como el filtro, luego había otras columnas como Number of LC o Ports per LC, pues esas columnas van hacia el área de valores. En la **figura 27** vemos un ejemplo de los campos asignados.

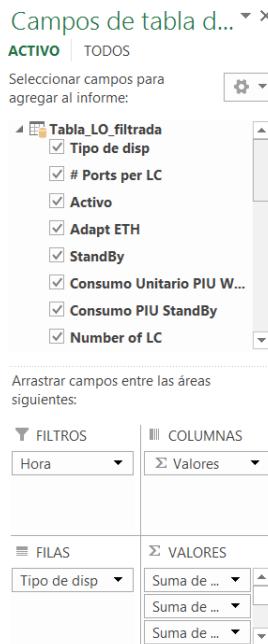


Figura 27: Ejemplo de selección de campos de un cubo

Un aspecto importante es que para valores repetidos en la tabla de un identificador/etiqueta concreto, en el cubo se suman los valores que tenga asignados para cada hora (o valor en filtro), eliminando la repetición.

Poniendo un ejemplo, si tenemos el conjunto de valores “camisa-50\$-2010” y “camisa-20\$-2010”, en la tabla “filtro=2010” nos aparecerá un único valor de fila “camisa” con el valor correspondiente de 70\$.

Cubo LO:

El primer cubo que crearemos usará la información de la tabla LO. Para seleccionar los campos tenemos que tener en cuenta de que valores queremos obtener la información, es decir tener una idea clara de los “identificadores” (etiquetas) de nuestros valores siguiendo una lógica, que en este caso es “tipo de disp.”, que serán las filas. El filtro serán las horas y el resto de valores de las tablas serán las columnas.

No estableceremos columnas “identificadoras” en los cubos debido a que la información que representa una columna simboliza una división de los valores establecidos en una fila y eso haría que el cubo sería demasiado grande, además que para eso tendríamos que tener más identificadores, cosa que no ocurre para el caso LO. Las columnas serán los valores del resto de campos de las tablas.

Para entender mejor esto, pongamos de ejemplo que tenemos un campo de fila llamado “camisa” identificado por “ropa” y existen dos campos de columnas identificados por “tipo” que son “tipo 1” y “tipo 2”, para cada uno tenemos valores identificados por “precio”, tendremos un precio para “tipo 1” y otro para “tipo 2” pero si omitimos la columna “tipo” tendremos el precio total de todos los tipos para “camisa”, quedándonos una única columna de “precio”.

El resultado final de nuestro cubo es (**figura 28**):

Figura 28: Tabla dinámica 40

En la **figura 29** vemos el cubo con sus correspondientes dimensiones de forma tridimensional:

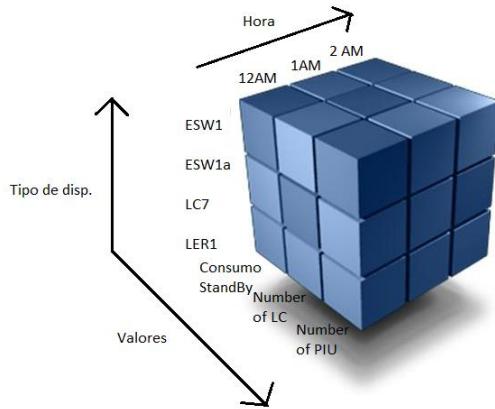


Figura 29: Cubo OLAP LO Eje x: Tipo de disp. Eje y: Valores Eje z: Hora

Cubo PH:

Para el cubo tipo PH creamos otra tabla dinámica, esta vez tenemos hasta cuatro posibles valores como identificadores.

Esto permite una extensa combinación de identificadores como campos de fila, columna o filtro, es decir, podemos tener “Technology” como filtro y “Type” como fila, o “Conf_Index” como fila y “Technology” como columna e incluso crear “relaciones anidadas”, que son valores de columna/fila dentro de un valor de columna/fila, por ejemplo “Conf_Index” como identificador de fila, “Technology” como primer identificador de columna y “Type” como segundo identificador de “columna”, esto generaría valores diferentes para un “Conf_index”, un “Technology” y un “Type” determinado, existiendo varios “Type” por “Technology” y un solo “Conf_index” para cada “Technology-Type”. En definitiva, otra forma de organizar y visualizar los datos.

Sin embargo, como tenemos que hacer cálculos posteriores, para facilitar una interfaz adecuada para que dichos cálculos se sirvan de la información del cubo, se usará únicamente “Conf_index” como campo de filas, se elige “Conf_index” ya que es el que mejor identifica los elementos de la red individualmente y, lo que es más importante, no repite ninguno de sus valores en las tablas iniciales, excepto el valor “en blanco” (excepcional al representar los PIU, Rack y repeaters de los Type de la red), es decir, es el que mejor identifica la información (los elementos/dispositivos) para su análisis en detalle. El filtro será “Hora”. Las columnas estarán compuestas por los demás valores de la tabla. **La figura 30 y 31** nos muestran la tabla dinámica creada.

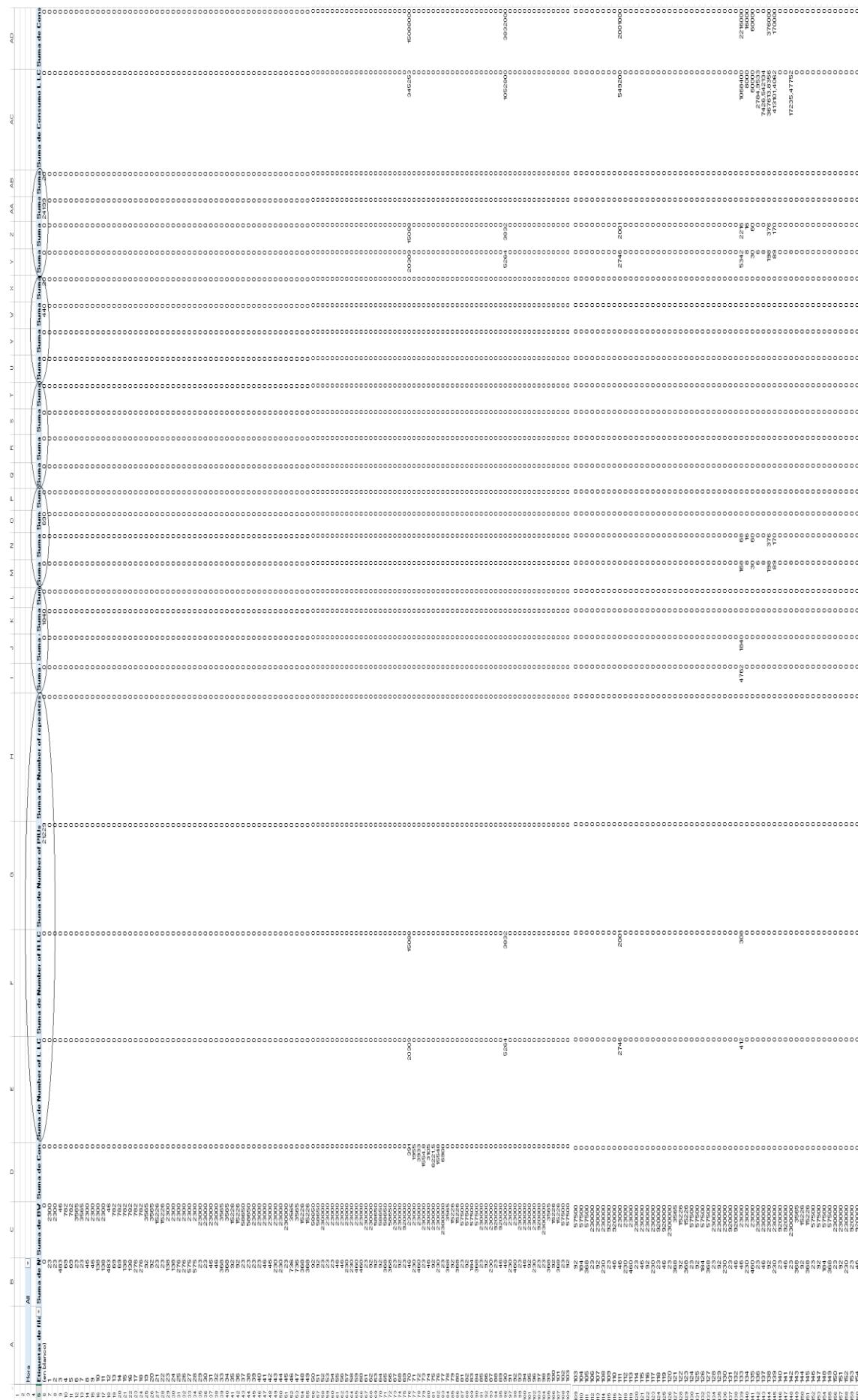


Figura 30: Tabla dinámica PH1

Figura 31: Tabla dinámica PH 2

En la la **figura 32** vemos el cubo OLAP PH desde un punto de vista tridimensional.

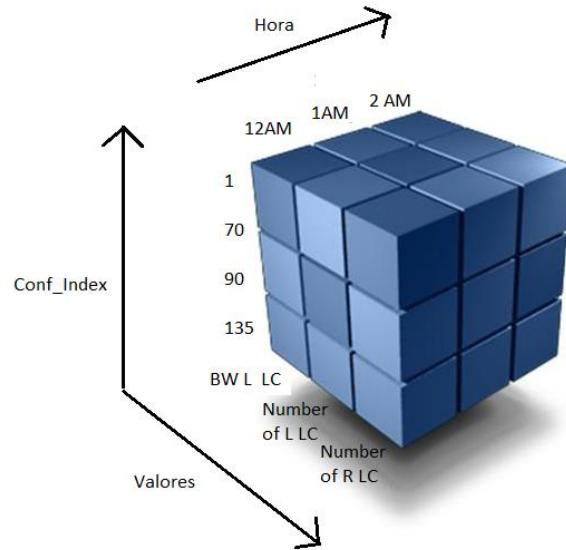


Figura 32: Cubo OLAP PH Eje x: Conf_Index Eje y: Valores Eje z: Hora

De esta forma hemos conseguido que las hojas que antes estaban separadas en varias, una por hora, estén ahora juntas en un único cubo.

Segmentación y actualización de datos:

Otra opción que ofrece la tabla dinámica es la “segmentación de los datos”, un equivalente a la opción filtro, con una diferencia de tipo visual, en la **figura 33** vemos un ejemplo.

	A	B	C	D	E	F	G	H	I
1	Hora	00:00							
2	Type	All							
3	Technology	(Varios elementos)							
4									
5	Etiquetas de fila								
6									
7	1	1	100	0					
8	2	1	100	0					
9	3	21	2	0					
10	4	3	34	0					
11	5	3	34	0					
12	6	1	195	0					
13	7	1	195	0					
14	8	2	100	0					
15	9	2	100	0					
16	10	6	100	0					
17	11	6	100	0					
18	12	21	2	0					
19	13	3	34	0					
20	14	3	34	0					
21	15	6	34	0					
22	16	6	34	0					
23	17	12	34	0					
24	18	12	34	0					
25	19	4	195	0					
26	20	4	195	0					
27	21	1	682	0					
28	22	1	682	0					
29	23	6	100	0					
30	24	6	100	0					
31	25	12	100	0					
32	26	12	100	0					
33	27	25	100	0					
34	28	25	100	0					
35	29	1	1000	0					
36	30	1	1000	0					
37	31	2	1000	0					
38	32	2	1000	0					
39	33	16	195	0					
40	34	16	195	0					

Figura 33: Ejemplo de Segmentación de datos

Ésta opción puede tener una serie de propiedades: Un nombre distinto al original (“Technology”), varios filtros simultáneos, conexión con varios cubos...

Por último imaginémonos que se ha producido un cambio en uno de los datos en las tablas originales LO y PH, ¿Se reflejaría dicho cambio en el cubo? Automáticamente no, habría que seleccionar la opción de “Actualizar”, ubicado en la sección “Analizar” dentro de “Herramientas de tabla dinámica” entonces nuestros datos nuevos ya se verían reflejados en el cubo.

4.5 Análisis del cubo

Una herramienta para el análisis de los cubos es “OLAP PivotTable Extensions” [21], para abrirla, pinchamos con el botón derecho en la tabla dinámica y la seleccionamos, esta herramienta la podemos usar porque nuestro origen de datos es de tipo OLAP, debido a que creamos la tabla dinámica con PowerPivot.

MDX:

El primer campo es una consulta de tipo MDX (MultiDimensional eXpressions) que está utilizando el cubo OLAP sobre el “servidor OLAP”, MDX es un lenguaje de consultas aplicado a bases de datos multidimensionales [22].

En la **figura 34** vemos un ejemplo de la consulta MDX del cubo LO en la lámina 12AM.



The screenshot shows the 'OLAP PivotTable Extensions' application window. The main area is a text editor containing an MDX query. The query is as follows:

```

MDX | Search | Filter List | Defaults | About
MDX Query: Format MDX query using web service? 
SELECT
{ [Measures].[Suma de # Ports per LC], [Measures].[Suma de Activo], [Measures].[Suma de Adapt_ETHERNET], [Measures].[Suma de StandBy], [Measures].[Suma de Consumo Unitario PIU Watt], [Measures].[Suma de Consumo PIU StandBy], [Measures].[Suma de Number of LC], [Measures].[Suma de Number of PIUs], [Measures].[Suma de Number of LC1], [Measures].[Suma de Number of PIUs1], [Measures].[Suma de Number of LC2], [Measures].[Suma de Number of PIUs2], [Measures].[Suma de Number of LC3], [Measures].[Suma de Number of PIUs3], [Measures].[Suma de Number of LC4], [Measures].[Suma de Number of PIUs4], [Measures].[Suma de Number of LC5], [Measures].[Suma de Number of PIUs5], [Measures].[Suma de Suma de LCs], [Measures].[Suma de Suma de PIUs] } DIMENSION PROPERTIES PARENT_UNIQUE_NAME
, MEMBER_VALUE
, HIERARCHY_UNIQUE_NAME ON COLUMNS
, NON EMPTY Hierarchize (
{
  DrilldownLevel (
    { [Tabla_LO_filtrada].[Tipo de disp].[All] }
  ,
  ,
  ,
  , INCLUDE_CALC_MEMBERS
  )
}
) DIMENSION PROPERTIES PARENT_UNIQUE_NAME
, MEMBER_VALUE
, HIERARCHY_UNIQUE_NAME ON ROWS
FROM [Model]
WHERE ( [Tabla_LO_filtrada].[Hora].&[1899-12-30T00:00:00] ) CELL PROPERTIES VALUE
, FORMAT_STRING
, LANGUAGE
, BACK_COLOR
, FORE_COLOR
, FONT_FLAGS

```

Figura 34: MDX

Esta información será muy importante cuando manejemos el cubo ya que incluye el nombre de identificadores que necesitaremos (“[Tabla_LO_filtrada].[Hora].&[1899-12-30T00:00:00]”).

Search:

El siguiente campo es el de “Search” o búsqueda, en el que podemos buscar cualquier valor contenido en el cubo, elegimos que tipo de valor queremos buscar y le escribimos, seleccionando el campo correspondiente a un tipo de dato (Medida calculada, lista de campo...). Podemos buscar varios valores con letras en común, por ejemplo si introducimos “Number” nos saldrán “Number of LC”, “Number of PIU”... Los valores disponibles a buscar estarán contenidos en cualquiera de las tablas con las cuales se pueda formar el cubo, no solo las existentes en el cubo creado. Podemos seleccionar los valores que queramos de cualquier dimensión y que nos aparezcan como únicos valores en las filas del cubo (con sus respectivas columnas), por lo que también tiene función de filtro.

En la **figura 35** vemos la búsqueda de campos en el cubo que tengan LC como parte del nombre que les identifica.

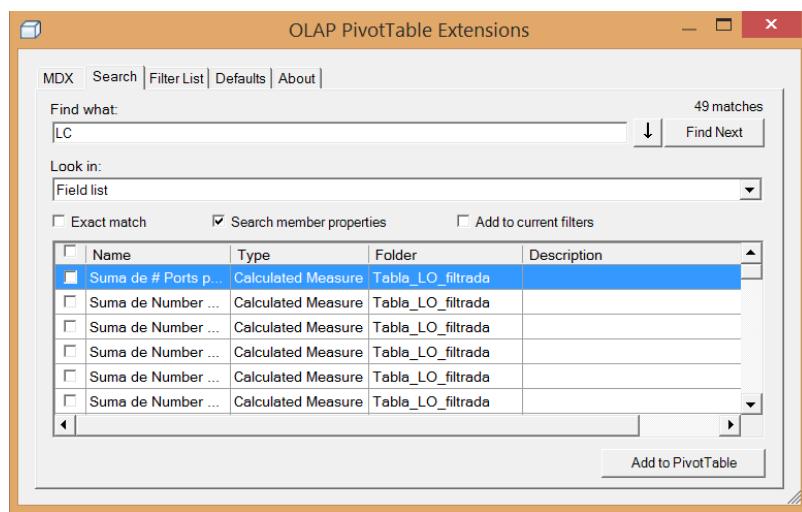


Figura 35: Search

Filter:

En el apartado Filter podemos buscar los elementos que queramos de un campo del cubo ubicado en filas o filtros y que la tabla nos muestre únicamente los valores asignados a dicho valor, esta función es similar a la del apartado Search con la diferencia de que Filter se restringe a los valores de las dimensiones del cubo (filas, columnas o filtros).

En la **figura 36** vemos como establecer los filtros del cubo mediante dos elementos del campo filas.

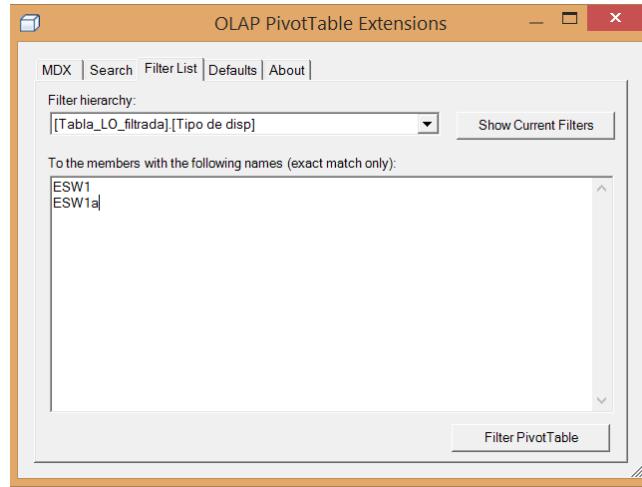


Figura 36: Filter

Default:

El apartado Default nos permite seleccionar una serie de opciones: Mostrar miembros calculados del servidor OLAP y actualizar los datos cuando se abra el archivo. En la **figura 37** vemos la ventana que se nos muestra al seleccionar dicha opción.

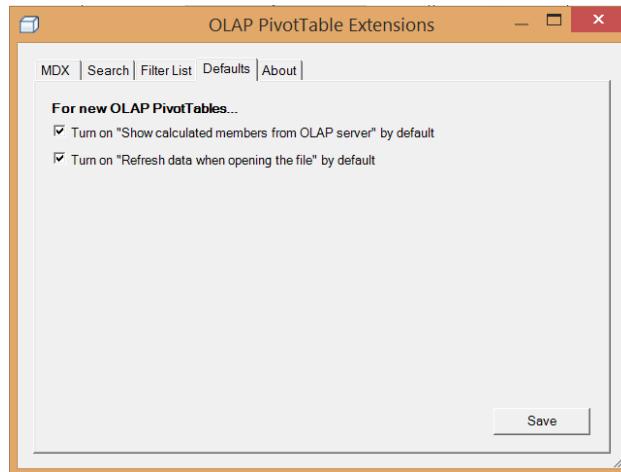


Figura 37: Defaults

“OLAP PivotTable Extensions” es especialmente útil cuando queremos obtener información detallada acerca del cubo no solo de los elementos que lo componen, si no del lenguaje de consulta que utiliza, además de que también ofrece funciones de búsqueda y filtro.

4.6 Manejo del cubo

En este apartado veremos cómo manejar el cubo para poder realizar operaciones con sus elementos. Veremos tres formas diferentes de hallar un mismo resultado final.

Parte LO:

Formulas LO:

Dependiendo del dispositivo, en función del grupo que pertenezca, tendremos una fórmula diferente, hasta un total de tres fórmulas. Realizaremos la operación principal, hallar los consumos.

$$\begin{aligned} ESW &= \text{Suma de PIUs} * \text{Consumo Unitario PIU Watt} \\ &\quad + (\text{Suma de PIUs (7 PM)} - \text{Suma de PIUs}) \\ &\quad * \text{Consumo Unitario PIU StandBy} \end{aligned}$$

$$\begin{aligned} LC &= \text{Ports Per LC} * \text{Activo} * \text{Suma de LC} + (\text{Suma de LC (7 PM)} - \text{Suma de LC}) \\ &\quad * \text{StandBy} \end{aligned}$$

$$\begin{aligned} LER y LSR &= \text{Suma de PIUs} * \text{Consumo Unitario PIU Watt} \\ &\quad + (\text{Suma de PIUs (7 PM)} - \text{Suma de PIUs}) \\ &\quad * \text{Consumo Unitario PIU StandBy} \end{aligned}$$

Las 7PM es la hora cargada, eso significa que refleja el número total de LCs y PIUs encendidos, la resta con respecto a otras horas muestra el número de elementos en StandBy en la hora determinada. Se calcula el consumo de los elementos encendidos más el consumo de los elementos en StandBy.

a) VALORCUBO:

Para esta parte usaremos una función de Excel que trabaja con cubos llamada VALORCUBO.

VALORCUBO extrae un valor del cubo que nosotros elijamos, en cualquier parte y lámina del cubo y lo deposita en una celda elegida.

Para poder utilizarlo, es necesario usar una de las opciones OLAP dentro del apartado herramientas OLAP llamado reconvertir en fórmulas, esta opción no habría sido posible si no habríamos creado la tabla dinámica desde PowerPivot, hecho esto, aparece representado en los elementos del cubo su “fórmula” equivalente, de esta forma en celdas aparte podemos hacer las operaciones extrayendo los datos con dicha función. En la **figura 38** vemos un ejemplo.

=VALORCUBO("ThisWorkbookDataModel";"[Tabla_LO_filtrada].[Hora].[All].[0:00:00]";\$A4; U\$3)

Figura 38: Expresión individual de VALORCUBO

El primer campo representa la conexión al cubo, en el segundo estableceremos la lámina del cubo de donde queremos extraer el valor, usando el identificador correcto, aunque en principio parece que hay muchas maneras de especificar este campo ("[Tabla].[Hora].&[19:00:00]", "[Tabla].[Hora].[19:00:00]"...) hay que tener cuidado de que el identificador establecido es el correcto: "[Tabla].[Hora].[All].[19:00:00]", la propia interfaz de Excel proporciona ese identificador; El tercer campo representa la fila donde se encuentra el valor, la letra asignada representa donde está ubicada la primera columna del cubo; Por último el cuarto campo representa la columna donde se encuentra el valor, siendo la letra la columna correspondiente y el número la fila donde está ubicado el nombre de la columna, este valor permanece fijo para todos los valores de una columna.

VALORCUBO extraerá valores de una lámina del cubo concreta excepto uno de ellos, que extraerá siempre el valor de las 19:00:00, así para cada hora. En la **figura 39** vemos una fórmula formada por valores extraídos con VALORCUBO de una celda determinada.

