

El Problema del Año 2000

Un Recuento Final

"Why should it?" muttered the Hatter "Does your watch tell you what year it is?"

"Of course not," Alice replied very readily: "but that's because it stays the same year for such a long time together"
Lewis Carroll, Alice in Wonderland.

M. en C. Eduardo René Rodríguez Ávila
Maestría en Informática.
SEPI-UPIICSA
e.r.rodriguez@ieee.org

Introducción.

El "problema del año 2000" es la designación a la necesidad de ajuste de todos los mecanismos con funcionamiento basados en tiempo y que sólo utilizan los dos últimos dígitos menos significativos del año para sus procesos de cálculo o registro. La implicaciones de este problema no sólo atañen al software, como la mayoría del público piensa por la enorme publicidad al respecto, sino que comprende un conjunto más amplio de elementos tanto lógicos como físicos. Los elementos lógicos comprenden desde los procedimientos de trabajo apoyados en formas impresas y el registro de fechas hasta las aplicaciones automatizadas cuyos algoritmos de funcionamiento están basados en la aritmética de fechas y tiempo. Los elementos físicos abarcan dispositivos que van desde lo completamente mecánico, hasta llegar a lo completamente electrónico, con lo electromecánico en un punto intermedio.

Este problema en particular ha llamado mucho la atención por el enorme esfuerzo que implica evitar problemas por algo que parece trivial hoy, pero que hace 20 años resultaba necesario, inofensivo y un gran acierto. El costo en esta empresa ha sorprendido a más de uno y maravillado a muchos más por la serie de tareas que hay que coordinar. Sin embargo, la historia tiene un lado negativo, ya que la publicidad entorno a este problema ha degenerado en mucho casos en una serie de esfuerzos inútiles y superfluos donde el objetivo se ha visto reemplazado por la organización de actividades mas que por la conversión en sí.

¿Qué es lo que pasará con la llegada del año 2000? Existen opiniones encontradas. Hay quienes auguran un caos total y están los que consideran que todo esto no ha sido más que una enorme treta y pérdida de tiempo explotada en mil y una formas para el beneficio de algunos y gasto de otros.

A sólo unos meses de llegar el año 2000 se presenta un recuento de los esfuerzos realizados en la búsqueda de los esfuerzos planteados. Este trabajo de investigación expone la naturaleza del problema, las soluciones encontradas, las implicaciones del proceso y, finalmente, concluye con una predicción de lo que posiblemente ocurrirá desde las primeras horas de la mañana del 31 de diciembre de 1999 hasta las últimas horas de la noche del 3