=VALORCUBO("ThisWorkbookDataModel";"Tabla_LO_filtrada").[Hora].[All].[0:00:00]"\$A4; \$U3)*VALORCUBO("ThisWorkbookDataModel";"Tabla_LO_filtrada").[Hora].[All].[0:00:00]"\$A4; \$S3)+VALORCUBO("ThisWorkbookDataModel";"Tabla_LO_filtrada").[Hora].[All].[19:00:00]"\$A4; \$U3)-VALORCUBO("ThisWorkbookDataModel";"Tabla_LO_filtrada").[Hora].[All].[0:00:00]"\$A4; \$U3)+VALORCUBO("ThisWorkbookDataModel";"Tabla_LO_filtrada").[Hora].[All].[0:00:00]"\$A4; \$S3)*VALORCUBO("ThisWorkbookDataModel";"Tabla_LO_filtrada").[Hora].[All].[0:00:00]"\$A4; \$U3)

Figura 39: Ejemplo de VALORCUBO (LO)

b) VBA:

El siguiente método usa el lenguaje VBA para extraer los datos del cubo y realizar las operaciones, para luego insertarlas en una hoja aparte.

```
'Activamos la lámina de cubo correspondiente a las 7PM.
ThisWorkbook.Worksheets("Cubo_LO").Activate
ThisWorkbook.Worksheets("Cubo_LO").PivotTables("Tabla dinámica2").PivotFields("Tabla_LO_filtrada").[Hora].CurrentPageName = "[Tabla_LO_filtrada].[Hora].4[1899-12-30T19:00:00]"

Call barra(1)

vsiete = vseiete_LO() 'Extraemos los valores de Suma de LC y Suma de PIU debido a que les necesitaremos posteriormente.

z1 = 0 'Iniciarizar indices de los vectores resultado.
z2 = 0
z3 = 0

c = True 'Variable booleana que distinguirá entre los valores de las decenas 1 o 2.
w = 0 'Iniciarizar decenas.
y = 0 'Iniciarizar unidades.

Call barra(2)
```

Figura 40: Código Calculos_LO 1

En esta parte almacenaremos los valores correspondientes a dos columnas de las 7PM (19:00:00) debido a que para realizar los cálculos posteriores necesitaremos dichos valores para cada lámina del cubo.

Previamente tenemos que activar la hoja asignada a las 19:00:00, para ello tenemos que conocer los identificadores OLAP del campo “Hora” y del campo “19:00:00”; Para el primero se fue recorriendo el “vector” PivotFields(i) y representado en pantalla hasta que nos mostró el identificador del campo “Hora” (“[Tabla_LO_filtrada].[Hora].[Hora]”) mientras que para el segundo se usó la herramienta “OLAP PivotTable Extensions” donde venía reflejado dicho identificador en la sección MDX (Ver parte del código MDX WHERE..., cerca del final de dicho código en la **figura 34**). Si nuestra tabla dinámica no sería de tipo OLAP, la forma de recorrer las láminas no sería la misma, por ejemplo habría otros identificadores.

```

For k = 1 To 23 'Iteración para cada hora, es decir, cada lámina del cubo.
    Call Chequeo(w, y, c) 'Comprobamos los valores de la hora que establecemos, para comprobar si tenemos que aumentar las decenas
    'y reiniciar las unidades.
    'Teniendo en cuenta que las 20:00 no existe en nuestro cubo.

    'Activamos la lámina del cubo correcta mediante w e y, formarán parte de la identificación de la lámina,
    'que cambiará en cada iteración.
    'Convirtiéndose en un valor tipo texto para que pueda formar parte de la identificación.

    ThisWorkbook.Worksheets("Cubo_LO").Activate
    ThisWorkbook.Worksheets("Cubo_LO").PivotTables("Tabla dinámica2").PivotFields("[Tabla_LO_filtrada].[Hora].[Hora]").CurrentPageName =
    Call Operaciones_LO(q1, q2, q3, z1, z2, z3, vsiete) 'Realizar las operaciones adecuadas.

    y = y + 1 'Aumentar unidades.

    Next k

    Call insval_LO(q1, q2, q3) 'Establecer nuestros valores en una tabla aparte, será la tabla de resultados.
    Call barra(3)

```

Figura 41: Código calculos_LO 2

Después de CurrentPageName (Identificador extraído de “OLAP PivotTable Extensions”, parte MDX):

```
"[Tabla_LO_filtrada].[Hora].&[1899-12-30T" + CStr(w) + CStr(y) + ":00:00]"
```

Figura 42: Identificador OLAP 1

En la siguiente parte iremos recorriendo las láminas y realizando las operaciones de cada dispositivo, para activar las láminas usaremos dos variables (decenas y unidades) que irán desde el “00” al “21”, las unidades se actualizarán cada iteración del ciclo mientras que las decenas se actualizarán mediante una función. Estas variables formarán parte del identificador OLAP del filtro que seleccionemos y permitirán que la activación de las hojas sea a la par que las iteraciones del ciclo donde realizaremos las operaciones.

Una vez realizadas las operaciones la insertaremos en la hoja resultado correspondiente.

c) DAX:

En este método no trabajaremos sobre el cubo, si no sobre las tablas en PowerPivot. Realizaremos la operación mediante el lenguaje DAX creándonos una columna calculada que contendrá los valores resultado del consumo.

Uno de los principales problemas es que la operación requiere datos que son exclusivamente de las 19:00:00, y DAX solo opera directamente con columnas y no con bloques de datos (En este caso, bloques de datos a las 19:00:00), para solucionar este problema necesitamos crearnos una columna en la que cada dispositivo tenga el valor correspondiente a las 19:00:00 de las columnas necesarias (Por ejemplo “Suma de LCs”) para cada repetición (una por hora) en la tabla.

En los siguientes ejemplos (**figura 43** y **figura 44**) vemos fragmentos del código DAX para las columnas calculadas, el primer fragmento se encarga de insertar los valores de Suma de LCs a las 7PM de los dispositivos, tenemos que especificar cada dispositivo por separado por lo que la tarea es larga y laboriosa, el siguiente fragmento es parte de la operación del consumo, para este caso tendremos hasta tres operaciones diferentes, con una sentencia IF definimos cada operación según el dispositivo elegido.

(Para consultar los códigos completos consultar anexo sección código DAX parte Suma de LCs 7PM y Consumos Watt/hr_LO).

```

= IF (
    [Tipo de disp] = "ESW1";
    LOOKUPVALUE (
        [Suma de LCs];
        [Tipo de disp]; "ESW1";
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
    IF (
        [Tipo de disp] = "ESW1a";
        LOOKUPVALUE (
            [Suma de LCs];
            [Tipo de disp]; "ESW1a";
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Tipo de disp] = "ESW2";
        LOOKUPVALUE (
            [Suma de LCs];
            [Tipo de disp]; "ESW2";
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Tipo de disp] = "ESW3";
        LOOKUPVALUE (
            [Suma de LCs];
            [Tipo de disp]; "ESW3";
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Tipo de disp] = "ESW4";
        LOOKUPVALUE (
            [Suma de LCs];
            [Tipo de disp]; "ESW4";
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );

```

Figura 43: Ejemplo de código DAX "7PM" Parte LO

```

=IF (
    [Tipo de disp] = "ESW1"
    || [Tipo de disp] = "ESW1a"
    || [Tipo de disp] = "ESW2"
    || [Tipo de disp] = "ESW3"
    || [Tipo de disp] = "ESW4";
    [Suma de PIUs]
    * [Consumo Unitario PIU Watt]
    + ( [Suma de PIUs 7PM] - [Suma de PIUs] )
    * [Consumo PIU StandBy];
    IF (
        [Tipo de disp] = "LC7"
        || [Tipo de disp] = "LC8"
        || [Tipo de disp] = "LC9"
        || [Tipo de disp] = "LC10"
        || [Tipo de disp] = "LC11"
        || [Tipo de disp] = "LC12"
        || [Tipo de disp] = "LC13"
        || [Tipo de disp] = "LC14"
        || [Tipo de disp] = "LC15"
        || [Tipo de disp] = "LC16"
        || [Tipo de disp] = "LC17"
        || [Tipo de disp] = "LC18"
        || [Tipo de disp] = "LC19"
        || [Tipo de disp] = "LC33"
        || [Tipo de disp] = "LC34"
        || [Tipo de disp] = "LC35"
        || [Tipo de disp] = "LC36"
        || [Tipo de disp] = "LC37"
        || [Tipo de disp] = "LC38"
        || [Tipo de disp] = "LC39"
        || [Tipo de disp] = "LC40"
        || [Tipo de disp] = "LC41"
        || [Tipo de disp] = "LC42"
        || [Tipo de disp] = "LC43"

```

Figura 44: Ejemplo de operación DAX parte LO

Una vez que tenemos la columna de resultados, para visualizarla correctamente, pasamos a Excel. Copiamos las columnas que formarán nuestra tabla de resultados: "Tipo de disp.", "Hora" y "Consumo Watt/hr" y organizamos los valores para ordenarlos después en otra tabla: Las horas las ordenamos de la 0 a las 23 y los dispositivos por orden alfabético.

En la **tabla 6** vemos una parte de la tabla LO desordenada:

Tabla 6: Ejemplo de tabla DAX LO

	A	B	C
1	Tipo de disp	Hora	Consumo Watt/Hr
2	ESw1	30/12/1899 0:00:00	495700
3	ESw1	30/12/1899 1:00:00	495700
4	ESw1	30/12/1899 2:00:00	495700
5	ESw1	30/12/1899 3:00:00	495700
6	ESw1	30/12/1899 4:00:00	495700
7	ESw1	30/12/1899 5:00:00	495700
8	ESw1	30/12/1899 6:00:00	495700
9	ESw1	30/12/1899 7:00:00	495700
10	ESw1	30/12/1899 8:00:00	495700
11	ESw1	30/12/1899 9:00:00	495700
12	ESw1	30/12/1899 10:00:00	495700
13	ESw1	30/12/1899 11:00:00	495700
14	ESw1	30/12/1899 12:00:00	495700
15	ESw1	30/12/1899 13:00:00	496100
16	ESw1	30/12/1899 14:00:00	496500
17	ESw1	30/12/1899 15:00:00	496500
18	ESw1	30/12/1899 16:00:00	496500
19	ESw1	30/12/1899 17:00:00	496500
20	ESw1	30/12/1899 18:00:00	496500
21	ESw1	30/12/1899 19:00:00	496500
22	ESw1	30/12/1899 21:00:00	496500
23	ESw1	30/12/1899 22:00:00	495700
24	ESw1	30/12/1899 23:00:00	495700
25	ESw1a	30/12/1899 0:00:00	0
26	ESw1a	30/12/1899 10:00:00	0

Ahora con un programa VBA ordenamos los datos en otra tabla aparte:

```
'Recogida de dispositivos
ThisWorkbook.Worksheets("Tabla_DAX_LO").Activate 'Activamos la tabla que contiene los valores a extraer.
Call barra(1) 'Barra de progreso.

s = Recdisp_LO() 'Extraemos los nombres de los dispositivos.
x = Recinf_LO() 'Recogemos los consumos.

ThisWorkbook.Worksheets("Cálculos_DAX_LO").Activate 'Activamos la hoja resultado.
Call barra(2) 'Barra de progreso.

Call insinf_LO(s, x, q) 'Insertamos la información
Call barra(3) 'Barra de progreso.

End Sub
```

Figura 45: Código Calculos_DAX_LO

Extraemos primero los nombres de los dispositivos, luego los consumos por dispositivo y luego les ordenamos todos en una única tabla recorriendo cada fila y cada columna.

(Para consultar el código completo que ordena la tabla de resultados ver anexo sección calculos_DAX_LO).

Tendremos una tabla resultado con tantas filas como dispositivos (44) y columnas como horas (23).

(Para consultar la tabla de resultados generada ver anexo sección Tabla de resultados LO).

d) Gráficas de algunos los elementos:

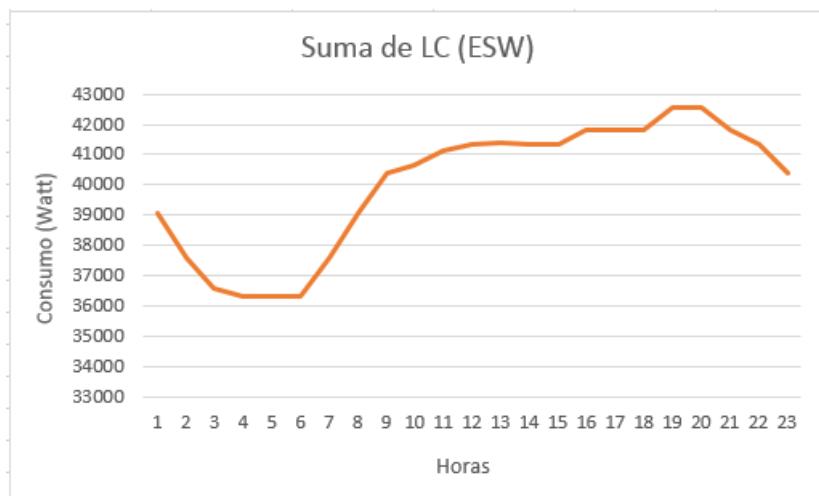
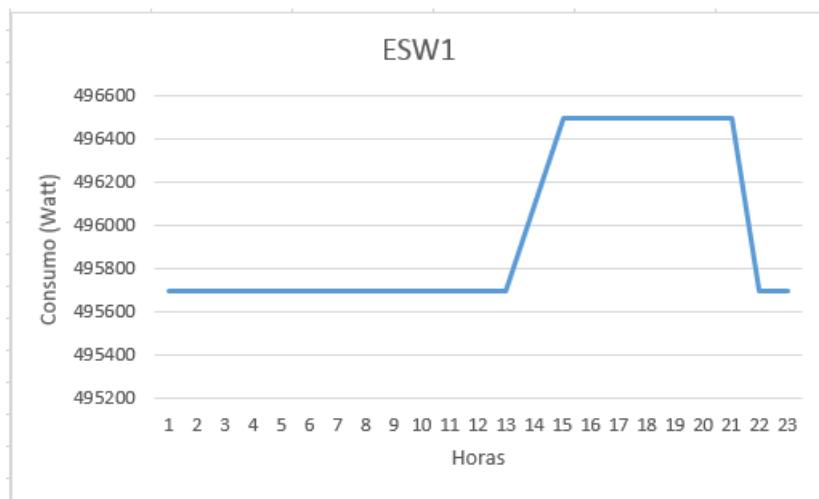


Figura 46: Ejemplo dispositivos ESW (Consumo total)

Entre las 13 y las 21 horas se produce una subida del consumo en ESW1. En la suma de sus LCs se ve una progresiva subida desde las 6 con un descenso a partir de las 20.

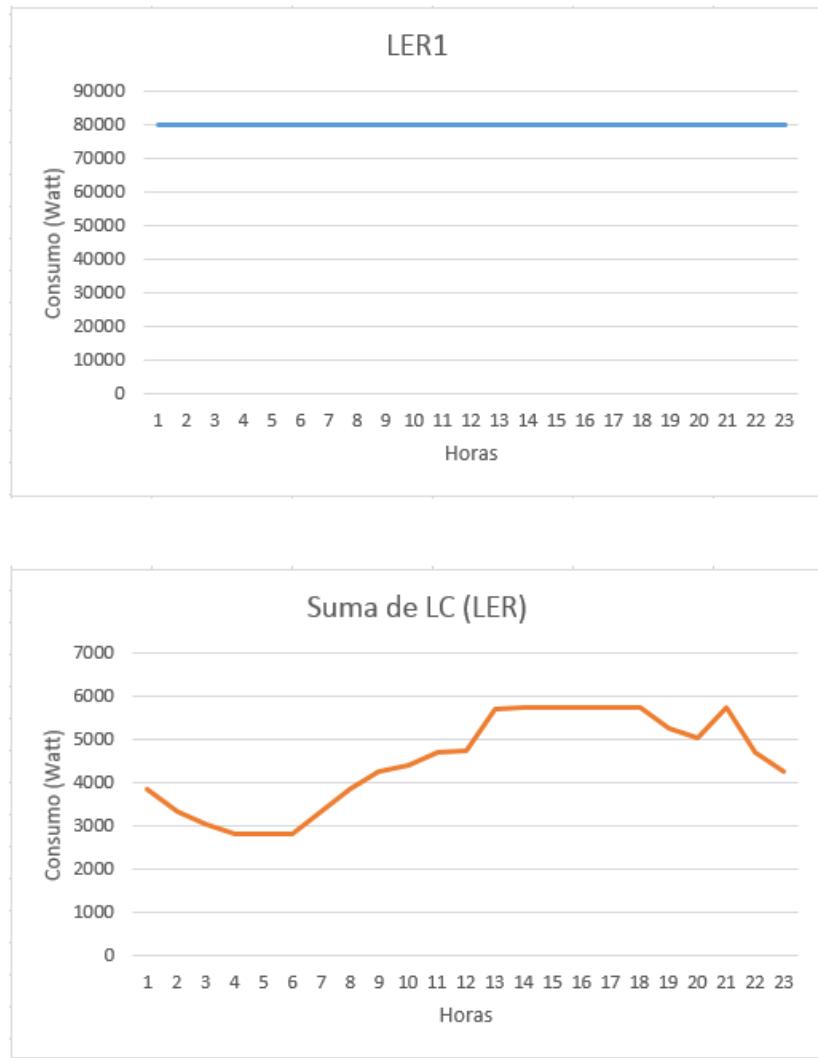


Figura 47: Ejemplo dispositivos LER (Consumo total)

El consumo permanece constante en el LER1. En la Suma de LC se produce un ligero ascenso, con un pico en las 21 horas.

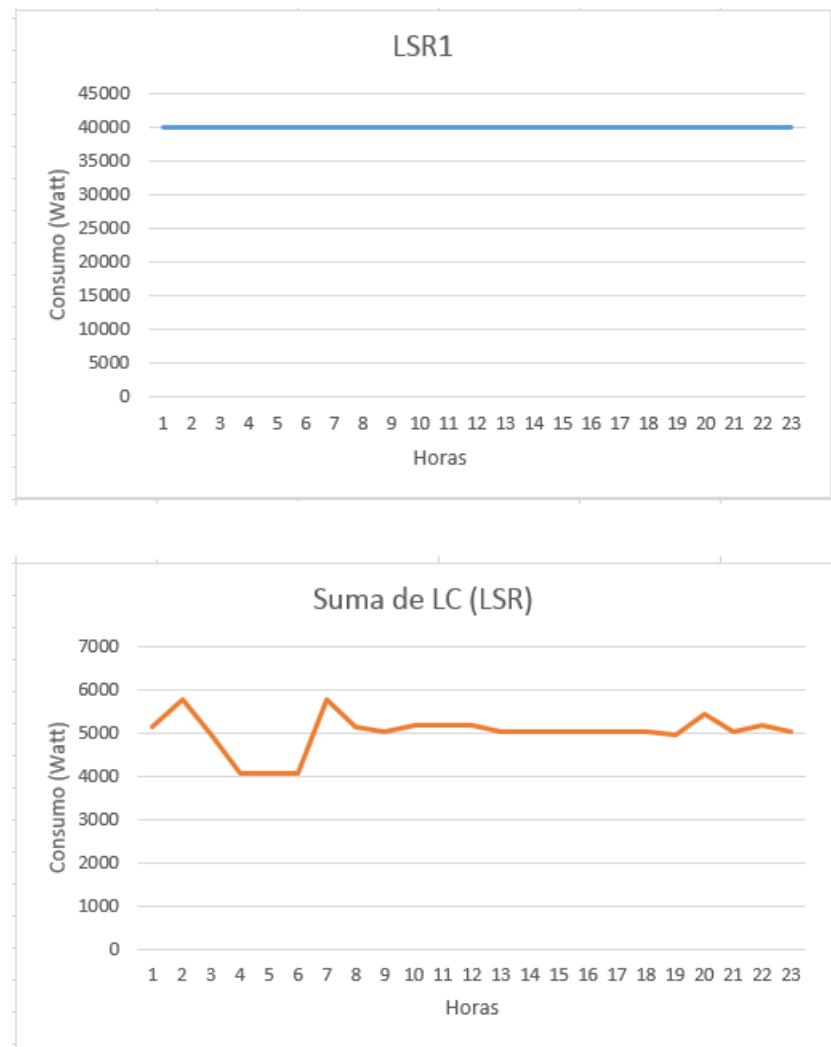


Figura 48: Ejemplo dispositivos LSR (Consumo total)

Valor constante en LSR1 y ligeras variaciones la Suma de LC, entre las 4 y las 7 se produce un descenso y en el resto de las horas el consumo se mantiene más o menos constante.

Parte PH:

Fórmula PH:

$$\text{Consumo} = \text{Consumo L LC} + \text{Consumo PIU} + (\text{Suma de R LC (7 PM)} - \text{Suma de R LC}) * 0.5 + (\text{Suma de L LC (7 PM)} - \text{Suma de L LC}) * 0.5$$

Mismo caso que la parte LO, se suman los consumos de los elementos “encendidos” con los de los elementos en “StandBy” siendo estos calculados mediante la resta con las 7PM, la hora cargada, aquí solo tenemos una operación para todos los dispositivos.

a) VALORCUBO

Formamos la fórmula extrayendo los valores del cubo necesarios mediante la función VALORCUBO.

La sintaxis es similar al caso LO, con valores de las 7PM siempre necesarios a extraer en cada hora, en este caso hay dos valores de este tipo por operación. La **figura 49** nos muestra una fórmula con valores extraídos con VALORCUBO en una celda concreta, igual que la parte LO.

Figura 49: Ejemplo de VALORCUBO (PH)

b) VBA:

El programa almacenará los valores y realizará las operaciones para cada lámina del cubo, similar a la parte LO, con la ventaja de que solo tenemos una operación que realizar, simplificando mucho las cosas.

```
'Activamos la lámina del cubo correspondiente a las 7PM.
ThisWorkbook.Worksheets("Cubo_PH").Activate
ThisWorkbook.Worksheets("Cubo_PH").PivotTables("Tabla dinámical").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName =
Call barra(1)
vsiete = vsiete_PH() 'Extraemos los valores de Suma de L LC y Suma de R LC que necesitaremos en posteriores cálculos.
```

Figura 50: Código calculos_PH 1

Extraemos los valores de las 7 PM activando la hoja correspondiente.

```

w = 0 'Inicializar decenas.
y = 0 'Inicializar unidades.
c = True 'Inicializar variable de cambio.

Call barra(2)

z = 0 'Inicializamos indice de vector resultado.
For k = 1 To 23

    Call Chequeo(w, y, c) 'Uso igual en el anterior programa.

    'Activamos cada lámina del cubo.
    ThisWorkbook.Worksheets("Cubo_PH").Activate
    ThisWorkbook.Worksheets("Cubo_PH").PivotTables("Tabla dinámica").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName =
    'Valor 267 de la hora 11 PM ignorado.

    Call Operaciones_PH(q, vsetie, z) 'Operamos.

    y = y + 1

Next k

Call insval_PH(q) 'Introducimos los valores resultado en tabla resultado.
Call barra(3)

```

Figura 51: Código calculos_PH 2

Detrás de CurrentPageName (identificador OLAP):

```
"[Tabla_PH].[Hora].&[1899-12-30T" + CStr(w) + CStr(y) + ":00:00]"
```

Figura 52: Identificador OLAP 2

Operamos en cada lámina del cubo actualizando los valores de hora w e y para desplazarnos por las láminas e insertamos los nombres de identificadores, los nombres de las columnas y los resultados.

c) DAX:

Para la parte DAX tenemos el mismo problema que en la parte LO, necesitamos crearnos una columna con los valores de las 7 PM para cada elemento, necesitamos los valores de “Suma de L LC” y “Suma de R LC”, para la columna de Consumos tenemos la mejora con respecto a la parte LO de que solo necesitamos una operación. Los dos ejemplos siguientes (**figura 53** y **figura 54**) nos muestran dos fragmentos de los ejemplos anteriores.

(Para consultar el código DAX completo ver anexo sección Suma de L LC 7PM y consumos Watt/hr_PH).

```


$$\begin{aligned}
& \text{IF } [\text{Conf\_Index}] = \text{BLANK}(); \\
& \quad \text{LOOKUPVALUE} \{ \\
& \quad \quad [\text{Suma de L LC}]; \\
& \quad \quad [\text{Conf\_Index}]; \\
& \quad \quad [\text{Hora}]; \text{VALUE} ( "30/12/1899 19:00:00" ) \\
& \quad \}; \\
& \text{IF } [\text{Conf\_Index}] = \text{VALUE("1")}; \\
& \quad \text{LOOKUPVALUE} \{ \\
& \quad \quad [\text{Suma de L LC}]; \\
& \quad \quad [\text{Conf\_Index}]; \text{VALUE("1")}; \\
& \quad \quad [\text{Hora}]; \text{VALUE} ( "30/12/1899 19:00:00" ) \\
& \quad \}; \\
& \text{IF } [\text{Conf\_Index}] = \text{VALUE("2")}; \\
& \quad \text{LOOKUPVALUE} \{ \\
& \quad \quad [\text{Suma de L LC}]; \\
& \quad \quad [\text{Conf\_Index}]; \text{VALUE("2")}; \\
& \quad \quad [\text{Hora}]; \text{VALUE} ( "30/12/1899 19:00:00" ) \\
& \quad \}; \\
& \text{IF } [\text{Conf\_Index}] = \text{VALUE("3")}; \\
& \quad \text{LOOKUPVALUE} \{ \\
& \quad \quad [\text{Suma de L LC}]; \\
& \quad \quad [\text{Conf\_Index}]; \text{VALUE("3")}; \\
& \quad \quad [\text{Hora}]; \text{VALUE} ( "30/12/1899 19:00:00" ) \\
& \quad \}; \\
& \text{IF } [\text{Conf\_Index}] = \text{VALUE("4")}; \\
& \quad \text{LOOKUPVALUE} \{ \\
& \quad \quad [\text{Suma de L LC}]; \\
& \quad \quad [\text{Conf\_Index}]; \text{VALUE("4")}; \\
& \quad \quad [\text{Hora}]; \text{VALUE} ( "30/12/1899 19:00:00" ) \\
& \quad \};
\end{aligned}$$


```

Figura 53: Fragmento del código DAX "7PM" Parte PH

$$= [\text{Consumo L LC}] + [\text{Consumo PIU}] + ([\text{Suma de R LC 7PM}] - [\text{Suma de R LC}]) * 0,5 + ([\text{Suma de L LC 7PM}] - [\text{Suma de L LC}]) * 0,5$$

Figura 54: Consumo PH DAX

Copiamos las columnas que nos interesan en una tabla aparte: Conf_Index, Hora y Consumos Watt/hr, las ordenamos: Hora de 0 a 23 y números de 1 a 267 seguidos de una sucesión de valores en blanco.

Creamos el programa que nos ordene la tabla:

```

Call barra(1)

ThisWorkbook.Worksheets("Tabla_DAX_PH").Activate 'Activamos hoja origen.

s = Recdisp_PH() 'Recogemos los índices que identifican los elementos.
x = Recinf_PH() 'Recogemos los consumos.

Call barra(2)

'A diferencia que en el caso LO, aquí insertamos los valores directamente.

ThisWorkbook.Worksheets("Cálculos_DAX_PH").Activate 'Activamos hoja resultado.

Call insinf_PH(s, x, q) 'Insertamos valores de identificación y de resultado (Consumos).

Call barra(3)

End Sub

```

Figura 55: Código calculos_DAX_PH

El mecanismo es similar al de la parte LO.

Un aspecto a comentar es el exceso de repetición de los valores “en blanco” (Type: PIU, Rack, repeaters), recordamos que el cubo nos “agrupaba” valores repetidos para una determinada hora, sin embargo, ahora estamos trabajando con columnas de PowerPivot, por lo que los valores que tenemos son individuales y repetidos.

(Para consultar el programa completo que ordena la tabla ver anexo sección calculos_DAX_PH).

La tabla resultado tendrá más filas que la tabla resultado de la parte LO, 268 en total (267 + valor en blanco), la tabla de resultados generada por la parte DAX tendrá más filas (346), las columnas serán las mismas, tantas como horas.

(Para consultar la tabla de resultados ver anexo sección Tabla de resultados PH).

d) Gráficas de los elementos tipo PH:

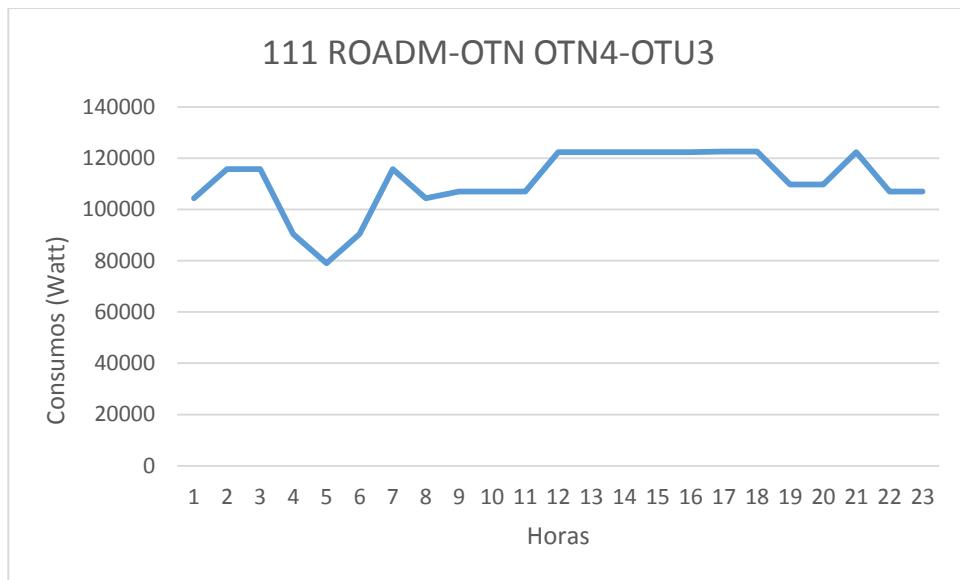


Figura 56: Gráfica del dispositivo 111

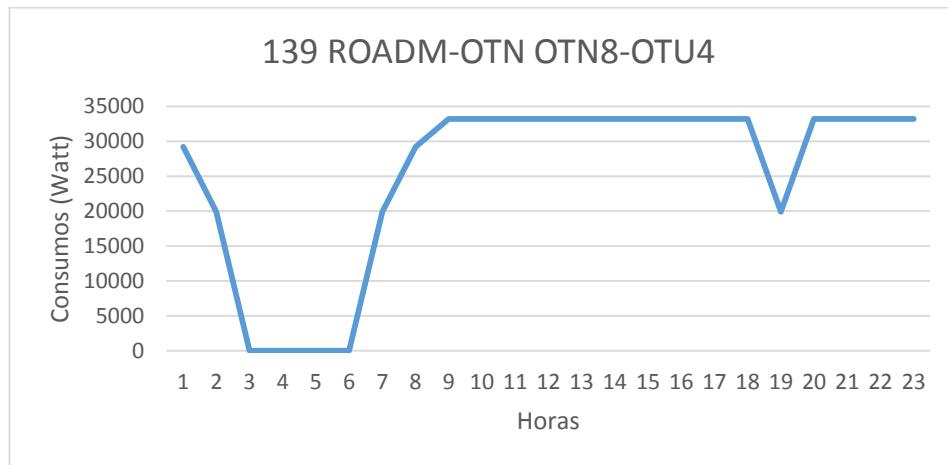


Figura 57: Gráfica del dispositivo 139

El dispositivo 111 tiene 1 GB de capacidad Add/Drop mientras que el 139 tiene 10 GB, ambos tienen una capacidad de 4 TB de anillo. Ambos dispositivos presentan una bajada en el consumo entre las 3 y las 6, el 111 pasa de casi 120.000 Watt a 80.000 Watt mientras que el 139 pasa de 30.000 Watt a 0 Watt. 111 tiene un pico de subida entre las 21 y las 22.

Diagrama de flujo y esquema:

Para comprender mejor cómo se han desarrollado los programas que manejan el cubo veremos un diagrama de flujo y un esquema (figura 58 y figura 59), como hicimos con la parte RecData. (Para ver los códigos completos consultar anexo sección calculos_LO y calculos_PH).

Diagrama de flujo:

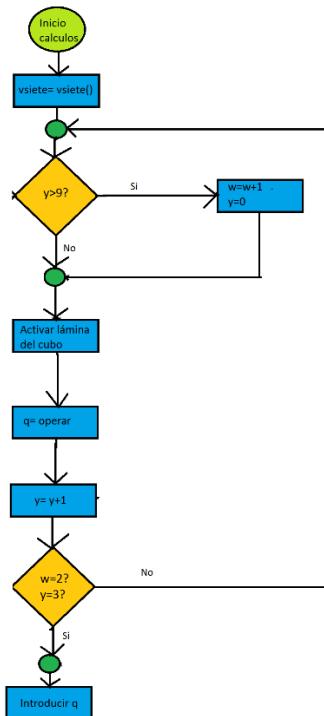


Figura 58: Diagrama de flujo calculos

Esquema Programas VBA:

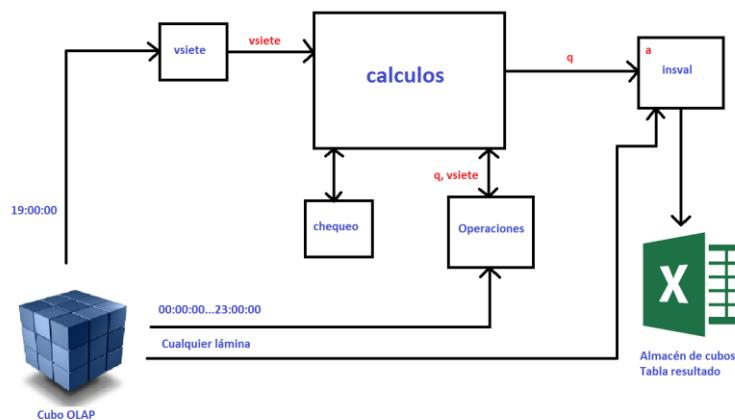


Figura 59: Esquema programa calculos

4.7 Análisis de resultados

Comparación de resultados:

a) VALORCUBO:

VALORCUBO es el más “sencillo” debido a que simplemente se trata de la repetición de una serie de fórmulas teniendo en cuenta algunos factores, como el dispositivo seleccionado para la parte LO (ESW, LC, LER o LSR), además de que es bastante fiable ya que es relativamente fácil identificar el motivo de algún error que pueda ocurrir en la solución. Su problema principal es que las expresiones generadas son muy largas y son muchas, haciendo el trabajo de introduciéndolas y modificándolas en caso de cambio un tanto laborioso, por lo que es un método poco flexible y dinámico. En la elaboración del proyecto se encontraron dificultades para definir correctamente la sintaxis de la función.

b) VBA:

VBA es el método más rápido, realiza las operaciones automáticamente a una velocidad aceptable, sin embargo, presenta más problemas que VALORCUBO, ya que requiere el aprendizaje de las técnicas de programación adecuadas, además de que los errores producidos son mucho más difíciles de identificar debido a la longitud del código, estos pueden ser debido a índices mal inicializados, a valores mal introducidos en un vector, a elementos no identificados... Sin embargo ante cualquier modificación de algún dato u operación se puede cambiar fácilmente su equivalente en el código, por lo que ofrece una gran flexibilidad.

c) DAX:

DAX es, en mi opinión personal, la peor opción de todas. Para lograr las columnas de los valores 7PM hay que introducir una gran cantidad de datos, que muchas veces PowerPivot tarda en procesar, y deben estar escritos adecuadamente, un solo error hace que los resultados sean incorrectos, además de que una vez establecidos los cálculos hay que organizarlos en varios pasos, utilizando programación adicional, con lo que el riesgo de error es elevado y las tareas a realizar se complican bastante.

d) Mejor opción:

En mi opinión personal, la opción más favorable es la opción VBA debido a que permite generar la tabla de resultado de la forma más rápida y es la más flexible a posibles modificaciones en las operaciones.

Resultados con hojas iniciales:

Todos los pasos anteriores nos han servido para ver distintas maneras de representar resultados trabajando con un cubo OLAP, ahora vamos a tratar de generar resultados usando el primer libro de todos, “resultados_detalle”, mediante un programa VBA, y compararemos los resultados con el caso anterior.

a) Programa VBA con hojas iniciales:

Ejemplo PH:

```
Workbooks.Open Filename:="C:\Users\Jesús\Desktop\resultados_detalle.xlsx"
Workbooks("resultados_detalle.xlsx").Activate 'Abrimos y activamos el libro donde se encuentran las hojas con información.
Call barra(0)
Call calculos_secundarios_PH(l1c, r1c, c1c, cpiu, p) 'Cálculos necesarios para otros cálculos.
Call calculos_primeros_PH(q, l1c, r1c, c1c, cpiu, p) 'Cálculo principal, el del consumo.
Call barra(1)
Call extraer_nombres_PH(t, s, p) 'Nombres de "enmarcado".
Call barra(2)
Call introducir_datos_PH(t, s, q) 'Introducir datos en hoja destino.
Call barra(3)
Workbooks("resultados_detalle.xlsx").Close 'Cerrar el libro donde están las tablas origen.
```

Figura 60: Código resultados_PH

En este programa realizamos los cálculos secundarios que antes se hicieron con DAX en PowerPivot, después operamos para cada hoja del libro, extraemos los nombres de las columnas y los dispositivos e introducimos los datos en una tabla de resultados final.

(Para consultar el código completo ver anexo sección resultados_calculos).

b) Diagrama de flujo y esquema:

El diagrama de flujo y el esquema (**figura 61** y **figura 62**) nos permitirán entender mejor el desarrollo del programa.

Diagrama de flujo:

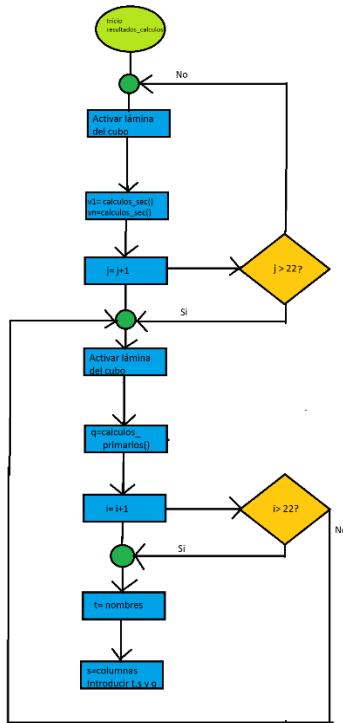


Figura 61: Diagrama de flujo resultados_calculos

Esquema:

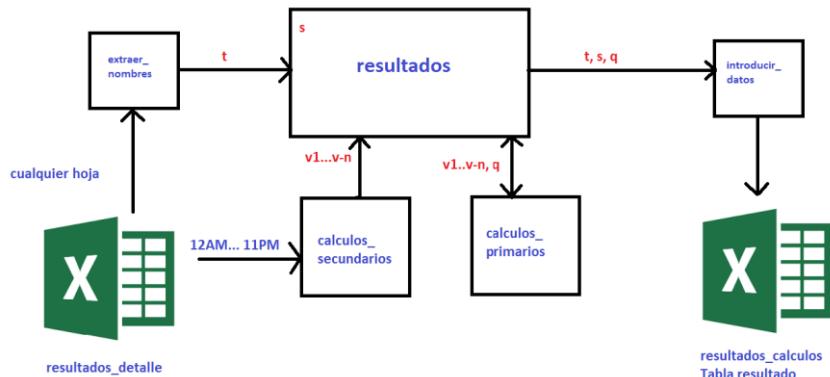


Figura 62: Esquema del programa resultados

Comparación resultados con cubo/sin cubo:

Como podemos observar, simplemente con un programa VBA hemos conseguido casi lo mismo que con todo el transcurso del proyecto; Sin embargo, esto es debido a que el escenario generado y desarrollado anteriormente es prácticamente una copia de lo representado en resultados_detalle, uno representado en varias hojas de un libro y el otro representado en varias láminas de un cubo.

Utilidad del cubo:

Sin embargo hay bastantes factores que justifican la utilidad del cubo:

Atendiendo a la parte PH, ¿Qué pasaría si queremos sacar los consumos referidos a un solo dispositivo de la parte Technology? Si observamos las hojas, vemos que hay muchos valores que se repiten, están desordenados y son muchos por lo que intentar repetir los cálculos usando un solo programa VBA nos complicaría demasiado la programación, con el cubo no habría apenas cambio en la programación, tan solo habría que modificar las filas y el índice de las filas en el programa VBA.

¿Y si queremos calcular el consumo de todos los tipos de una tecnología concreta para todas las horas? Con el cubo esto lo podemos hacer de forma rápida mientras que directamente con resultados_detalle tardaríamos más.

En la parte LO, ¿Si quisiéramos saber que dispositivos tienen un valor de 10 “Number of LC” a las 10 de la tarde?

Podríamos manualmente verlo en resultados_detalle pero el cubo nos lo ofrecería de forma inmediata. El potencial del cubo no solo radica en la extracción de los datos sino también en la visualización de los mismos.

La **figura 63** nos da una visión general del desarrollo del proyecto, puntuizando que con el cubo podemos tener más resultados que con las hojas iniciales.

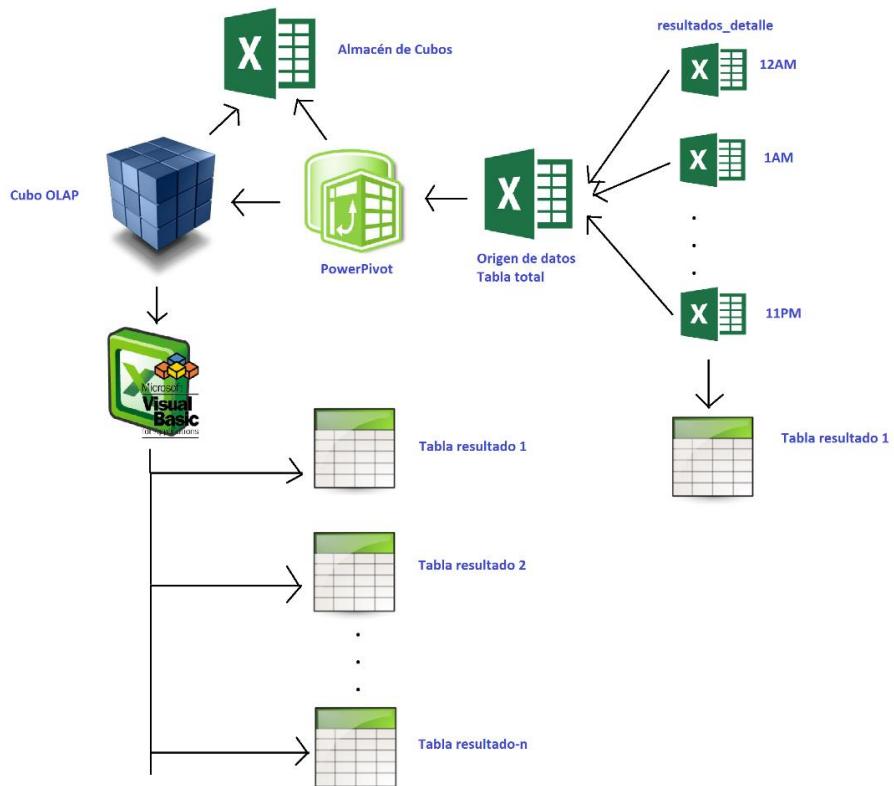


Figura 63: Esquema del desarrollo del proyecto

En él podemos observar que aunque el camino hasta los resultados finales ha sido algo largo, el poder tener multitud de resultados diferentes compensa ese hecho.

5. Otros Ejemplos

Este apartado sirve para presentar otros casos distintos al anterior en los que utilizaremos distintos cubos para generar distintos resultados.

Cubo con filas con el campo technology:

En la **figura 64** vemos un cubo basado en la tabla PH en el que las filas son technology en vez de Conf_Index.

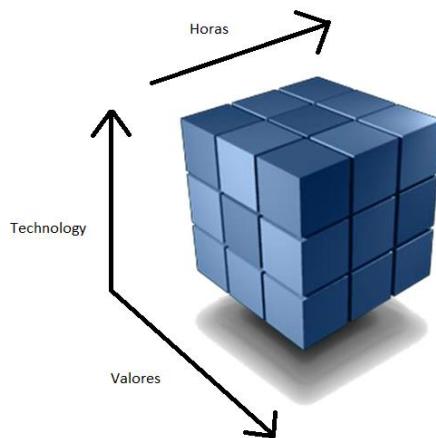


Figura 64: Cubo con filas technology Eje x: Technology Eje y: Valores Eje z: Horas

a) Programa VBA technology:

```
'Activamos la lámina del cubo correspondiente a las 7PM.
ThisWorkbook.Worksheets("Cubo_PH_technology").Activate
ThisWorkbook.Worksheets("Cubo_PH_technology").PivotTables("Tabla dinámica").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName = "[Tabla_PH].[Hora].&[1899-12-30T19:00:00]"
Call barra(1)
vsiete = vsiete_PH() 'Extraemos los valores de Suma de L LC y Suma de R LC que necesitaremos en posteriores cálculos.
w = 0 'Inicializar decenas.
y = 0 'Inicializar unidades.
c = True 'Inicializar variable de cambio.
Call barra(2)
z = 0 'Inicializamos indice de vector resultado.
For k = 1 To 23
    Call Chequeo(w, y, c) 'Uso igual en el anterior programa.

    'Activamos cada lámina del cubo.
    ThisWorkbook.Worksheets("Cubo_PH_technology").Activate
    ThisWorkbook.Worksheets("Cubo_PH_technology").PivotTables("Tabla dinámica").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName =
    'Valor 267 de la hora 11 PM ignorado.
    Call Operaciones_PH(q, vsiete, z) 'Operamos.
    y = y + 1
Next k
Call insval_PH(q) 'Introducimos los valores resultado en tabla resultado.
```

Figura 65: Código calculos_PH_technology

"[Tabla_PH].[Hora].&[1899-12-30T" + CStr(w) + CStr(y) + ":00:00]"

Figura 66: Identificador OLAP 3

El programa es exactamente igual al programa “cálculos”, con la diferencia de un cambio de los índices de los ciclos que se encargaban de extraer la información de las filas debido a que ahora tenemos en dichas filas el campo “Technology”. La información provista por “Technology” es más “concreta” que la provista por “Conf_Index”.

(Para ver la tabla resultado consultar anexo sección Tabla de resultados technology).

a) Gráficas:

Extraeremos algunas gráficas que nos den información de los dispositivos, en este caso los que tengan valor:

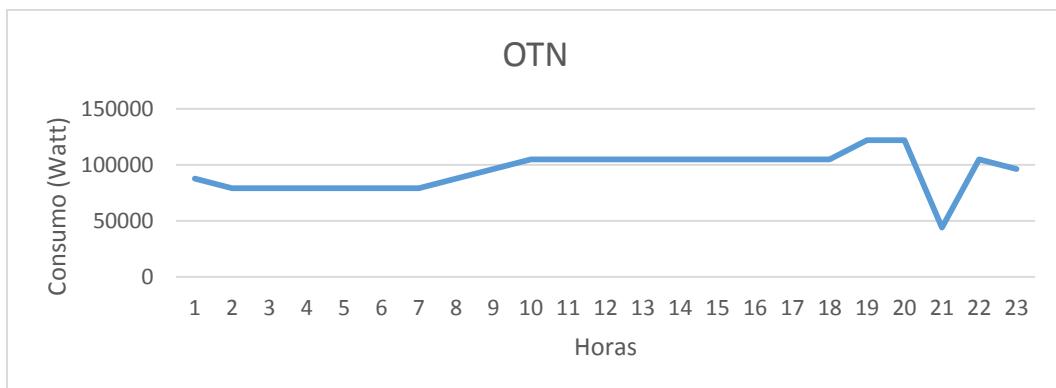


Figura 67: Gráfica OTN

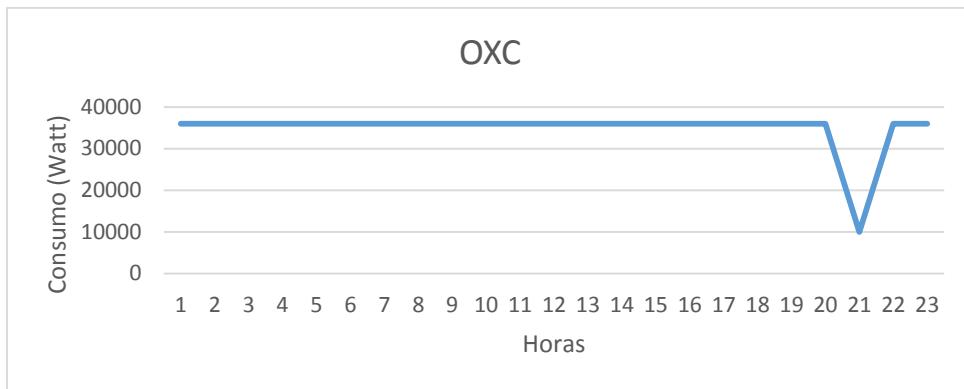


Figura 68: Gráfica OXC

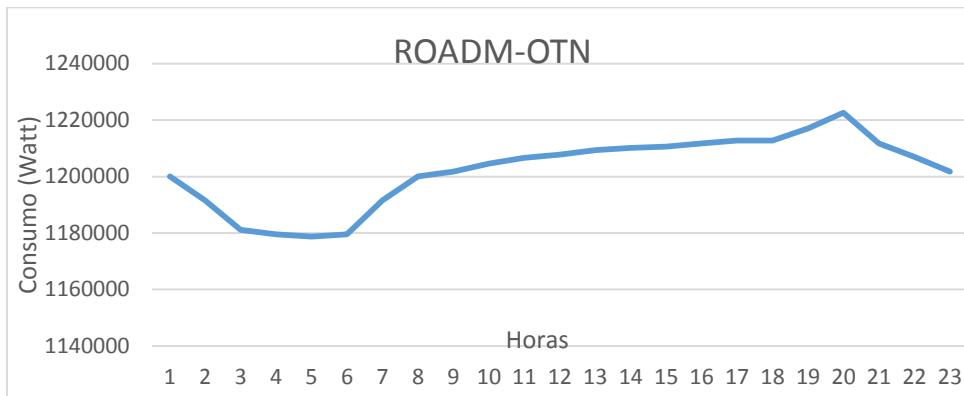


Figura 69: Gráfica ROADM-OTN

Todas estas gráficas dan información de los dispositivos totales de la red.

Cubo con filas type y filtro technology:

Creamos un cubo (**figura 70**) con el filtro technology y filas type:

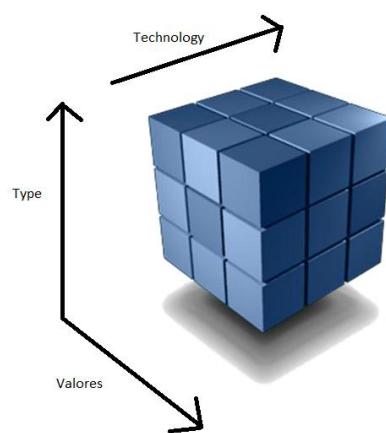


Figura 70: Cubo Type-technology Eje x: Type Eje y: Valores Eje z: Technology

a) Programa type-technology:

Realizaremos un programa que active la tecnología “OTN” y calcule los consumos para cada tipo de dicha tecnología en el total de las horas:

```

Sub calculos_hibridos()
    ThisWorkbook.Worksheets("Cubo_PH_hibrido").Activate
    ThisWorkbook.Worksheets("Cubo_PH_hibrido").PivotTables("Tabla_dinámica1").PivotFields("[Tabla_PH].[Technology].[Technology]").CurrentPageName = "[Tabla_PH].[Technology]&[OTN]"
    'Activar technology.

    Dim z As Integer 'Índice
    Dim cllc, crlc, t, q As Variant 'Datos

    Call vsietec(cllc, crlc, z) 'Datos a las 7 PM.

    ThisWorkbook.Worksheets("Cubo_PH_hibrido").Activate
    ThisWorkbook.Worksheets("Cubo_PH_hibrido").PivotTables("Tabla_dinámica1").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName = "[Tabla_PH].[Hora].[All]"
    'Technology a todas las horas

    Call extraer_datos(t, z) 'Datos nombres Type.

    Call operar(q, cllc, crlc, z) 'Operaciones.

    Call introducir_info(q, t, z) 'Introducir en tabla.

End Sub

```

Figura 71: Código *calculos_hibridos* (Type-Technology)

Esto es aplicable a cada “Technology” que queramos y para cada hora, sustituyendo en los correspondientes identificadores OLAP y cambiando los índices de los ciclos según corresponda.

b) Tabla

En la siguiente tabla (**tabla 7**) vemos los consumos de cada tipo de OTN en el total de las horas:

Tabla 7: Tabla de resultados OTN

	A	B
1	OTN	Consumo Total
2	OTN1	2176593
3	OTN2	0
4	OTN4	0
5	OTN8	0
6	Total general	2176593

Cubos con filas horas y filtro dispositivos:

Para este apartado modificaremos los ejes de los cubos LO y PH anteriormente creados, cambiando el eje x (filas), por el eje z (filtros), eso hará que la información representada sea de forma diferente y, por tanto, también el análisis y el manejo de dicho cubo.

a) Parte LO:

Cambiamos las horas de la sección filtros a la sección filas y el Tipo de disp. de la sección filas a la sección filtros, esto producirá que para cada filtro se nos muestre la información del resto de valores de la tabla referidos a cada hora del día; Esto es útil para ver como un solo dispositivo evoluciona en el transcurso de las horas, al contrario que en el anterior ejemplo, en el que veíamos como evolucionaban el conjunto de los dispositivos en una sola hora.

En la **figura 72** vemos el cubo de forma tridimensional.

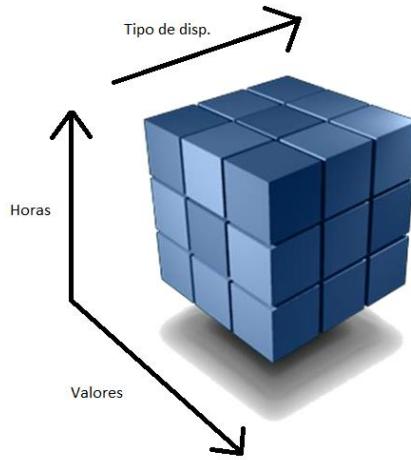


Figura 72: Cubo LO hora Eje x: Horas Eje y: Valores Eje z: Tipo de disp.

En la **figura 73** vemos un fragmento de la tabla dinámica, en la lámina ESW1.

A	B	C	D	E	F	G	H
1	Tipo de disp	ESW1					
2							
3	Etiquetas de fila	Suma de # Ports per LC	Suma de Activo	Suma de Adapt ETH	Suma de StandBy	Suma de Consumo Unitario PIU Watt	Suma de Consumo PIU StandBy
4	0:00:00	10	0	0	0	500	300
5	1:00:00	10	0	0	0	500	300
6	2:00:00	10	0	0	0	500	300
7	3:00:00	10	0	0	0	500	300
8	4:00:00	10	0	0	0	500	300
9	5:00:00	10	0	0	0	500	300
10	6:00:00	10	0	0	0	500	300
11	7:00:00	10	0	0	0	500	300
12	8:00:00	10	0	0	0	500	300
13	9:00:00	10	0	0	0	500	300
14	10:00:00	10	0	0	0	500	300
15	11:00:00	10	0	0	0	500	300
16	12:00:00	10	0	0	0	500	300
17	13:00:00	10	0	0	0	500	300
18	14:00:00	10	0	0	0	500	300
19	15:00:00	10	0	0	0	500	300
20	16:00:00	10	0	0	0	500	300
21	17:00:00	10	0	0	0	500	300
22	18:00:00	10	0	0	0	500	300
23	19:00:00	10	0	0	0	500	300
24	20:00:00	10	0	0	0	500	300
25	21:00:00	10	0	0	0	500	300
26	22:00:00	10	0	0	0	500	300
27	Total general	230	0	0	0	11500	6900

Figura 73: Ejemplo de tabla dinámica LO "hora"

Se mostrarán los mismos resultados que en el ejemplo anterior, es decir, el consumo de un dispositivo a lo largo de un día, con la diferencia de que esta vez para una lámina del cubo tendremos la información completa de lo que a un dispositivo se refiere. Se usará un programa VBA para lograr este resultado.

a) Programa VBA:

```

Workbooks.Open ("C:\Users\Jesús\Desktop\Origen de datos.xlsx")
Workbooks("Origen de datos.xlsx").Worksheets("Tabla_LO_filtrada").Activate 'Abrimos el libro donde se encuentran nuestras tablas originales.
Call barra(1)
q = recdisp_LO2() 'Recogemos los dispositivos de nuestra tabla.
Call elmrep_LO(q, s, x) 'Eliminamos valores repetidos para que solo se maneje una lámina del cubo por tipo.
Workbooks("Origen de datos.xlsx").Close 'Cerramos el libro origen de datos.

```

Figura 74: Código calculos_hora_LO 1

En esta parte recogemos los nombres que actúan de filtros en nuestro cubo para poder ir desplazándonos por las diferentes láminas, para ello extraemos los nombres de la tabla de datos la cual creamos nuestro cubo, lo hacemos porque es la mejor fuente para extraer dicha información.

Después tenemos que asegurarnos de que no hay valores que se repitan en nuestro vector de dispositivos, cosa que puede ser posible ya que fue nuestro cubo el que se encargó de eliminar las repeticiones. La variable x se encargará de darnos la información del tamaño del vector de elementos sin repetir, que no es otra que el número de elementos del filtro del cubo. Cerramos el libro de origen de datos debido a que ya no tiene utilidad.

```

Call barra(2)
z = 0
For i = 0 To x 'Número de Filtrar.
    'Activando cada lámina del cubo según los valores contenidos en el vector adecuado. w y z forman parte del identificador OLAP.
    ThisWorkbook.Worksheets("Cubo_10_hora").Activate
    ThisWorkbook.Worksheets("Cubo_10_hora").PivotTables("Tabla_dinámica2").PivotFields("[Tabla_LO_filtrada].[Tipo de disp].[Tipo de disp]").CurrentPageName =
        For j = 4 To 26 'Para cada hora habrá un valor, el ciclo esta vez estará fuera de la función de operaciones, cada hora será una fila de una lámina.
            Call operar_LO(i, w, z, i, j) 'Operamos.
        Next j
    Next i
    ThisWorkbook.Worksheets("Cálculos_LO hora").Activate 'Activamos la hoja resultado.
    Call insdat_LO(s, w, x) 'Introducimos los datos resultado, filas y columnas, siendo las columnas los filtros del cubo.
    Call barra(3)

```

Figura 75: Código calculos_hora_LO 2

Detrás de CurrentPageName:

```
"[Tabla_LO_filtrada].[Tipo de disp].&[" + s(i) + "]"
```

Figura 76: Identificador OLAP 4

Recorremos cada lámina del cubo con el identificador OLAP mediante el vector de dispositivos (s(i)) y, para cada lámina, realizamos las operaciones necesarias, que se repetirán en cada hora (fila) con sus respectivos datos por hora. Esto presenta un cambio con respecto al anterior ejemplo, en el que en cada hora teníamos diferentes dispositivos con diferentes operaciones por fila.

b) Tabla y gráficas

Acabadas las operaciones, pasamos a insertar los datos en la tabla correspondiente.

En la **tabla 8** vemos un fragmento de los resultados obtenidos.

Tabla 8: Ejemplo de tabla de resultados LO "hora"

	A	B	C	D	E	F	G	H	I	LC8	LC9	J	LC10	LC11	K	LC12	LC13	M	LC14	LC15	N	LC16	P	LC17	Q	LC18
1	Hora	ESW1	ESW1a	ESW2	ESW3	ESW4	LC7																			
2	0:00:00	495700	0	0	0	0	0	17487	0	848,5	0	18024	0	1621,4	0	1082,6	0	0	4							
3	1:00:00	495700	0	0	0	0	0	17388	0	933	0	17345	0	1890,8	3	0	4									
4	2:00:00	495700	0	0	0	0	0	17289	0	1017,5	0	16666	0	1621,4	3	0	4									
5	3:00:00	495700	0	0	0	0	0	17289	0	1017,5	0	16666	0	1352	3	0	4									
6	4:00:00	495700	0	0	0	0	0	17289	0	1017,5	0	16666	0	1352	3	0	4									
7	5:00:00	495700	0	0	0	0	0	17289	0	1017,5	0	16666	0	1352	3	0	4									
8	6:00:00	495700	0	0	0	0	0	17388	0	933	0	17345	0	1890,8	3	0	4									
9	7:00:00	495700	0	0	0	0	0	17487	0	848,5	0	18024	0	1621,4	0	1082,6	0	0	4							
10	8:00:00	495700	0	0	0	0	0	17223	0	933	0	18703	0	1890,8	1622,4	0	4									
11	9:00:00	495700	0	0	0	0	0	17223	0	848,5	0	1904,25	0	1890,8	1622,4	0	4									
12	10:00:00	495700	0	0	0	0	0	17025	0	848,5	0	1972,15	0	1890,8	1622,4	0	4									
13	11:00:00	495700	0	0	0	0	0	17025	0	764	0	20061	0	1890,8	1622,4	0	4									
14	12:00:00	495700	0	0	0	0	0	16959	0	848,5	0	20061	0	1890,8	1622,4	0	4									
15	13:00:00	496100	0	0	0	0	0	17157	0	679,5	0	20061	0	1890,8	1082,6	0	470									
16	14:00:00	496500	0	0	0	0	0	17256	0	595	0	20061	0	1890,8	1082,6	0	470									
17	15:00:00	496500	0	0	0	0	0	17190	0	679,5	0	20061	0	1352	1622,4	0	936									
18	16:00:00	496500	0	0	0	0	0	17190	0	679,5	0	20061	0	1352	1622,4	0	936									
19	17:00:00	496500	0	0	0	0	0	17190	0	679,5	0	20061	0	1352	1622,4	0	936									
20	18:00:00	496500	0	0	0	0	0	17289	0	595	0	20740	0	1352	1622,4	0	936									
21	19:00:00	496500	0	0	0	0	0	17289	0	595	0	20740	0	1352	1622,4	0	936									
22	21:00:00	496500	0	0	0	0	0	17190	0	679,5	0	20061	0	1352	1622,4	0	936									
23	22:00:00	495700	0	0	0	0	0	16926	0	848,5	0	20061	0	1890,8	1622,4	0	4									
24	23:00:00	495700	0	0	0	0	0	17223	0	933	0	18703	0	1890,8	1622,4	0	4									

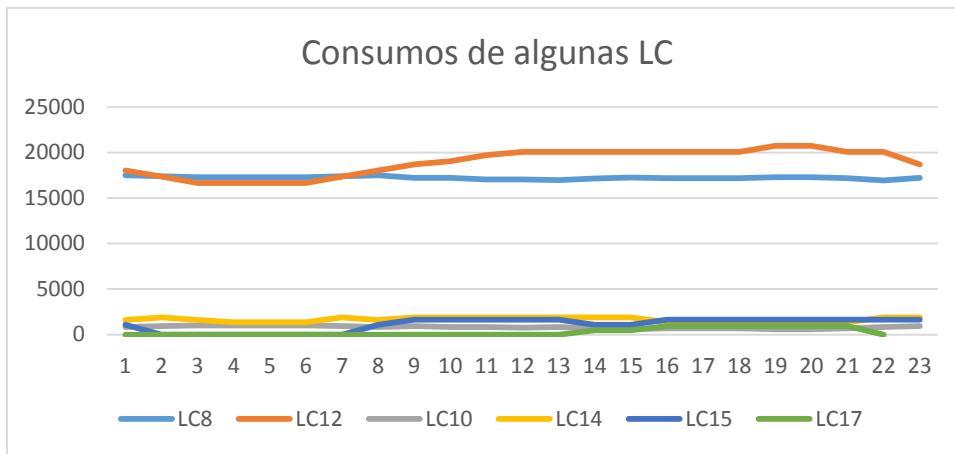


Figura 77: Consumos de algunas LC

Observamos que es la misma tabla que la obtenida con el cubo LO, pero transpuesta: Filas en lugar de columnas y columnas en lugar de filas.

b) Parte PH:

Cambiamos las filas “Conf_Index” por el filtro “Hora”, para cada valor de Conf_Index tendremos sus valores en cada hora, similar al caso LO. Observamos el resultado en las siguientes figuras (figura 78 y figura 79).

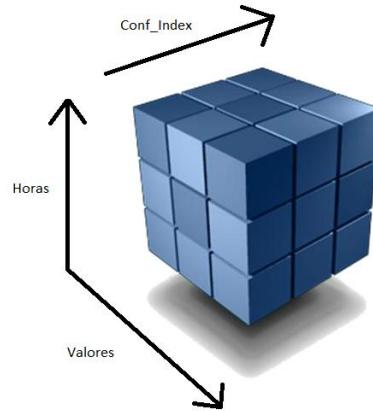


Figura 78: Cubo PH horas Eje x: Horas Eje y: Valores Eje z: Conf_Index

A	B	C	D	E	F	G	H
1	Conf_Index	1					
2	Etiquetas de fila	Suma de Nº de L portas	Suma de BW L LC MBps	Suma de Consumo por tarjeta	Suma de Number of L LC	Suma de Number of R LC	Suma de Number of PIUs
3							Suma de Number of R LC
4	0:00:00	1	100	0	0	0	0
5	1:00:00	1	100	0	0	0	0
6	2:00:00	1	100	0	0	0	0
7	3:00:00	1	100	0	0	0	0
8	4:00:00	1	100	0	0	0	0
9	5:00:00	1	100	0	0	0	0
10	6:00:00	1	100	0	0	0	0
11	7:00:00	1	100	0	0	0	0
12	8:00:00	1	100	0	0	0	0
13	9:00:00	1	100	0	0	0	0
14	10:00:00	1	100	0	0	0	0
15	11:00:00	1	100	0	0	0	0
16	12:00:00	1	100	0	0	0	0
17	13:00:00	1	100	0	0	0	0
18	14:00:00	1	100	0	0	0	0
19	15:00:00	1	100	0	0	0	0
20	16:00:00	1	100	0	0	0	0
21	17:00:00	1	100	0	0	0	0
22	18:00:00	1	100	0	0	0	0
23	19:00:00	1	100	0	0	0	0
24	21:00:00	1	100	0	0	0	0
25	22:00:00	1	100	0	0	0	0
26	23:00:00	1	100	0	0	0	0
27	Total general	23	2300	0	0	0	0

Figura 79: Ejemplo de tabla dinámica PH "hora"

a) Programa VBA:

```
Workbooks.Open ("C:\Users\Jesús\Desktop\Origen de datos.xlsx")
Workbooks("Origen de Datos.xlsx").Worksheets("Tabla_PH").Activate 'Activar libro donde está la hoja origen de datos.

Call barra(1)

q = recdisp_PH2() 'Recogemos los dispositivos.

Call elmrep_PH(q, s, x) 'Eliminamos los repetidos.

c = convnum(s, x) 'los identificadores son números, para un correcto procesado posterior, los convertimos en números.

Call estval(uno, dos, tres, cuatro) 'Rellenamos los vectores que servirán para crear los identificadores válidos OLAP.

k = 0 'Inicializamos los índices de los vectores identificadores antes de entrar en el ciclo.
l = 0
b = 0

For i = 0 To x 'Valores numéricos

Call cambform(p, c, k, l, b, i, uno, dos, tres, cuatro) 'Conversión a formato OLAP de cada valor identificador para cada valor numérico.

Next i

Workbooks("Origen de datos.xlsx").Close 'Cerramos el libro origen de datos.
```

Figura 80: Código calculos hora PH 1

En esta parte se procede a extraer los datos de “Conf_index” del libro origen, se eliminan las repeticiones, que en este caso hay bastantes, todas del valor “en blanco”.

Sin embargo esta vez necesitamos un paso adicional debido a que parte de la identificación tipo OLAP no coincide con el nombre de los valores extraídos, tal como pasaba con la parte LO, para ello necesitamos realizar unos pasos de conversión que puedan servir para crear un identificador.

Los elementos de “Conf_index” son números, y cada número tiene asignada una identificación específica: 1=1., 2=2., 10=1.E1, 11=1.1 E1, 111=1.01E2...

```
Call barra(2)

z = 0 'Inicializamos el índice del vector de resultados.
For i = 0 To x

'Vamos activando cada lámina del cubo.
ThisWorkbook.Worksheets("Cubo_PH hora").Activate
ThisWorkbook.Worksheets("Cubo_PH hora").PivotTables("Tabla dinámica4").PivotFields("[Tabla_PH].[Conf_Index].[Conf_Index]").CurrentPageName =
Call operar_PH(v, z) 'Operamos, esta vez el ciclo le introducimos en la función.

Next i

ThisWorkbook.Worksheets("Cálculos_PH hora").Activate 'Activamos la hoja resultado.
Call insdat_PH(s, v, x) 'Insertamos los valores filas, columnas y resultados.

Call barra(3)
```

Figura 81: Código calculos hora PH 2

Detrás de CurrentPageName:

```
"[Tabla_PH].[Conf_Index].&" + p(i)
```

Figura 82: Identificador OLAP 5

Una vez tenemos los identificadores correctos pasamos a activar cada lámina del cubo, que en total serán 268 (267 + en blanco) y para cada lámina obtendremos una misma operación para cada una de las 23 filas (horas), esta vez todas las operaciones serán las mismas para cada "Conf_index". Al igual que en el caso LO, nuestra tabla estará transpuesta.

b) Tabla y gráficas

En la **tabla 9** vemos un fragmento de la tabla de resultados que se nos genera, al igual que la parte LO será la misma que lo obtenida con el cubo PH pero transpuesta.

Tabla 9: Ejemplo de tabla de resultados PH "hora"

	ED	EE	EF	EG	EH	EL	EJ	EK	EL	EM	EN	EO	EP	EQ	ER	ES	ET
1	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148
2	0	132255,5	0	7997	0	0	30856,8483	29209,9442	0	0	2	0	0	0	0	0	0
3	0	125662	0	7997	0	0	26858,8483	19927,7665	0	0	2	0	0	0	0	0	0
4	0	132856	5997	15994	0	1855,63553	13293,6777	7,5	0	0	2	0	0	0	0	0	0
5	0	126065	5997	23991	927,317767	1855,63553	9581,4066	7,5	0	0	2	0	0	0	0	0	0
6	0	125267	5997	23991	927,317767	1855,63553	9581,4066	7,5	0	0	2	0	0	0	0	0	0
7	0	126065	5997	23991	927,317767	1855,63553	9581,4066	7,5	0	0	2	0	0	0	0	0	0
8	0	125662	0	7997	0	0	26858,8483	19927,7665	0	0	2	0	0	0	0	0	0
9	0	132255,5	0	7997	0	0	30856,8483	29209,9442	0	0	2	0	0	0	0	0	0
10	0	141642	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
11	0	144435	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
12	0	146430	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
13	0	147627	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
14	0	149223	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
15	0	150021	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
16	0	150420	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
17	0	151617	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
18	0	152415	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
19	0	152415	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
20	0	166400	0	0	0	0	46276,6264	19927,7665	0	0	0	8617,73876	0	0	0	0	0
21	0	166400	0	0	0	0	38566,3553	33207,9442	0	0	0	8617,73876	0	0	0	0	0
22	0	151617	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
23	0	146829	0	0	0	0	38566,3553	33207,9442	0	0	2	0	0	0	0	0	0
24	0	141642	0	0	0	0	38566,3553	33207,9442	0	0	0	2	0	0	0	0	0

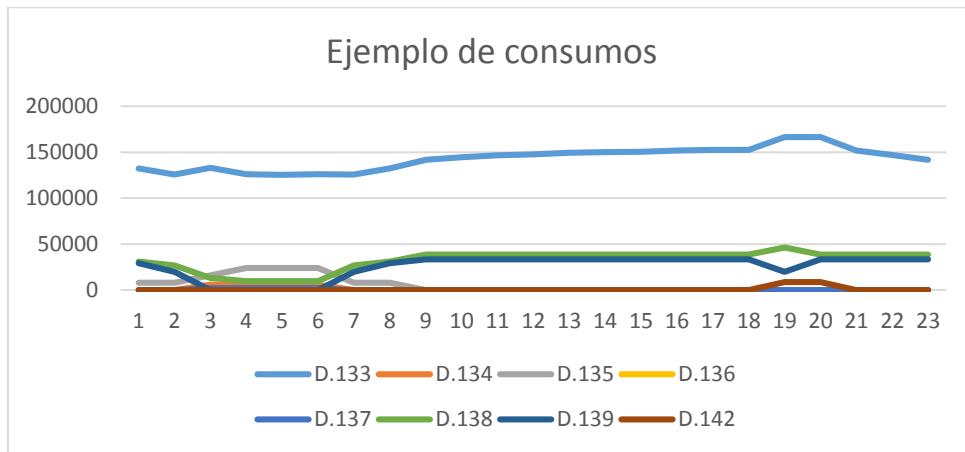


Figura 83: Consumos de algunos dispositivos

c) Diagrama de flujo y esquemas

Como en apartados anteriores crearemos un diagrama de flujo y un esquema (figura 84 y figura 85) referidos a la parte VBA para entender mejor su funcionamiento.

Diagrama de flujo:

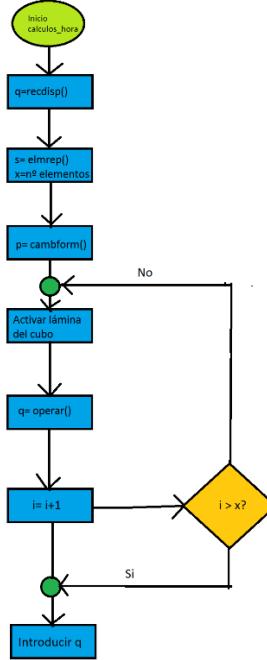


Figura 84: Diagrama de flujo calculos_hora

Esquema de los programas:

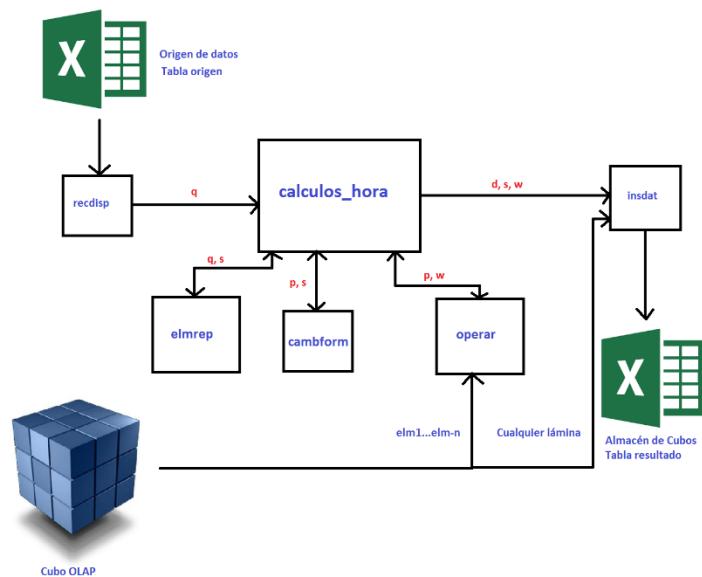


Figura 85: Esquema del programa calculos_hora

Analizando los casos anteriores, parece que un recorrido del cubo mediante los valores del filtro como texto no parece la mejor opción ya que nos complica el programa más que un filtro por horas.

d) Cubos con otras fuentes de datos:

Sin embargo, hay una opción que nos permite ahorrarnos el identificar los elementos repetidos y eliminarlos, que es extraer esa información de otro de los cubos creados anteriormente en vez de las tablas, las filas del cubo ya eran los identificadores ordenados correctamente y sin repeticiones. Éste es otro ejemplo de la utilidad de los cubos: Servir como fuente de datos a otros cubos relacionados con él. La **figura 86** nos muestra mejor ese concepto.

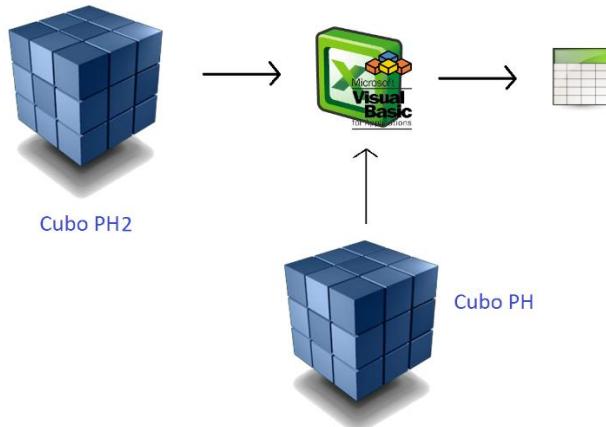


Figura 86: Fuente de datos cubo

Conclusiones y líneas futuras

Las técnicas OLAP poseen un gran potencial de análisis y procesado de datos, aunque también requiere que el usuario que las maneje tenga un cierto nivel de conocimiento del software “Microsoft Excel” y que, para un uso más completo, se tengan también unos conocimientos de Visual Basic, esto hace que se requieren usuarios algo “cualificados” para aprovechar dichas técnicas.

La utilidad de un almacén de datos tipo OLAP depende en gran medida del uso que se le vaya a dar a una cantidad de datos concreta y del tamaño y complejidad de éstos. Casos pequeños y sencillos quizás no necesiten de dichas técnicas pero para casos grandes y complejos su necesidad se hace cada vez más imprescindible a medida que aumentan dichos factores.

¿Compensa la creación de un cubo para los cálculos del proyecto? Depende, para el caso LO quizás no sea imprescindible ya que solo existe un identificador de elementos (Tipo de disp.) y la información a visualizar no permite muchas variantes. Para el caso PH en cambio gana mucho más significado debido a que tenemos una mayor variedad de combinaciones de elementos (Conf_Index, Technology...) y nos interesa que la información esté lo más resumida, sencilla y organizada posible.

Pero un cubo no solo ofrece opciones para extraer sus datos sino también nos ayuda a visualizar los datos de forma eficiente, aspecto anteriormente comentado, por lo que su creación siempre ofrece garantías, en este caso tanto para la parte LO como para la parte PH.

Para líneas futuras se podría buscar alguna forma de acceder o modificar la estructura del cubo (La tabla dinámica) mediante alguna interfaz gráfica específica, distinta a la representada en Excel. Otra posible línea futura es el uso de alguna herramienta que analice el cubo con más detalle, permitiendo sacar una gran cantidad de información útil que en principio no se reflejaría como “directa”. En definitiva, metodologías que optimicen el uso del cubo.

REFERENCIAS

- [1]. <http://www.webmining.cl/2011/01/proceso-de-extraccion-de-conocimiento/>
- [2]. https://www.incibe.es/blogs/post/Empresas/BlogSeguridad/Articulo_y_comentarios/mineria_datos_big_data_seguridad
- [3]. <http://www.techopedia.com/7/29678/technology-trends/what-is-the-difference-between-big-data-and-data-mining>
- [4]. <http://hipertextual.com/2014/10/big-data-2>
- [5]. <https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/>
- [6]. <http://www.lantares.com/blog/velocidad-variedad-y-volumen-las-3-magnitudes-clave-de-big-data>
- [7]. <http://www.winshuttle.es/big-data-historia-cronologica/>
- [8]. http://www.eldiario.es/turing/Big_Data-fondo_marino-ciencia-mapas_0_419608272.html
- [9]. <http://www.uoc.edu/web/esp/art/uoc/molina1102/molina1102.html>
- [10]. http://www.sinnexus.com/business_intelligence/
- [11]. http://telos.fundaciontelefonica.com/seccion=1268&idioma=es_ES&id=2013062110130001&activo=6.do
- [12]. http://www.strategyand.pwc.com/media/file/Strategyand_Benefiting-from-Big-Data_A-New-Approach-for-the-Telecom-Industry.pdf

- [13]. https://www.informatica.com/content/dam/informatica-com/global/amer/us/collateral/executive-brief/big-data-telecommunications_ebook_2190.pdf
- [14]. <http://www.elmundo.es/economia/2014/05/29/53860ac0e2704e361f8b4583.html>
- [15]. www.huawei.com/ilink/en/download/HW_323807
- [16]. <http://www2.rhernando.net/modules/tutorials/doc/bd/dw.html>
- [17]. http://www.sinnexus.com/business_intelligence/datamart.aspx
- [18]. <http://www.informatica-hoy.com.ar/telefonos-celulares/Cubo-OLAP-una-base-de-datos-multidimensional.php>
- [19]. <https://support.office.com/es-cl/article/Power-Pivot-an%C3%A1lisis-de-datos-eficaz-y-modelado-de-datos-en-Excel-d7b119ed-1b3b-4f23-b634-445ab141b59b?ui=es-ES&rs=es-CL&ad=CL>
- [20]. [https://msdn.microsoft.com/es-es/library/Ee634217\(v=SQL.120\).aspx](https://msdn.microsoft.com/es-es/library/Ee634217(v=SQL.120).aspx)
- [21]. <https://olappivottableextend.codeplex.com/>
- [22]. [https://msdn.microsoft.com/es-es/library/ms145514\(v=SQL.120\).aspx](https://msdn.microsoft.com/es-es/library/ms145514(v=SQL.120).aspx)
- [23]. <http://carminabakerlovesit.com/2013/11/03/sobrevive-al-exceso-de-informacion-infoxicacion/>
- [24]. http://laura-topicos.blogspot.com.es/2010_05_01_archive.html
- [25]. <http://www.ibmbigdatahub.com/infographic/four-vs-big-data>
- [26]. <http://www.ibmbigdatahub.com/blog/transforming-telecommunications-industry-customer-centricity>
- [27]. <http://www.athena-solutions.com/library/articles/info-mgmt/hubandspokeonthewane.shtml>
- [28]. <http://snowplowanalytics.com/documentation/tools/olap/>
- [29]. [http://cdn2.hubspot.net/hub/239039/file-1959511423-pdf/docs/\[IC\]_OFFER_EBOOK- Big data y sector telecomunicacionesv2.pdf](http://cdn2.hubspot.net/hub/239039/file-1959511423-pdf/docs/[IC]_OFFER_EBOOK- Big data y sector telecomunicacionesv2.pdf)
- [30]. <https://exceltotal.com/tablas-dinamicas-en-excel/>
- [31]. [https://msdn.microsoft.com/es-es/library/gg399157\(v=sql.110\).aspx](https://msdn.microsoft.com/es-es/library/gg399157(v=sql.110).aspx)
- [32]. <https://exceltotal.com/funciones/cubo/>
- [33]. [https://technet.microsoft.com/es-es/library/ff487170\(v=sql.105\).aspx](https://technet.microsoft.com/es-es/library/ff487170(v=sql.105).aspx)
- [34]. http://www.aulaclic.es/macros-excel/t_3_11.htm
- [35]. <https://cursovisualexcel.wordpress.com/2011/08/18/funciones/>
- [36]. <https://support.office.com/es-hn/article/VALORCUBO-funci%C3%B3n-VALORCUBO-8733da24-26d1-4e34-9b3a-84a8f00dcbe0?ui=es-ES&rs=es-HN&ad=HN>
- [37]. <https://support.office.com/es-co/article/C%C3%A1lculos-de-Power-Pivot-7e4475e2-85c2-4496-9045-431412137b37?ui=es-ES&rs=es-CO&ad=CO>
- [38]. <http://www.mrexcel.com/forum/excel-questions/844645-set-up-filter-olap-pivot-using-visual-basic-applications.html>
- [39]. <https://msdn.microsoft.com/en-us/library/ee634396.aspx>
- [40]. <https://exceltotal.com/introduccion-a-las-macros/>
- [41]. <https://cursovisualexcel.wordpress.com/2011/08/29/las-funciones-imputbox-y-msgbox/>
- [42]. <https://msdn.microsoft.com/es-es/library/wak0wfyt.aspx>
- [43]. <http://microsoft.public.es.vba.narkive.com/BTOL6QTx/reloj-de-arena-durante-procedimiento-vba-de-excel>
- [44]. <https://exceltotal.com/creando-una-funcion-vba/>
- [45]. <https://es.wikipedia.org/wiki/Pentaho>
- [46]. <http://www.pentaho.com/product/product-overview>

- [47]. <https://es.wikipedia.org/wiki/Hadoop>
- [48]. <http://www.enriquedans.com/2011/11/hadoop-el-elefante-omnipresente.html>
- [49]. [https://technet.microsoft.com/es-es/library/ms175609\(v=sql.90\).aspx](https://technet.microsoft.com/es-es/library/ms175609(v=sql.90).aspx)
- [50]. Paredes A and García AE. Energy efficiency in national Internet networks. Master Thesis in Business and Information Technologies. University of Cantabria; 2012. Available at URL: <http://hdl.handle.net/10902/3804>. Accessed 2014 Sep 30.

ANEXO

TABLAS

Tabla Total LO

Tabla filtrada LO (12AM y 1AM):

Tabla total PH (12AM):

(Siguiente parte de tabla de datos a las 12AM)

Tabla DAX LO (ESW, ESW1a, LC10):

	A	B	C		A	B	C		A	B	C	
	Tipo de disp	Hora	Consumo Watt/h		ESW1a	30/12/1899	0		ESW1a	30/12/1899	0	
1	ESW1	30/12/1899 0:00:00	495700		25	ESW1a 0:00:00	30/12/1899	0	25	ESW1a 1:00:00	30/12/1899	933
2	ESW1	30/12/1899 1:00:00	495700		26	ESW1a 1:00:00	30/12/1899	0	26	ESW1a 2:00:00	30/12/1899	1017.5
3	ESW1	30/12/1899 2:00:00	495700		27	ESW1a 2:00:00	30/12/1899	0	27	ESW1a 3:00:00	30/12/1899	1017.5
4	ESW1	30/12/1899 3:00:00	495700		28	ESW1a 3:00:00	30/12/1899	0	28	ESW1a 4:00:00	30/12/1899	1017.5
5	ESW1	30/12/1899 4:00:00	495700		29	ESW1a 4:00:00	30/12/1899	0	29	ESW1a 5:00:00	30/12/1899	1017.5
6	ESW1	30/12/1899 5:00:00	495700		30	ESW1a 5:00:00	30/12/1899	0	30	ESW1a 6:00:00	30/12/1899	933
7	ESW1	30/12/1899 6:00:00	495700		31	ESW1a 6:00:00	30/12/1899	0	31	ESW1a 7:00:00	30/12/1899	848.5
8	ESW1	30/12/1899 7:00:00	495700		32	ESW1a 7:00:00	30/12/1899	0	32	ESW1a 8:00:00	30/12/1899	933
9	ESW1	30/12/1899 8:00:00	495700		33	ESW1a 8:00:00	30/12/1899	0	33	ESW1a 9:00:00	30/12/1899	848.5
10	ESW1	30/12/1899 9:00:00	495700		34	ESW1a 9:00:00	30/12/1899	0	34	ESW1a 10:00:00	30/12/1899	848.5
11	ESW1	30/12/1899 10:00:00	495700		35	ESW1a 10:00:00	30/12/1899	0	35	ESW1a 11:00:00	30/12/1899	764
12	ESW1	30/12/1899 11:00:00	495700		36	ESW1a 11:00:00	30/12/1899	0	36	ESW1a 12:00:00	30/12/1899	848.5
13	ESW1	30/12/1899 12:00:00	495700		37	ESW1a 12:00:00	30/12/1899	0	37	ESW1a 13:00:00	30/12/1899	679.5
14	ESW1	30/12/1899 13:00:00	495700		38	ESW1a 13:00:00	30/12/1899	0	38	ESW1a 14:00:00	30/12/1899	595
15	ESW1	30/12/1899 14:00:00	495700		39	ESW1a 14:00:00	30/12/1899	0	39	ESW1a 15:00:00	30/12/1899	679.5
16	ESW1	30/12/1899 15:00:00	496500		40	ESW1a 15:00:00	30/12/1899	0	40	ESW1a 16:00:00	30/12/1899	679.5
17	ESW1	30/12/1899 16:00:00	496500		41	ESW1a 16:00:00	30/12/1899	0	41	ESW1a 17:00:00	30/12/1899	679.5
18	ESW1	30/12/1899 17:00:00	496500		42	ESW1a 17:00:00	30/12/1899	0	42	ESW1a 18:00:00	30/12/1899	595
19	ESW1	30/12/1899 18:00:00	496500		43	ESW1a 18:00:00	30/12/1899	0	43	ESW1a 19:00:00	30/12/1899	679.5
20	ESW1	30/12/1899 19:00:00	496500		44	ESW1a 19:00:00	30/12/1899	0	44	ESW1a 20:00:00	30/12/1899	679.5
21	ESW1	30/12/1899 20:00:00	496500		45	ESW1a 21:00:00	30/12/1899	0	45	ESW1a 22:00:00	30/12/1899	848.5
22	ESW1	30/12/1899 21:00:00	495700		46	ESW1a 22:00:00	30/12/1899	0	46	ESW1a 23:00:00	30/12/1899	933
23	ESW1	30/12/1899 22:00:00	495700		47	ESW1a 23:00:00	30/12/1899	0	47	ESW1a 23:00:00	30/12/1899	933
24	ESW1	30/12/1899 23:00:00	495700						139	LC10 23:00:00		

Tabla DAX PH (1, 70, 90):

A	B	C	Conf_Ind	Hora	Consumo Watt/hr	
1	30/12/1899	0	1589	70	30/12/1899	671011
2	1 0:00:00	0	1590	70	30/12/1899	671011
3	1 1:00:00	0	1591	70	30/12/1899	671011
4	1 2:00:00	0	1592	70	30/12/1899	671011
5	1 3:00:00	0	1593	70	30/12/1899	671011
6	1 4:00:00	0	1594	70	30/12/1899	671011
7	1 5:00:00	0	1595	70	30/12/1899	671011
8	1 6:00:00	0	1596	70	30/12/1899	671011
9	1 7:00:00	0	1597	70	30/12/1899	671011
10	1 8:00:00	0	1598	70	30/12/1899	671011
11	1 9:00:00	0	1599	70	30/12/1899	671011
12	1 10:00:00	0	1600	70	30/12/1899	671011
13	1 11:00:00	0	1601	70	30/12/1899	671011
14	1 12:00:00	0	1602	70	30/12/1899	671011
15	1 13:00:00	0	1603	70	30/12/1899	671011
16	1 14:00:00	0	1604	70	30/12/1899	671011
17	1 15:00:00	0	1605	70	30/12/1899	671011
18	1 16:00:00	0	1606	70	30/12/1899	671011
19	1 17:00:00	0	1607	70	30/12/1899	671011
20	1 18:00:00	0	1608	70	30/12/1899	671011
21	1 19:00:00	0	1609	70	30/12/1899	671011
22	1 21:00:00	0	1610	70	30/12/1899	671011
23	1 22:00:00	0	1611	70	30/12/1899	671011
24	1 23:00:00	0				

Tabla de resultados LO:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
1	Tipo de disc	Consumo w																					
2	ESW1	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	495700	
3	ESW1a	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	ESW2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	ESW3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	ESW4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	LC10	848,5	933	1017,5	1017,5	1017,5	1017,5	933	848,5	933	848,5	848,5	764	848,5	848,5	764	848,5	848,5	764	848,5	848,5	764	848,5
8	LC11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	LC12	18024	17345	16666	16666	16666	16666	17345	18024	18703	190425	197215	20061	20061	20061	20061	20061	20061	20061	20061	20061	20061	
10	LC13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	LC14	16214	1890,8	16214	1352	1352	1890,8	16214	1890,8	1890,8	1890,8	1890,8	1352	1352	1352	1352	1352	1352	1352	1352	1352	1352	1352
12	LC15	1082,6	3	3	3	3	3	3	3	3	1082,6	16224	16224	16224	16224	16224	16224	16224	16224	16224	16224	16224	16224
13	LC16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	LC17	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
15	LC18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	LC23	547	772	844	844	844	772	547	613	673	745	613	646	646	646	646	646	646	646	646	646	646	646
17	LC34	338	169	338	338	338	169	338	169	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	LC35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	LC36	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
20	LC37	1632	819	548	548	548	1361	1632	1903	2174	1903	1903	1903	1903	1903	1903	1903	1903	1903	1903	1903	1903	
21	LC38	1088	1088	1088	1088	1088	1088	1088	1088	1088	1088	1088	1631	2177	2177	2177	2177	2177	2177	2177	2177	2177	2177
22	LC39	239	6	6	6	6	6	239	472	472	472	472	472	472	472	472	472	472	472	472	472	472	472
23	LC40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	LC41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	LC42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	LC43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	LC58	598	70	202	598	598	70	598	70	268	268	268	136	136	136	136	136	136	136	136	136	136	136
28	LC59	0	338	1358	1358	1358	169	338	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	LC60	0	679	679	679	679	679	135	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	LC61	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
31	LC62	271	0	542	542	542	0	271	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	LC63	3809	3266	1094	551	551	3266	3809	4352	4895	4895	4895	4895	4895	4895	4895	4895	4895	4895	4895	4895	4895	4895
33	LC64	470	1402	1402	1402	1402	470	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
34	LC65	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
35	LC66	0	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	
36	LC67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
37	LC68	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
38	LC7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
39	LC8	17487	17388	17289	17289	17289	17289	17289	17223	17223	17223	17223	17223	17223	17223	17223	17223	17223	17223	17223	17223	17223	
40	LC9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
41	LER1	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	80000	
42	LER2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
43	LSR1	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	40000	
44	LSR2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
45	LSR3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabla de resultados PH:

Tabla technology:

PROGRAMAS VBA

RecData_LO:

```
Sub RecData_LO()
    With Application      'Para optimizar la ejecución del programa. Deshabilitamos eventos,
    alertas...
        .Cursor = xlWait
        .ScreenUpdating = False
        .EnableEvents = False
        .Visible = True
        .DisplayAlerts = False
        .DisplayStatusBar = True
    End With

    Call barra(0) 'Barra de progreso.

    Dim z As Integer 'Variable que servirá como índice de vectores.
    Dim p(500), q(500) As String 'Variables que almacenarán texto introducido manualmente.
    Dim s, t, x As Variant 'Variables que almacenarán nombres de columnas, nombres de
    dispositivos y valores numéricicos.
    'Son de tipo "Variant" debido a que es un tipo necesario para igualar matrices, igualando
    a las funciones que se encargan de almacenar la información.
    Dim f1 As Integer 'índices que indican la posición de los límites superior de las filas,
    actualizándose para recorrer los distintos bloques.
    Dim f2 As Integer

    p(0) = "12AM_LO" 'Creamos un vector con el nombre de la hojas las cuales vamos a acceder.
    p(1) = "1AM_LO"
    p(2) = "2AM_LO"
    p(3) = "3AM_LO"
    p(4) = "4AM_LO"
    p(5) = "5AM_LO"
    p(6) = "6AM_LO"
    p(7) = "7AM_LO"
    p(8) = "8AM_LO"
    p(9) = "9AM_LO"
    p(10) = "10AM_LO"
    p(11) = "11AM_LO"
    p(12) = "12PM_LO"
    p(13) = "1PM_LO"
    p(14) = "2PM_LO"
    p(15) = "3PM_LO"
    p(16) = "4PM_LO"
    p(17) = "5PM_LO"
    p(18) = "6PM_LO"
    p(19) = "7PM_LO"
    p(20) = "8PM_LO"
    p(21) = "9PM_LO"
    p(22) = "10PM_LO"
    p(23) = "11PM_LO"

    q(0) = "12 AM" 'Vector que contiene las horas.
    q(1) = "1 AM"
    q(2) = "2 AM"
    q(3) = "3 AM"
    q(4) = "4 AM"
    q(5) = "5 AM"
    q(6) = "6 AM"
    q(7) = "7 AM"
    q(8) = "8 AM"
    q(9) = "9 AM"
    q(10) = "10 AM"
    q(11) = "11 AM"
    q(12) = "12 PM"
    q(13) = "1 PM"
    q(14) = "2 PM"
    q(15) = "3 PM"
    q(16) = "4 PM"
    q(17) = "5 PM"
```

```

q(18) = "6 PM"
q(19) = "7 PM"
q(20) = "9 PM"
q(21) = "10 PM"
q(22) = "11 PM"

Workbooks.Open Filename:="C:\Users\Jesús\Desktop\resultados_detalle.xlsx"
Workbooks("resultados_detalle.xlsx").Activate 'Abrimos y activamos el libro donde se
encuentran las hojas con información.
'Obtención de los nombres de las columnas y los dispositivos.

Workbooks("resultados_detalle.xlsx").Worksheets(p(0)).Activate 'Activación de la primera
hoja establecida en el array "12AM_LO".

s = recol_LO() 'Obtención del nombre de las columnas (no todas, solo las necesarias).
t = recnomb_LO() 'Obtención del nombre de los dispositivos.

ThisWorkbook.Worksheets("Tabla_LO").Activate 'Activación de la hoja la cual estará la
tabla total.

Call barra(1) 'Barra de progreso.

Call intcol_LO(s) 'En la hoja destino introducimos el nombre de las columnas.

f1 = 2 'Índices que representan la primera fila y la última del bloque de datos en una
sola hora.
f2 = 102

For y = 0 To 22 'Ciclo que irá recogiendo y depositando datos a medida hasta recorrer
todas las horas.

    Workbooks("resultados_detalle.xlsx").Worksheets(p(y)).Activate 'Activamos la hoja
correspondiente para extraer sus datos.

    x = recinf_LO() 'Recogemos la información más importante, la numérica referida al número
de puertos, el consumo, número de tarjetas.

    ThisWorkbook.Worksheets("Tabla_LO").Activate 'Activación de hoja adecuada para
inserción de datos, en el libro destino,
'el cuál ejecutará este código.

    Call intinf_LO(f1, f2, t, x, q, y) 'Función más relevante, introducimos en nuestra
tabla destino los valores de los dispositivos.
    'Los valores asignados a cada dispositivo y la hora correspondiente,
    'todo con los índices necesarios para establecer el lugar de la tabla y extraer la hora
correcta.

    'Actualizar índices para ir recorriendo nuestra tabla, los índices van representando
los valores límite de cada fila en cada bloque,
    'siendo las columnas constantes.
    f1 = f1 + 101
    f2 = f2 + 101

Next y

Call barra(2) 'Barra de progreso.

Call ceros_LO

'En esta parte crearemos una hoja nueva con la tabla original filtrada con valores no
deseados eliminados.

Call filtrar(q, s) 'Necesitamos q y s ya que son valores que permanecen igual en la tabla
sin filtrar y en la filtrada.

Call barra(3) 'Barra de progreso.

Workbooks("resultados_detalle.xlsx").Close 'Cerramos el libro que contiene tablas.

With Application 'Restablecemos los valores.

```

```

.Cursor = xlDefault
.ScreenUpdating = True
.EnableEvents = True
.DisplayAlerts = True
'.DisplayStatusBar=False

End With

End Sub

Function recol_LO() 'Función que recoge los nombres de las columnas, solo extraeremos los
nombres necesarios para su uso.

'Dichos nombres se encuentran en la segunda fila.
'Las columnas van variando, no estando todas seguidas, teniendo cuidado al seleccionar
las correctas.
'Todo se irá almacenando en un único vector.

Dim kl(20) As String 'Son valores no numéricos, tipo String.

z = 0 'Inicialización del índice del vector.

'Ports per LC
kl(z) = Cells(2, 3).Value
z = z + 1

'Consumo Unitario Puerto: Activo, AdaptETH, StandBY Consumo Unitario PIU Watt (Por
Plugin): Consumo Unitario PIU Watt, Consumo PIU StandBy.
For j = 7 To 11
    kl(z) = Cells(2, j).Value
    z = z + 1
Next j

'Number of LC, Number of PIU.
For j = 13 To 24
    kl(z) = Cells(2, j).Value
    z = z + 1
Next j

recol_LO = kl

End Function

Function recnomb_LO() 'Función que recoge el nombre de los dispositivos.

Dim kl(150) As String 'Los dispositivos son todos de tipo String (ESW1, ESW1a, LC7...).

z = 0 'Inicialización del índice del vector.
For i = 3 To 103
    kl(z) = Cells(i, 1).Value 'Se encuentran en la primera columna de la tabla.
    z = z + 1
Next i

recnomb_LO = kl

End Function

Sub intcol_LO(s As Variant) 'Esta subrutina sirve para introducir el nombre de las columnas
en la hoja destino.

'Todo se podrá haber hecho en un único ciclo.

'Tipo de dispositivo (ESW1, ESW1a...).
Cells(1, 1).Value = "Tipo de disp"

z = 0 'Inicialización del índice del vector.

'Ports per LC.

```

```

'Consumo Unitario Puerto: Activo, AdaptETH, StandBY Consumo Unitario PIU Watt (Por
Plugin): Consumo Unitario PIU Watt, Consumo PIU StandBy.
'Number of LC, Number of PIU.

For j = 2 To 19
    Cells(1, j).Value = s(z)
    z = z + 1
Next j

'Introducción de la palabra Hora, no existente en las hojas las cuales extraemos los datos.
Cells(1, 20).Value = "Hora"

End Sub

Function recinf_LO() 'Esta función sirve para recoger los datos numéricos de las columnas
de la tabla de cada hoja.

Dim kl(5000) As Double 'Necesitamos que el tipo de dato sea Double debido a la existencia
de valores enteros.

'Las filas serán constantes, iguales en cada hoja, al igual que las columnas.
'Todo se almacenará en un vector.

z = 0 'Inicialización del índice del vector.

'Ports per LC.
For i = 3 To 103
    kl(z) = Cells(i, 3).Value
    z = z + 1
Next i

'Consumo Unitario Puerto: Activo, AdaptETH, StandBY Consumo Unitario PIU Watt (Por
Plugin): Consumo Unitario PIU Watt, Consumo PIU StandBy.
For i = 3 To 103
    For j = 7 To 11
        kl(z) = Cells(i, j).Value
        z = z + 1
    Next j
Next i

'Number of LC, Number of PIU.
For i = 3 To 103
    For j = 13 To 24
        kl(z) = Cells(i, j).Value
        z = z + 1
    Next j
Next i

recinf_LO = kl

End Function

Sub intinf_LO(f1 As Integer, f2 As Integer, t As Variant, x As Variant, q() As String,
ByVal k As Integer)
'Esta función sirve para introducir la información en la hoja destino.

z = 0 'Inicialización del índice del vector.

'Nombres de dispositivos.
For i = f1 To f2
    Cells(i, 1).Value = t(z)
    z = z + 1
Next i

'Ports per LC.
z = 0
For i = f1 To f2
    Cells(i, 2).Value = x(z)
    z = z + 1
Next i

```

```

'Consumo Unitario Puerto: Activo, AdaptETH, StandBY Consumo Unitario PIU Watt (Por
Plugin): Consumo Unitario PIU Watt, Consumo PIU StandBy.
For i = f1 To f2
    For j = 3 To 7
        Cells(i, j).Value = x(z)
        z = z + 1
    Next j
Next i

'Number of LC, Number of PIU.
For i = f1 To f2
    For j = 8 To 19
        Cells(i, j).Value = x(z)
        z = z + 1
    Next j
Next i

'Hora, el valor vendrá dado por el índice del ciclo, que coincide con la posición del
array la cual contiene la hora correspondiente.
For i = f1 To f2
    Cells(i, 20).Value = q(k)
Next i

End Sub

Sub ceros_LO()

For i = 2 To 1013 'Rellenar los valores en blanco con valores 0.
    For j = 2 To 19

        If IsEmpty(Cells(i, j).Value) Then
            Cells(i, j).Value = 0
        End If

    Next j
Next i

End Sub

Sub filtrar(q() As String, s As Variant) 'Esta función sirve para filtrar la tabla total,
eliminando valores no deseados.
'y representando en una hoja aparte aquellos que si necesitamos.

f1 = 2 'Inicializamos los índices.
f2 = 102

Dim bs(5000) As String 'Nombres de los dispositivos.
Dim b(1000000) As Double 'Valores de los dispositivos.
Dim z1 As Double 'Índice específico para uno de los vectores.

ThisWorkbook.Worksheets("Tabla_LO").Activate
'Recoger valores de dispositivos que varían siempre.
z = 0 'Inicializar índices de vectores, uno para la información de texto y otro para la
información numérica.
z1 = 0

For n = 1 To 23 'Ciclo que extrae los valores requeridos.

    For i = f1 To f2 'Filas.
        For j = 1 To 19 'Columnas.

            If j = 1 Then 'Si estamos en la columna uno, la que contiene los nombres de los
dispositivos.
                If IsEmpty(Cells(i, 1).Value) = False And Cells(i, 1).Value <> 0 Then 'Si el
dispositivo no es "vacío" o "0".
                    bs(z) = Cells(i, j).Value
                    z = z + 1
                End If
            Else 'Si estamos en una columna que indica información numérica.
                If IsEmpty(Cells(i, 1).Value) = False And Cells(i, 1).Value <> 0 Then 'Si el
dispositivo al cuál pertenece dicha información no es "vacío" o "0".

```

```

        b(z1) = Cells(i, j).Value
        z1 = z1 + 1
    End If
End If

Next j

Next i

f1 = f1 + 101 'Actualizar filas.
f2 = f2 + 101

Next n

'Insertamos los valores en la hoja 23 veces.
ThisWorkbook.Worksheets("Tabla_LO_filtrada").Activate

z = 0
For i = 2 To 1013 'Filas totales de nuestra tabla, 1013 es el valor total, el valor real
será menor ya que hemos filtrado valores.
    'Sin columnas ya que solo es valor de nombre de dispositivos.
    Cells(i, 1).Value = bs(z)
    z = z + 1
Next i

z1 = 0
For i = 2 To 1013 'Filas totales de nuestra tabla, reales son menos, valor "simbólico".
    For j = 2 To 19 'Columnas totales.
        Cells(i, j).Value = b(z1)
        z1 = z1 + 1
    Next j
Next i

'Inserción de hora correctamente.
z = 0
f1 = 2 'índices inicial y final del bloque de elementos filtrado, inferior al 100 inicial.
f2 = 45

For i = 2 To 24
    For j = f1 To f2
        ThisWorkbook.Worksheets("Tabla_LO_filtrada").Cells(j, 20).Value = q(z)
    Next j
    z = z + 1
    f1 = f1 + 44 'Actualización de índices.
    f2 = f2 + 44
Next i

'Introducción nombre de columnas.
z = 0
For j = 2 To 19
    Cells(1, j).Value = s(z)
    z = z + 1
Next j

'Valores introducidos manualmente no existentes en las hojas.
Cells(1, 1).Value = "Tipo de disp"
Cells(1, 20).Value = "Hora"

End Sub

```

RecData_PH:

Código casi idéntico que RecData_LO.

Las diferencias más importantes son:

- Las páginas que recorremos son 12AM_PH, 1AM_PH...
- Se introducen 4 valores identificadores en cada iteración del ciclo en vez de 1.
- No es necesario filtrar la tabla.

calculos_LO:

```
Sub calculos_LO()

With Application 'Para optimizar la ejecución del programa. Desabilitamos eventos, alertas...
    .Cursor = xlWait
    .ScreenUpdating = False
    .EnableEvents = False
    .Visible = True
    .DisplayAlerts = False
    .DisplayStatusBar = True
End With

Call barra(0)

Dim z As Integer 'Índices de vectores.
Dim z1 As Integer
Dim z2 As Integer
Dim z3 As Integer
Dim q1(1000000) As Double 'Vectores resultados de operación.
Dim q2(1000000) As Double
Dim q3(1000000) As Double
Dim vsiete As Variant 'Matriz que almacenará Suma de LC y Suma de PIU.
Dim c As Boolean 'Variable de "cambio".
Dim w As Integer 'Decenas.
Dim y As Integer 'Unidades.

'Activamos la lámina de cubo correspondiente a las 7PM.
ThisWorkbook.Worksheets("Cubo_LO").Activate
ThisWorkbook.Worksheets("Cubo_LO").PivotTables("Tabla
dinámica2").PivotFields("[Tabla_LO_filtrada].[Hora].[Hora]").CurrentPageName =
"[Tabla_LO_filtrada].[Hora].&[1899-12-30T19:00:00]"

Call barra(1)

vsiete = vsiete_LO() 'Extraemos los valores de Suma de LC y Suma de PIU debido a que les
necesitaremos posteriormente.

z1 = 0 'Inicializar Índices de los vectores resultado.
z2 = 0
z3 = 0

c = True 'Variable booleana que distinguirá entre los valores de las decenas 1 o 2.
w = 0 'Inicializar decenas.
y = 0 'Inicializar unidades.

Call barra(2)

For k = 1 To 23 'Iteración para cada hora, es decir, cada lámina del cubo.

    Call Chequeo(w, y, c) 'Comprobamos los valores de la hora que establecemos, para
    comprobar si tenemos que aumentar las decenas y reiniciar las unidades.
    'Teniendo en cuenta que las 20:00 no existe en nuestro cubo.

    'Activamos la lámina del cubo correcta mediante w e y, formarán parte de la identificación
    de la lámina, que cambiará en cada iteración.
    'Convirtiéndose en un valor tipo texto para que pueda formar parte de la identificación.

    ThisWorkbook.Worksheets("Cubo_LO").Activate
    ThisWorkbook.Worksheets("Cubo_LO").PivotTables("Tabla
    dinámica2").PivotFields("[Tabla_LO_filtrada].[Hora].[Hora]").CurrentPageName =
    "[Tabla_LO_filtrada].[Hora].&[1899-12-30T" + CStr(w) + CStr(y) + ":00:00]"

    Call Operaciones_LO(q1, q2, q3, z1, z2, z3, vsiete) 'Realizar las operaciones adecuadas.

    y = y + 1 'Aumentar unidades.
```

```

Next k

Call insval_LO(q1, q2, q3) 'Establecer nuestros valores en una tabla aparte, será la tabla
de resultados.

Call barra(3)

With Application 'Reestablecer valores.

    .Cursor = xlDefault
    .ScreenUpdating = True
    .EnableEvents = True
    .DisplayAlerts = True
    '.DisplayStatusBar = False

End With

End Sub

Function vsiete_LO() 'Esta subrutina sirve para almacenar los valores de Suma de LC y Suma
de PIU, necesarios en posteriores cálculos.

    Dim lcpiu1(2, 200) As Integer

    z = 0
    For i = 4 To 47
        lcpiu1(0, z) = Cells(i, 20).Value 'Valores Suma de LC
        z = z + 1
    Next i

    'Suma de PIU
    z = 0
    For i = 4 To 47
        lcpiu1(1, z) = Cells(i, 21).Value 'Valores Suma de PIU
        z = z + 1
    Next i

    vsiete_LO = lcpiu1

End Function

Sub Chequeo(w As Integer, y As Integer, c As Boolean) 'Esta subrutina sirve para comprobar
los valores de la hora, representados en decenas y unidades.

    If y > 9 And c = True Then
        w = 1 'Aumentar decenas
        y = 0 'Reiniciar unidades
        c = False 'Cambiar valor para el próximo cambio

    Else

        If y > 9 And c = False Then
            w = 2 'Aumentar decenas
            y = 1 'Reiniciar unidades
        End If

    End If

End Sub

Sub Operaciones_LO(q1() As Double, q2() As Double, q3() As Double, z1 As Integer, z2 As
Integer, z3 As Integer, lcpiu As Variant) 'Esta subrutina sirve
'para realizar las operaciones en cada lámina del cubo.

    'Habrá una operación diferente por cada grupo de dispositivos. Cada operación se almacenará
    en un vector diferente.
    'Habrá un resultado diferente por cada fila, obteniendo cada valor de las columnas
    correspondientes.

```

```

'z Servirá para recorrer el vector de las 7PM (lcpiu), reiniciándose en cada hora, mientras
que z1, z2, z3 serán los índices de los valores resultado,
'acumulándose siempre.

z = 0

For s = 4 To 8
    'Operación grupo ESW.
    q1(z1) = Cells(s, 21).Value * Cells(s, 6).Value + (lcpiu(1, z) - Cells(s, 21).Value) *
    Cells(s, 7).Value
    z1 = z1 + 1
    z = z + 1
Next s

For s = 9 To 42
    'Operación grupo LC.
    q2(z2) = Cells(s, 2).Value * Cells(s, 3).Value * Cells(s, 20).Value + (lcpiu(0, z) -
    Cells(s, 20).Value) * Cells(s, 5).Value
    z2 = z2 + 1
    z = z + 1
Next s

For s = 43 To 47 'Operación grupo LER y LSR.
    q3(z3) = Cells(s, 21).Value * Cells(s, 6).Value + (lcpiu(1, z) - Cells(s, 21).Value) *
    Cells(s, 7).Value
    z3 = z3 + 1
    z = z + 1
Next s

End Sub

Sub insval_LO(q1() As Double, q2() As Double, q3() As Double) 'Esta subrutina sirve para
insertar los valores en la hoja resultado.

Dim a(1000) As String

'Los valores de los nombres de los dispositivos se extraerán de la última lámina del cubo
activa.
z = 0
For i = 4 To 47
    a(z) = Cells(i, 1).Value
    z = z + 1
Next i

ThisWorkbook.Worksheets("Cálculos_VBA_LO").Activate 'Hoja que contiene la tabla resultado.
z = 0
For i = 2 To 45
    Cells(i, 1).Value = a(z)
    z = z + 1
Next i

Cells(1, 1).Value = "Tipo de disp"           'Valores correspondientes a las horas,
                                                 'columnas de la tabla.
Cells(1, 2).Value = "Consumo Watt/hr 12AM"
Cells(1, 3).Value = "Consumo Watt/hr 1AM"
Cells(1, 4).Value = "Consumo Watt/hr 2AM"
Cells(1, 5).Value = "Consumo Watt/hr 3AM"
Cells(1, 6).Value = "Consumo Watt/hr 4AM"
Cells(1, 7).Value = "Consumo Watt/hr 5AM"
Cells(1, 8).Value = "Consumo Watt/hr 6AM"
Cells(1, 9).Value = "Consumo Watt/hr 7AM"
Cells(1, 10).Value = "Consumo Watt/hr 8AM"
Cells(1, 11).Value = "Consumo Watt/hr 9AM"
Cells(1, 12).Value = "Consumo Watt/hr 10AM"
Cells(1, 13).Value = "Consumo Watt/hr 11AM"
Cells(1, 14).Value = "Consumo Watt/hr 12PM"
Cells(1, 15).Value = "Consumo Watt/hr 1PM"
Cells(1, 16).Value = "Consumo Watt/hr 2PM"
Cells(1, 17).Value = "Consumo Watt/hr 3PM"
Cells(1, 18).Value = "Consumo Watt/hr 4PM"
Cells(1, 19).Value = "Consumo Watt/hr 5PM"

```

```

Cells(1, 20).Value = "Consumo Watt/hr 6PM"
Cells(1, 21).Value = "Consumo Watt/hr 7PM"
Cells(1, 22).Value = "Consumo Watt/hr 9PM"
Cells(1, 23).Value = "Consumo Watt/hr 10PM"
Cells(1, 24).Value = "Consumo Watt/hr 11PM"

z1 = 0 'Inicialización de indices de vectores resultados.
z2 = 0
z3 = 0

For j = 2 To 24 'Cada columna significa una hora diferente.
'En cada fila insertaremos los valores.

    For i = 2 To 6 'ESW
        Cells(i, j).Value = q1(z1)
        z1 = z1 + 1
    Next i

    For i = 7 To 40 'LC
        Cells(i, j).Value = q2(z2)
        z2 = z2 + 1
    Next i

    For i = 41 To 45 'LER y LSR
        Cells(i, j).Value = q3(z3)
        z3 = z3 + 1
    Next i

Next j

End Sub

```

calculos_DAX_LO:

```
Sub calculos_DAX_LO()

Dim q(30) As String 'Vector nombre de columnas
Dim z As Integer 'Índice de vectores
Dim s As Variant 'Vector nombre de dispositivos tipo Variant para igualar a matrices.
Dim x As Variant 'Vector que almacena los consumos.

Call barra(0) 'Barra de progreso.

q(0) = "Consumos 12 AM" 'Vector que contiene el nombre de las columnas.
q(1) = "Consumos 1 AM"
q(2) = "Consumos 2 AM"
q(3) = "Consumos 3 AM"
q(4) = "Consumos 4 AM"
q(5) = "Consumos 5 AM"
q(6) = "Consumos 6 AM"
q(7) = "Consumos 7 AM"
q(8) = "Consumos 8 AM"
q(9) = "Consumos 9 AM"
q(10) = "Consumos 10 AM"
q(11) = "Consumos 11 AM"
q(12) = "Consumos 12 PM"
q(13) = "Consumos 1 PM"
q(14) = "Consumos 2 PM"
q(15) = "Consumos 3 PM"
q(16) = "Consumos 4 PM"
q(17) = "Consumos 5 PM"
q(18) = "Consumos 6 PM"
q(19) = "Consumos 7 PM"
q(20) = "Consumos 8 PM"
q(21) = "Consumos 9 PM"
q(22) = "Consumos 10 PM"
q(23) = "Consumos 11 PM"

'Recogida de dispositivos

ThisWorkbook.Worksheets("Tabla_DAX_LO").Activate 'Activamos la tabla que contiene los
valores a extraer.

Call barra(1) 'Barra de progreso.

s = Recdisp_LO() 'Extraemos los nombres de los dispositivos.
x = Recinf_LO() 'Recogemos los consumos.

ThisWorkbook.Worksheets("Cálculos_DAX_LO").Activate 'Activamos la hoja resultado.

Call barra(2) 'Barra de progreso.

Call insinf_LO(s, x, q) 'Insertamos la información

Call barra(3) 'Barra de progreso.

End Sub

Function Recdisp_LO() 'Esta función sirve para recoger los nombres de dispositivos.
'La disposición de los nombres será diferente cada 23 valores, ya que cada fila seguirá
el orden disp-hora-valor repitiéndose el valor de disp
'23 veces para cada hora distinta.

Dim s1(500) As String
Dim m As Integer

z = 0
m = 2 'Primer índice de fila.
For i = 1 To 44 'Número total de dispositivos
    s1(z) = Cells(m, 1).Value
    m = m + 23 'Cambiar de dispositivo.
    z = z + 1

```

```

Next i

Recdisp_LO = s1

End Function

Function Recinf_LO() 'Esta subrutina sirve para recoger la información de los consumos.

Dim x1(10000) As Double

z = 0
For i = 2 To 1013 'Número de filas totales.
    x1(z) = Cells(i, 3).Value
    z = z + 1
Next i

Recinf_LO = x1

End Function

Sub insinf_LO(s As Variant, x As Variant, q() As String) 'Esta subrutina sirve para
"enmarcar" la tabla de resultados mediante la introducción del nombre de los
'dispositivos y el nombre de los consumos.

Cells(1, 1).Value = "Tipo de disp." 'Valor que introducimos manualmente.

z = 0
For i = 2 To 45
    Cells(i, 1).Value = s(z)
    z = z + 1
Next i

z = 0
For j = 2 To 24
    Cells(1, j).Value = q(z)
    z = z + 1
Next j

z = 0
For i = 2 To 45 'Valores para cada dispositivo.
    For j = 2 To 24 'Valores para cada hora.
        Cells(i, j).Value = x(z)
        z = z + 1
    Next j
Next i

End Sub

```

calculos_PH:

Código casi idéntico a Calculos_LO.
Las diferencias más importantes son:

- Los datos que se extraen a las 7 PM son distintos.
- El "identificador OLAP" será distinto.
- Solo realizamos una operación en vez de 3.

calculos_LO_hora:

Código casi idéntico a `calculos_PH_hora`.
Diferencias más importantes:

- No es necesario realizar un paso de conversión de los valores a introducir en el "Identificador OLAP".
- Menor número de operaciones.

calculos_PH_hora:

```
Sub calculos_PH_hora()

With Application 'Para optimizar la ejecución del programa. Deshabilitamos eventos, alertas...
    .Cursor = xlWait
    .ScreenUpdating = False
    .EnableEvents = False
    .Visible = True
    .DisplayAlerts = False
    .DisplayStatusBar = True
End With

Call barra(0)

Dim z As Integer 'Índice de vector.
Dim x As Integer 'Número de identificadores no repetidos.
Dim uno(15) As String 'Vectores que contienen las diferentes partes de un identificador OLAP.
Dim dos(15) As String
Dim tres(150) As String
Dim cuatro(5) As String
Dim p(500) As String 'Nombre de los identificadores OLAP al completo.
Dim q, c As Variant 'Vector de identificadores Y de identificadores de forma numérica.
Dim s(1000) As String 'Vector de identificadores sin repetir.
Dim k As Integer 'Índices de vectores que irán recorriendo los diferentes vectores que componen los identificadores OLAP.
Dim l As Integer
Dim b As Integer
Dim v(1000000) As Double 'Vector de resultado de operaciones.

Workbooks.Open ("C:\Users\Jesús\Desktop\Origen de datos.xlsx")
Workbooks("Origen de Datos.xlsx").Worksheets("Tabla_PH").Activate 'Activar libro donde está la hoja origen de datos.

Call barra(1)

q = recdisp_PH2() 'Recogemos los dispositivos.

Call elmrep_PH(q, s, x) 'Eliminamos los repetidos.

c = convnum(s, x) 'los identificadores son números, para un correcto procesado posterior, los convertimos en números.

Call estval(uno, dos, tres, cuatro) 'Rellenamos los vectores que servirán para crear los identificadores válidos OLAP.

k = 0 'Inicializamos los índices de los vectores identificadores antes de entrar en el ciclo.
l = 0
b = 0

For i = 0 To x 'Valores numéricos

    Call cambform(p, c, k, l, b, i, uno, dos, tres, cuatro) 'Conversión a formato OLAP de cada valor identificador para cada valor numérico.

    Next i

    Workbooks("Origen de datos.xlsx").Close 'Cerramos el libro origen de datos.

    Call barra(2)

    z = 0 'Inicializamos el índice del vector de resultados.
    For i = 0 To x
```

```

'Vamos activando cada lámina del cubo.
ThisWorkbook.Worksheets("Cubo_PH_hora").Activate
ThisWorkbook.Worksheets("Cubo_PH_hora").PivotTables("Tabla
dinámica4").PivotFields("[Tabla_PH].[Conf_Index].[Conf_Index]").CurrentPageName      =
"[Tabla_PH].[Conf_Index].&" + p(i)

Call operar_PH(v, z) 'Operamos, esta vez el ciclo le introducimos en la función.

Next i

ThisWorkbook.Worksheets("Cálculos_PH_hora").Activate 'Activamos la hoja resultado.

Call insdat_PH(s, v, x) 'Insertamos los valores filas, columnas y resultados.

Call barra(3)

With Application

    .Cursor = xlDefault
    .ScreenUpdating = True
    .EnableEvents = True
    .DisplayAlerts = True
    '.DisplayStatusBar = False

End With

End Sub

Function recdisp_PH2() 'Esta función sirve para extraer los identificadores de los
dispositivos.
Dim q1(1000) As String

'Extraer dispositivos.
z = 0
For i = 2 To 346
    q1(z) = Cells(i, 1).Value
    z = z + 1
Next i

recdisp_PH2 = q1

End Function

Sub elmrep_PH(q As Variant, s() As String, x As Integer) 'Esta subrutina sirve para crear
un vector sin valores identificativos repetidos.

Dim s1(1000) As String

'Ciclo de comparación.

For i = 0 To 344
    For j = (i + 1) To 344
        If q(i) = q(j) Then 'Convertir valores repetidos en 0.
            q(j) = "borrar"
        End If
    Next j
Next i

x = 0
For i = 0 To 344
    If q(i) <> "borrar" Then 'Crear un nuevo array con valores únicos.
        s(x) = q(i)
        x = x + 1 'Restar uno que se añade por exceso.
    End If
Next i

x = x - 1

```

```

End Sub

Sub estval(uno() As String, dos() As String, tres() As String, cuatro() As String) 'Esta
subrutina sirve para establecer valores en los vectores.
'El contenido de los vectores forma parte del identificador OLAP.

uno(0) = "1."
uno(1) = "2."
uno(2) = "3."
uno(3) = "4."
uno(4) = "5."
uno(5) = "6."
uno(6) = "7."
uno(7) = "8."
uno(8) = "9."

dos(0) = ""
dos(1) = "1"
dos(2) = "2"
dos(3) = "3"
dos(4) = "4"
dos(5) = "5"
dos(6) = "6"
dos(7) = "7"
dos(8) = "8"
dos(9) = "9"

tres(0) = ""
tres(1) = "01"
tres(2) = "02"
tres(3) = "03"
tres(4) = "04"
tres(5) = "05"
tres(6) = "06"
tres(7) = "07"
tres(8) = "08"
tres(9) = "09"
tres(10) = "1"
tres(11) = "11"
tres(12) = "12"
tres(13) = "13"
tres(14) = "14"
tres(15) = "15"
tres(16) = "16"
tres(17) = "17"
tres(18) = "18"
tres(19) = "19"
tres(20) = "2"
tres(21) = "21"
tres(22) = "22"
tres(23) = "23"
tres(24) = "24"
tres(25) = "25"
tres(26) = "26"
tres(27) = "27"
tres(28) = "28"
tres(29) = "29"
tres(30) = "3"
tres(31) = "31"
tres(32) = "32"
tres(33) = "33"
tres(34) = "34"
tres(35) = "35"
tres(36) = "36"
tres(37) = "37"
tres(38) = "38"
tres(39) = "39"
tres(40) = "4"
tres(41) = "41"
tres(42) = "42"
tres(43) = "43"

```

```

tres(44) = "44"
tres(45) = "45"
tres(46) = "46"
tres(47) = "47"
tres(48) = "48"
tres(49) = "49"
tres(50) = "5"
tres(51) = "51"
tres(52) = "52"
tres(53) = "53"
tres(54) = "54"
tres(55) = "55"
tres(56) = "56"
tres(57) = "57"
tres(58) = "58"
tres(59) = "59"
tres(60) = "6"
tres(61) = "61"
tres(62) = "62"
tres(63) = "63"
tres(64) = "64"
tres(65) = "65"
tres(66) = "66"
tres(67) = "67"
tres(68) = "68"
tres(69) = "69"
tres(70) = "7"
tres(71) = "71"
tres(72) = "72"
tres(73) = "73"
tres(74) = "74"
tres(75) = "75"
tres(76) = "76"
tres(77) = "77"
tres(78) = "78"
tres(79) = "79"
tres(80) = "8"
tres(81) = "81"
tres(82) = "82"
tres(83) = "83"
tres(84) = "84"
tres(85) = "85"
tres(86) = "86"
tres(87) = "87"
tres(88) = "88"
tres(89) = "89"
tres(90) = "9"
tres(91) = "91"
tres(92) = "92"
tres(93) = "93"
tres(94) = "94"
tres(95) = "95"
tres(96) = "96"
tres(97) = "97"
tres(98) = "98"
tres(99) = "99"

cuatro(0) = "E1"
cuatro(1) = "E2"

End Sub

Function convnum(s As Variant, x As Integer) 'Esta función sirve para convertir en números
los valores identificativos, que son tipo array.

Dim c1(500) As Integer

For i = 0 To x
    If IsNumeric(s(i)) Then
        c1(i) = CInt(s(i))
    Else

```

```

c1(i) = 0 'Array en blanco = 0 para poder identificarle de alguna manera numérica.
End If
Next i

convnum = c1

End Function

Sub cambiform(p() As String, c As Variant, k As Variant, l As Integer, b As Integer, ByVal
i As Integer, uno() As String, dos() As String, tres() As String, cuatro() As String)
'Esta subrutina sirve para crear un vector de identificadores OLAP mediante la comparación
con el identificador numérico, convirtiéndole al formato
'correspondiente.

Select Case c(i)

Case 1 To 9
p(i) = "[" + uno(k) + "]" 'Valores del 1 al 9.
k = k + 1
If k > 8 Then 'Si sobrepasamos el 9, reiniciamos el vector.
k = 0
End If

Case 10 To 99
p(i) = "[" + uno(k) + dos(l) + cuatro(0) + "]"
l = l + 1 'Primero será dos quien irá cambiando del 0 al 9, siendo cuatro un valor
fijo, exponente E1.
If l > 9 Then
l = 0
k = k + 1 'Cambiamos decenas cada 10 unidades.
End If
If k > 8 Then
k = 0 'Reiniciamos decenas.
End If

Case 100 To 199
p(i) = "[" + uno(0) + tres(b) + cuatro(1) + "]"
b = b + 1 'tres tiene valores de dos cifras del 00 al 99, iremos cambiando, exponente
E2.
If b > 99 Then
b = 0
End If

Case 1s >= 200
p(i) = "[" + uno(1) + tres(b) + cuatro(1) + "]"
b = b + 1 'Cogemos el segundo valor de uno, que identifica al dos, volvemos a recorrer
tres y cuatro seguirá siendo E2.
If b > 99 Then
b = 0
End If

Case Else
p(i) = "" 'Valor en blanco, existente como filtro del cubo.

End Select

End Sub

Sub operar_PH(v() As Double, z As Integer) 'Esta subrutina sirve para crear un vector de
resultados.
'Al igual que en el caso LO, habrá una operación por lámina, repitiéndose 23 veces por
cada hora, existiendo una operación única.
'para cada identificador. Los valores de las 7PM estarán establecidos en la fila
correspondiente (23).

For j = 4 To 26 'Filas de la lámina del cubo.
v(z) = Cells(j, 29).Value + Cells(j, 30).Value + (Cells(23, 26).Value - Cells(j,
26).Value) * 0.5 + (Cells(23, 25).Value - Cells(j, 25).Value) * 0.5
z = z + 1
Next j

```

```

End Sub

Sub insdat_PH(s As Variant, v() As Double, x As Integer) 'Esta función sirve para introducir
los valores en la tabla resultado.

Dim g(30) As String 'Vector horas.

g(0) = "00:00:00" 'Vector Horas.
g(1) = "01:00:00"
g(2) = "02:00:00"
g(3) = "03:00:00"
g(4) = "04:00:00"
g(5) = "05:00:00"
g(6) = "06:00:00"
g(7) = "07:00:00"
g(8) = "08:00:00"
g(9) = "09:00:00"
g(10) = "10:00:00"
g(11) = "11:00:00"
g(12) = "12:00:00"
g(13) = "13:00:00"
g(14) = "14:00:00"
g(15) = "15:00:00"
g(16) = "16:00:00"
g(17) = "17:00:00"
g(18) = "18:00:00"
g(19) = "19:00:00"
g(20) = "21:00:00"
g(21) = "22:00:00"
g(22) = "23:00:00"

z = 0
For i = 2 To 24
    Cells(i, 1).Value = g(z) 'Columnas.
    z = z + 1
Next i

z = 0
For j = 2 To (x + 2)
    Cells(1, j).Value = s(z) 'Filas.
    z = z + 1
Next j

Cells(1, 1).Value = "Conf_Index" 'Nombre de primera columna.

z = 0
For i = 2 To (x + 2)
    For j = 2 To 24
        Cells(j, i).Value = v(z) 'Resultados.
        z = z + 1
    Next j
Next i

End Sub

```

calculos_PH_hora2:

Código casi idéntico a Calculos_PH_hora.

La única diferencia es que esta vez para extraer los valores que formen parte del filtro activaremos una lámina del Cubo_PH para extraerlos de allí en vez de extraerlos de la tabla PH.

calculos_PH_technology:

```
Sub calculos_PH_technology()

With Application 'Para optimizar la ejecución del programa. Deshabilitamos eventos, alertas...
    .Cursor = xlWait
    .ScreenUpdating = False
    .EnableEvents = False
    .Visible = True
    .DisplayAlerts = False
    .DisplayStatusBar = True

End With

Call barra(0)

Dim z As Integer 'Índice de vector resultado.
Dim z1 As Integer 'Índice de vector lrc.
Dim q(1000000) As Double 'Vector resultado.
Dim vsiete As Variant 'Vector valores 7PM de tipo Variant para ser igualado a una matriz.
Dim c As Boolean 'Variable contador.
Dim w As Integer 'Decenas de hora.
Dim y As Integer 'Unidades de hora.

'Activamos la lámina del cubo correspondiente a las 7PM.
ThisWorkbook.Worksheets("Cubo_PH_technology").Activate
ThisWorkbook.Worksheets("Cubo_PH_technology").PivotTables("Tabla dinámical").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName = "[Tabla_PH].[Hora].&[1899-12-30T19:00:00]"

Call barra(1)

vsiete = vsiete_PH() 'Extraemos los valores de Suma de L LC y Suma de R LC que necesitaremos en posteriores cálculos.

w = 0 'Inicializar decenas.
y = 0 'Inicializar unidades.
c = True 'Inicializar variable de cambio.

Call barra(2)

z = 0 'Inicializamos índice de vector resultado.
For k = 1 To 23

    Call Chequeo(w, y, c) 'Uso igual en el anterior programa.

    'Activamos cada lámina del cubo.
    ThisWorkbook.Worksheets("Cubo_PH_technology").Activate
    ThisWorkbook.Worksheets("Cubo_PH_technology").PivotTables("Tabla dinámical").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName = "[Tabla_PH].[Hora].&[1899-12-30T" + CStr(w) + CStr(y) + ":00:00]"
    'Valor 267 de la hora 11 PM ignorado.

    Call Operaciones_PH(q, vsiete, z) 'Operamos.

    y = y + 1

Next k

Call insval_PH(q) 'Introducimos los valores resultado en tabla resultado.

Call barra(3)

With Application 'Reestablecemos valores.

    .Cursor = xlDefault
    .ScreenUpdating = True
    .EnableEvents = True
```

```

    .DisplayAlerts = True
    ' .DisplayStatusBar = False

End With

End Sub

Function vsiete_PH() 'Esta función sirve para almacenar los valores Suma de L LC y Suma
de R LC de las 7PM.

Dim lrc1(2, 1000) As Integer 'Matriz bidimensional que almacenará los valores, cada
dimensión es una columna de la lámina del cubo.
'Poner especial cuidado en establecer la columna correcta.

z = 0
For i = 4 To 73
    lrc1(0, z) = Cells(i, 25).Value
    z = z + 1
Next i

z = 0
For i = 4 To 73
    lrc1(1, z) = Cells(i, 26).Value
    z = z + 1
Next i

vsiete_PH = lrc1

End Function

Sub Operaciones_PH(q() As Double, lrc As Variant, z As Integer) 'Esta subrutina sirve para
realizar las operaciones.
'Dichas operaciones serán almacenadas en un vector.
'z será el índice del vector mientras que z1 será el índice del vector de lrc que se
reiniciará en cada hora, siendo z acumulativo.
'Solo disponemos de una operación para todos los elementos.

z1 = 0

For i = 4 To 73
    q(z) = Cells(i, 29).Value + Cells(i, 30).Value + (lrc(1, z1) - Cells(i, 26).Value) *
0.5 + (lrc(0, z1) - Cells(i, 25).Value) * 0.5
    z = z + 1
    z1 = z1 + 1
Next i

End Sub

Sub insval_PH(q() As Double) 'Esta subrutina sirve para introducir el resultado en una
tabla aparte.

Dim a(1000) As String 'Vector que almacenará los identificadores.

'Extraeremos los valores de la última lámina del cubo activa.

z = 0
For i = 4 To 73
    a(z) = Cells(i, 1).Value
    z = z + 1
Next i

ThisWorkbook.Worksheets("Cálculos_PH_technology").Activate 'Hoja con la tabla resultado.

z = 0
For i = 2 To 71
    Cells(i, 1).Value = a(z)
    z = z + 1
Next i

Cells(1, 1).Value = "Technology" 'Valores columna.

```

```

Cells(1, 2).Value = "Consumo Watt/hr 12AM"
Cells(1, 3).Value = "Consumo Watt/hr 1AM"
Cells(1, 4).Value = "Consumo Watt/hr 2AM"
Cells(1, 5).Value = "Consumo Watt/hr 3AM"
Cells(1, 6).Value = "Consumo Watt/hr 4AM"
Cells(1, 7).Value = "Consumo Watt/hr 5AM"
Cells(1, 8).Value = "Consumo Watt/hr 6AM"
Cells(1, 9).Value = "Consumo Watt/hr 7AM"
Cells(1, 10).Value = "Consumo Watt/hr 8AM"
Cells(1, 11).Value = "Consumo Watt/hr 9AM"
Cells(1, 12).Value = "Consumo Watt/hr 10AM"
Cells(1, 13).Value = "Consumo Watt/hr 11AM"
Cells(1, 14).Value = "Consumo Watt/hr 12PM"
Cells(1, 15).Value = "Consumo Watt/hr 1PM"
Cells(1, 16).Value = "Consumo Watt/hr 2PM"
Cells(1, 17).Value = "Consumo Watt/hr 3PM"
Cells(1, 18).Value = "Consumo Watt/hr 4PM"
Cells(1, 19).Value = "Consumo Watt/hr 5PM"
Cells(1, 20).Value = "Consumo Watt/hr 6PM"
Cells(1, 21).Value = "Consumo Watt/hr 7PM"
Cells(1, 22).Value = "Consumo Watt/hr 9PM"
Cells(1, 23).Value = "Consumo Watt/hr 10PM"
Cells(1, 24).Value = "Consumo Watt/hr 11PM"

z = 0
For j = 2 To 24
    For i = 2 To 71
        Cells(i, j).Value = q(z) 'Insertar los valores resultado en el orden correcto.
        z = z + 1
    Next i
Next j

End Sub

```

calculos_hibridos:

```
Sub calculos_hibridos()

ThisWorkbook.Worksheets("Cubo_PH_hibrido").Activate
ThisWorkbook.Worksheets("Cubo_PH_hibrido").PivotTables("Tabla
dinámical").PivotFields("[Tabla_PH].[Technology].[Technology]").CurrentPageName =
"[Tabla_PH].[Technology].&[OTN]"
'Activar technology.

Dim z As Integer 'Índice
Dim cllc, crlc, t, q As Variant 'Datos

Call vsierte(cllc, crlc, z) 'Datos a las 7 PM.

ThisWorkbook.Worksheets("Cubo_PH_hibrido").Activate
ThisWorkbook.Worksheets("Cubo_PH hibrido").PivotTables("Tabla
dinámical").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName =
"[Tabla_PH].[Hora].[All]"
'Technology a todas las horas

Call extraer_datos(t, z) 'Datos nombres Type.

Call operar(q, cllc, crlc, z) 'Operaciones.

Call introducir_info(q, t, z) 'Introducir en tabla.

End Sub

Sub vsierte(cllc As Variant, crlc As Variant, z As Integer)

ThisWorkbook.Worksheets("Cubo_PH_hibrido").Activate
ThisWorkbook.Worksheets("Cubo_PH_hibrido").PivotTables("Tabla
dinámical").PivotFields("[Tabla_PH].[Hora].[Hora]").CurrentPageName =
"[Tabla_PH].[Hora].&[1899-12-30T19:00:00]"
'Activar 7 PM.

Dim cllc1(10), crlc1(10) As Double

z = 0
For i = 5 To 9
cllc1(z) = Cells(i, 25).Value 'Extraer.
crlc1(z) = Cells(i, 26).Value
z = z + 1
Next i

cllc = cllc1
crlc = crlc1

End Sub

Sub extraer_datos(t As Variant, z As Integer)

Dim t1(10) As String

z = 0
For i = 5 To 9
t1(z) = Cells(i, 1).Value 'Extraer type
z = z + 1
Next i

t = t1

End Sub

Sub operar(q As Variant, cllc As Variant, crlc As Variant, z As Integer)

Dim q1(10) As Double

z = 0
```

```

For i = 5 To 9
    q1(z) = Cells(i, 30).Value + Cells(i, 29).Value + (crlc(z) - Cells(i, 26).Value) * 0.5 +
    (cllc(z) - Cells(i, 25).Value) * 0.5 'Operar
    z = z + 1
Next i

q = q1

End Sub

Sub introducir_info(q As Variant, t As Variant, z As Integer)

ThisWorkbook.Worksheets("Cálculos_híbridos").Activate 'Tabla resultado

Cells(1, 1) = "OTN"                      'Encabezados.
Cells(1, 2) = "Consumo Total"

z = 0
For i = 2 To 6
    Cells(i, 1) = t(z) 'Type.
    Cells(i, 2) = q(z) 'Consumos.
    z = z + 1
Next i

End Sub

```

resultados_LO:

```
Sub resultados_LO()

With Application      'Para optimizar la ejecución del programa. Deshabilitamos eventos,
alerts...

    .Cursor = xlDefault
    .ScreenUpdating = False
    .EnableEvents = False
    .Visible = True
    .DisplayAlerts = False
    .DisplayStatusBar = True

End With

Call barra(0)

Dim lc, piu, p, t, s As Variant 'Variables resultado.
Dim p1(50) As String 'Vector hojas.

p1(0) = "12AM_LO"  'Creamos un vector con el nombre de la hojas las cuales vamos a acceder.
p1(1) = "1AM_LO"
p1(2) = "2AM_LO"
p1(3) = "3AM_LO"
p1(4) = "4AM_LO"
p1(5) = "5AM_LO"
p1(6) = "6AM_LO"
p1(7) = "7AM_LO"
p1(8) = "8AM_LO"
p1(9) = "9AM_LO"
p1(10) = "10AM_LO"
p1(11) = "11AM_LO"
p1(12) = "12PM_LO"
p1(13) = "1PM_LO"
p1(14) = "2PM_LO"
p1(15) = "3PM_LO"
p1(16) = "4PM_LO"
p1(17) = "5PM_LO"
p1(18) = "6PM_LO"
p1(19) = "7PM_LO"
p1(20) = "9PM_LO"
p1(21) = "10PM_LO"
p1(22) = "11PM_LO"
p = p1

Workbooks.Open Filename:="C:\Users\Jesús\Desktop\resultados_detalle.xlsx"
Workbooks("resultados_detalle.xlsx").Activate 'Abrimos y activamos el libro donde se
encuentran las hojas con información.

Call calculos_secundarios_LO(lc, piu, p) 'Cálculos necesarios para otros cálculos.

Call calculos_primarios_LO(q, lc, piu, p) 'Cálculo principal, el del consumo.

Call barra(1)

Call extraer_nombres_LO(t, s, p) 'Nombres de "enmarcado".

Call barra(2)

Call introducir_datos_LO(t, s, q) 'Introducir datos en hoja destino.

Call barra(3)

Workbooks("resultados_detalle.xlsx").Close 'Cerrar el libro donde están las tablas origen.

With Application 'Restablecemos los valores.
```

```

.Cursor = xlDefault
.ScreenUpdating = True
.EnableEvents = True
.DisplayAlerts = True
'.DisplayStatusBar=False

End With

End Sub

Sub calculos_secundarios_L0(lc As Variant, piu As Variant, p As Variant) 'Esta subrutina
sirve para hacer los cálculos secundarios.

'Se extraerán los datos de cada una de las hojas y se calculará para cada hoja.
'Se harán en matrices tantas filas como hojas y columnas como resultados.

Dim z As Integer
Dim slc(50, 100000), spiu(50, 100000) As Integer 'Valores a calcular.

For j = 0 To 22
Workbooks("resultados_detalle.xlsx").Worksheets(p(j)).Activate 'Activar hoja.

z = 3
For i = 3 To 103
slc(j, z) = Cells(i, 13).Value + Cells(i, 15).Value + Cells(i, 17).Value + Cells(i,
19).Value + Cells(i, 21).Value + Cells(i, 23).Value 'Suma de LC.
spiu(j, z) = Cells(i, 14).Value + Cells(i, 16).Value + Cells(i, 18).Value + Cells(i,
20).Value + Cells(i, 22).Value + Cells(i, 24).Value 'Suma de PIU.
z = z + 1
Next i

Next j

lc = slc
piu = spiu

End Sub

Sub calculos_primeros_L0(q As Variant, lc As Variant, piu As Variant, p As Variant)

Dim z As Integer
Dim q1(50, 100000) As Double 'Matriz solución

z = 0

'Operamos cuidando los indices.
For i = 0 To 22
Workbooks("resultados_detalle.xlsx").Worksheets(p(i)).Activate

For j = 3 To 7
q1(i, j) = piu(i, j) * Cells(j, 10).Value + (piu(19, j) - piu(i, j)) * Cells(j, 11).Value
'ESW
Next j
For j = 8 To 19
q1(i, j) = lc(i, j) * Cells(j, 3).Value * Cells(j, 7).Value + (lc(19, j) - lc(i, j)) *
Cells(j, 9).Value 'LC
Next j
For j = 49 To 50
q1(i, j) = piu(i, j) * Cells(j, 10).Value + (piu(19, j) - piu(i, j)) * Cells(j, 11).Value
'LER
Next j
For j = 51 To 61
q1(i, j) = lc(i, j) * Cells(j, 3).Value * Cells(j, 7).Value + (lc(19, j) - lc(i, j)) *
Cells(j, 9).Value 'LC
Next j
For j = 76 To 78
q1(i, j) = piu(i, j) * Cells(j, 10).Value + (piu(19, j) - piu(i, j)) * Cells(j, 11).Value
'LSR

```

```

Next j
For j = 79 To 89
q1(i, j) = lc(i, j) * Cells(j, 3).Value * Cells(j, 7).Value + (lc(19, j) - lc(i, j)) *
Cells(j, 9).Value 'LC
Next j

Next i

q = q1

End Sub

Sub extraer_nombres_LO(t As Variant, s As Variant, p As Variant) 'Esta subrutina sirve
para extraer los datos de enmarcado: filas y columnas.

Workbooks("resultados_detalle.xlsx").Worksheets(p(0)).Activate

Dim z As Integer
Dim t1(500), s1(50) As String

z = 0
For i = 3 To 103
t1(z) = Cells(i, 1).Value 'filas.
z = z + 1
Next i

s1(0) = "12 AM" 'Vector que contiene las horas.
s1(1) = "1 AM"
s1(2) = "2 AM"
s1(3) = "3 AM"
s1(4) = "4 AM"
s1(5) = "5 AM"
s1(6) = "6 AM"
s1(7) = "7 AM"
s1(8) = "8 AM"
s1(9) = "9 AM"
s1(10) = "10 AM"
s1(11) = "11 AM"
s1(12) = "12 PM"
s1(13) = "1 PM"
s1(14) = "2 PM"
s1(15) = "3 PM"
s1(16) = "4 PM"
s1(17) = "5 PM"
s1(18) = "6 PM"
s1(19) = "7 PM"
s1(20) = "8 PM"
s1(21) = "9 PM"
s1(22) = "10 PM"
s1(23) = "11 PM"

t = t1
s = s1

End Sub

Sub introducir_datos_LO(t As Variant, s As Variant, q As Variant) 'Esta subrutina sirve
para introducir todos los datos en la tabla destino.

Dim z As Integer

ThisWorkbook.Worksheets("resultados_LO").Activate 'Activar tabla destino.

Cells(1, 1).Value = "Tipo de disp" 'Identificador elementos.

z = 0
For j = 2 To 24
Cells(1, j).Value = s(z) 'Columnas
z = z + 1
Next j

z = 0

```

```
For i = 3 To 89
Cells(i, 1).Value = t(z) 'Filas
z = z + 1
Next i

z = 0
For j = 2 To 24
For i = 3 To 89
Cells(i, j).Value = q(z, i) 'Resultados, cuidado con los indices.
Next i
z = z + 1
Next j

End Sub
```

resultados_PH:

Código casi idéntico que `resultados_LO`.

Las diferencias más importantes son:

- Los cálculos secundarios.

- Los cálculos primarios.

- Las hojas que vamos recorriendo para ir realizando las operaciones.

barra:

```

Sub barra(x As Integer)

Dim b(4) As String 'Variable que contendrá las barras o los huecos.
Dim p As Variant 'Variable con una barra de muestra.
Dim k As Variant

'Inicialización.
p = "||||||||||||||||||"

b(0) = " "
b(1) = " "
b(2) = " "
b(3) = " "

If x > 3 Then
    MsgBox ("Error en la introducción del número")
Else

    If x < 3 Then      'Mensaje final

        k = "Cargando... | "

    Else

        k = "Fin de carga | "

    End If

    For i = 0 To x

        b(i) = p 'Rellenamos vector tantas veces como la definida en x

    Next i

    Application.StatusBar = k + b(0) + b(1) + b(2) + b(3) + " |"      'Rellenamos la barra
final.

End If

End Sub

```

Códigos DAX

Suma de LC:

```
= [Number of LC]+[Number of LC1]+[Number of LC2]+[Number of LC3]+[Number of LC4]+[Number of LC5]
```

Suma de PIU:

```
= [Number of PIUs]+[Number of PIUs1]+[Number of PIUs2]+[Number of PIUs3]+[Number of PIUs4]+[Number of PIUs5]
```

Suma de LC 7PM:

```
= IF (
    [Tipo de disp] = "ESW1";
    LOOKUPVALUE (
        [Suma de LCs];
        [Tipo de disp]; "ESW1";
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
    IF (
        [Tipo de disp] = "ESW1a";
        LOOKUPVALUE (
            [Suma de LCs];
            [Tipo de disp]; "ESW1a";
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
        IF (
            [Tipo de disp] = "ESW2";
            LOOKUPVALUE (
                [Suma de LCs];
                [Tipo de disp]; "ESW2";
                [Hora]; VALUE ( "30/12/1899 19:00:00" )
            );
            IF (
                [Tipo de disp] = "ESW3";
                LOOKUPVALUE (
                    [Suma de LCs];
                    [Tipo de disp]; "ESW3";
                    [Hora]; VALUE ( "30/12/1899 19:00:00" )
                );
                IF (
                    [Tipo de disp] = "ESW4";
                    LOOKUPVALUE (
                        [Suma de LCs];
                        [Tipo de disp]; "ESW4";
                        [Hora]; VALUE ( "30/12/1899 19:00:00" )
                    );
                    IF (
                        [Tipo de disp] = "LC7";
                        LOOKUPVALUE (
                            [Suma de LCs];
                            [Tipo de disp]; "LC7";
                            [Hora]; VALUE ( "30/12/1899 19:00:00" )
                        );
                        IF (
                            [Tipo de disp] = "LC8";
                            LOOKUPVALUE (
                                [Suma de LCs];
                                [Tipo de disp]; "LC8";
                                [Hora]; VALUE ( "30/12/1899 19:00:00" )
                            );
                            IF (
                                [Tipo de disp] = "LC9";
                                LOOKUPVALUE (
                                    [Suma de LCs];
                                    [Tipo de disp]; "LC9";
                                    [Hora]; VALUE ( "30/12/1899 19:00:00" )
                                );
                                IF (
                                    [Tipo de disp] = "LC10";
                                    LOOKUPVALUE (
```

```

[Suma de LCs];
[Tipo de disp]; "LC10";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC11";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC11";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC12";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC12";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC13";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC13";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC14";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC14";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC15";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC15";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC16";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC16";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC17";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC17";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC18";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC18";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC33";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC33";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC34";
LOOKUPVALUE (
[Suma de LCs];

```

```

[Tipo de disp]; "LC34";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC35";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC35";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC36";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC36";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC37";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC37";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC38";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC38";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC39";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC39";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC40";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC40";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC41";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC41";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC42";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC42";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC43";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC43";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC58";
LOOKUPVALUE (

```

```

[Suma de LCs];
[Tipo de disp]; "LC58";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC59";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC59";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC60";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC60";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC61";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC61";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC62";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC62";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC63";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC63";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC64";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC64";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC65";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC65";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC66";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC66";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC67";
LOOKUPVALUE (
[Suma de LCs];
[Tipo de disp]; "LC67";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Tipo de disp] = "LC68";
LOOKUPVALUE (
[Suma de LCs];

```


Suma de PIU 7PM:

Código prácticamente idéntico a Suma de LC 7PM, solo cambia uno de los campos.
Ejemplo:

```
= IF (
[Tipo de disp] = "ESW1";
LOOKUPVALUE (
[Suma de PIUs];
[Tipo de disp]; "ESW1";
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);
```

Consumo Watt/hr_LO:

```

=IF (
    [Tipo de disp] = "ESW1"
    || [Tipo de disp] = "ESW1a"
    || [Tipo de disp] = "ESW2"
    || [Tipo de disp] = "ESW3"
    || [Tipo de disp] = "ESW4";
    [Suma de PIUs]
        * [Consumo Unitario PIU Watt]
        + ( [Suma de PIUs 7PM] - [Suma de PIUs] )
        * [Consumo PIU StandBy];
    IF (
        [Tipo de disp] = "LC7"
        || [Tipo de disp] = "LC8"
        || [Tipo de disp] = "LC9"
        || [Tipo de disp] = "LC10"
        || [Tipo de disp] = "LC11"
        || [Tipo de disp] = "LC12"
        || [Tipo de disp] = "LC13"
        || [Tipo de disp] = "LC14"
        || [Tipo de disp] = "LC15"
        || [Tipo de disp] = "LC16"
        || [Tipo de disp] = "LC17"
        || [Tipo de disp] = "LC18"
        || [Tipo de disp] = "LC33"
        || [Tipo de disp] = "LC34"
        || [Tipo de disp] = "LC35"
        || [Tipo de disp] = "LC36"
        || [Tipo de disp] = "LC37"
        || [Tipo de disp] = "LC38"
        || [Tipo de disp] = "LC39"
        || [Tipo de disp] = "LC40"
        || [Tipo de disp] = "LC41"
        || [Tipo de disp] = "LC42"
        || [Tipo de disp] = "LC43"
        || [Tipo de disp] = "LC58"
        || [Tipo de disp] = "LC59"
        || [Tipo de disp] = "LC60"
        || [Tipo de disp] = "LC61"
        || [Tipo de disp] = "LC62"
        || [Tipo de disp] = "LC63"
        || [Tipo de disp] = "LC64"
        || [Tipo de disp] = "LC65"
        || [Tipo de disp] = "LC66"
        || [Tipo de disp] = "LC67"
        || [Tipo de disp] = "LC68";
        [# Ports per LC] * [Activo]
            * [Suma de LCs]
            + ( [Suma de LC 7PM] - [Suma de LCs] )
            * [StandBy];
        IF (
            [Tipo de disp] = "LER1"
            || [Tipo de disp] = "LER2"
            || [Tipo de disp] = "LSR1"
            || [Tipo de disp] = "LSR2"
            || [Tipo de disp] = "LSR3";
            [Suma de PIUs]
                * [Consumo Unitario PIU Watt]
                + ( [Suma de PIUs 7PM] - [Suma de PIUs] )
                * [Consumo PIU StandBy];
            0
        )
    )
)

```

Suma de L LC:

= [Number of L LC]+[Number of L LC1]+[Number of L LC2]+[Number of L LC3]+[Number of L LC4]

Suma de R LC:

= [Number of R LC]+[Number of R LC1]+[Number of R LC2]+[Number of R LC3]+[Number of R LC4]

Suma de PIUs:

= [Number of PIUs]+[Number of PIUs1]+[Number of PIUs2]+[Number of PIUs3]+[Number of PIUs4]

Suma de repeaters:

= [Number of repeaters]+[Number of repeaters1]+[Number of repeaters2]+[Number of repeaters3]+[Number of repeaters4]

Consumo L LC:

=IF([Conf_Index]=70||[Conf_Index]=71||[Conf_Index]=72||[Conf_Index]=73||[Conf_Index]=74||[Conf_Index]=75;([Suma de L LC]*[Consumo por tarjeta]);[Nº of L port s]*[Suma de L LC]*POWER([BW L LC Mbps],2/3))

Consumo PIU:

= [Suma de R LC]*1000

Suma de L LC 7PM:

```
= IF (
[Conf_Index] = BLANK();
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; BLANK();
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("1");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("1");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("2");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("2");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("3");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("3");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("4");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("4");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("5");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("5");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("6");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("6");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("7");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("7");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("8");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("8");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("9");
    LOOKUPVALUE (
```

```

[Suma de L LC];
[Conf_Index]; VALUE("9");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("10");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("10");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("11");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("11");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("12");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("12");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("13");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("13");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("14");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("15");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("16");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("16");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("17");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("17");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("18");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("LC35");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("19");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("19");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("20");
LOOKUPVALUE (
[Suma de L LC];

```

```

[Conf_Index]; VALUE("20");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("21");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("21");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("22");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("22");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("23");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("23");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("24");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("24");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("25");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("25");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("26");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("26");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("27");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("27");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("28");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("28");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("29");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("29");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("30");

```

```

LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("30");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("31");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("31");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("32");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("32");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("33");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("33");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("34");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("34");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("35");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("35");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("36");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("36");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("37");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("37");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("38");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("38");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("39");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("39");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("40");

```

```

LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("40");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("41");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("41");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("42");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("42");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("43");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("43");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("44");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("44");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("45");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("45");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("46");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("46");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("47");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("47");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("48");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("48");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("49");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("49");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("50");
    LOOKUPVALUE (

```

```

[Suma de L LC];
[Conf_Index]; VALUE("50");
[Hora]; VALUE ( "30/12/1899 19:00:00" )

IF (
[Conf_Index] = VALUE("51");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("51");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("52");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("52");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("53");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("53");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("54");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("54");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("55");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("55");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("56");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("56");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("57");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("57");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("58");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("58");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("59");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("59");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("60");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("60");

```

```

[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("61");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("61");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("62");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("62");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("63");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("63");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("64");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("64");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("65");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("65");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("66");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("66");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("67");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("67");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("68");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("68");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("69");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("69");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("70");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("70");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

```

```

);
IF (
[Conf_Index] = VALUE("71");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("71");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("72");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("72");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("73");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("73");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("74");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("74");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("75");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("75");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("76");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("76");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("77");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("77");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("78");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("78");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("79");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("79");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("80");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("80");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

```

```

IF (
[Conf_Index] = VALUE("81");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("81");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("82");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("82");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("83");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("83");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("84");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("84");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("85");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("85");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("86");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("86");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("87");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("87");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("88");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("88");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("89");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("89");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("90");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("90");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (

```

```

[Conf_Index] = VALUE("91");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("91");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("92");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("92");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("93");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("93");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("94");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("94");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("95");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("95");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("96");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("96");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("97");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("97");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("98");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("98");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("99");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("99");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("100");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("100");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("101");

```

```

LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("101");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("102");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("102");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("103");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("103");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("104");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("104");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("105");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("105");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("106");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("106");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("107");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("107");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("108");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("108");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("109");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("109");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("110");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("110");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("111");
    LOOKUPVALUE (

```

```

[Suma de L LC];
[Conf_Index]; VALUE("111");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("112");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("112");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("113");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("113");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("114");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("114");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("115");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("115");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("116");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("116");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("117");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("117");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("118");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("118");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("119");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("119");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("120");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("120");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("121");
LOOKUPVALUE (
[Suma de L LC];

```

```

[Conf_Index]; VALUE("121");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("122");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("122");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("123");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("123");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("124");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("124");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("125");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("125");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("126");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("126");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("127");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("127");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("128");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("128");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("129");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("129");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("130");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("130");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("131");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("131"));

```

```

[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("132");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("132");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("133");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("133");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("134");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("134");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("135");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("135");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("136");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("136");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("137");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("137");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("138");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("138");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("139");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("139");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("140");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("140");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("141");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("141");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

```

```

);
IF (
[Conf_Index] = VALUE("142");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("142");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("143");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("143");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("144");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("144");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("145");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("145");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("146");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("146");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("147");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("147");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("148");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("148");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("149");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("149");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("150");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("150");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("151");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("151");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

```

```

IF (
[Conf_Index] = VALUE("152");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("152");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("153");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("153");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("154");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("154");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("155");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("155");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("156");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("156");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("157");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("157");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("158");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("158");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("159");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("159");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("160");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("160");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("161");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("161");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (

```

```

[Conf_Index] = VALUE("162");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("162");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("163");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("163");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("164");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("164");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("165");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("165");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("166");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("166");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("167");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("167");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("168");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("168");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("169");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("169");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("170");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("170");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("171");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("171");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

IF (
[Conf_Index] = VALUE("172");

```

```

        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("172");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("173");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("173");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("174");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("174");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("175");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("175");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("176");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("176");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("177");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("177");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("178");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("178");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("179");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("179");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("180");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("180");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("181");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("181");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    IF (
        [Conf_Index] = VALUE("182");
        LOOKUPVALUE (

```

```

[Suma de L LC];
[Conf_Index]; VALUE("182");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("183");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("183");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("184");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("184");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("185");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("185");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("186");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("186");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("187");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("187");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("188");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("188");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("189");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("189");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("190");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("190");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("191");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("191");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("192");
LOOKUPVALUE (

```

```

[Suma de L LC];
[Conf_Index]; VALUE("192");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("193");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("193");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("194");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("194");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("195");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("195");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("196");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("196");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("197");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("197");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("198");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("198");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("199");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("199");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("200");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("200");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("201");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("201");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("202");
LOOKUPVALUE (
[Suma de L LC];

```

```

[Conf_Index]; VALUE("202");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("203");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("203");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("204");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("204");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("205");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("205");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("206");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("206");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("207");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("207");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("208");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("208");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("209");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("209");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("210");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("210");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("211");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("211");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("212");
LOOKUPVALUE (
[Suma de L LC];

```

```

[Conf_Index]; VALUE("212");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("213");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("213");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("214");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("214");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("215");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("215");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("216");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("216");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("217");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("217");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("218");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("218");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("219");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("219");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("220");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("220");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("221");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("221");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("222");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("222"));

```

```

[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("223");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("223");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("224");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("224");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("225");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("225");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("226");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("226");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("227");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("227");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("228");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("228");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("229");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("229");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("230");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("230");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("231");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("231");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

IF (
[Conf_Index] = VALUE("232");
LOOKUPVALUE (
[Suma de L LC];
[Conf_Index]; VALUE("232");
[Hora]; VALUE ( "30/12/1899 19:00:00" )
);

```

```

    );
    IF (
        [Conf_Index] = VALUE("233");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("233");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("234");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("234");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("235");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("235");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("236");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("236");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("237");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("237");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("238");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("238");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("239");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("239");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("240");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("240");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("241");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("241");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

    IF (
        [Conf_Index] = VALUE("242");
        LOOKUPVALUE (
            [Suma de L LC];
            [Conf_Index]; VALUE("242");
            [Hora]; VALUE ( "30/12/1899 19:00:00" )
        );
    );

```

```

        );
IF (
[Conf_Index] = VALUE("243");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("243");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("244");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("244");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("245");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("245");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("246");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("246");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("247");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("247");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("248");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("248");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("249");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("249");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("250");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("250");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("251");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("251");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );
IF (
[Conf_Index] = VALUE("252");
    LOOKUPVALUE (
        [Suma de L LC];
        [Conf_Index]; VALUE("252");
        [Hora]; VALUE ( "30/12/1899 19:00:00" )
    );

```

```

);
IF (
[Conf_Index] = VALUE("253");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("253");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("254");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("254");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("255");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("255");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("256");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("256");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("257");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("257");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("258");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("258");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("259");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("259");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("260");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("260");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("261");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("261");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
IF (
[Conf_Index] = VALUE("262");
LOOKUPVALUE (
    [Suma de L LC];
    [Conf_Index]; VALUE("262");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);

```


Suma de R LC 7PM:

Código prácticamente idéntico a Suma de L LC excepto en uno de sus campos.

Ejemplo:

```
IF (
[Conf_Index] = VALUE("1");
LOOKUPVALUE (
    [Suma de R LC];
    [Conf_Index]; VALUE("1");
    [Hora]; VALUE ( "30/12/1899 19:00:00" )
);
```

Consumo Watt/hr_PH:

```
= [Consumo L LC] + [Consumo PIU] + ( [Suma de R LC 7PM] - [Suma de R LC] ) * 0,5 + ([Suma de L LC 7PM] - [Suma de L LC]) * 0,5
```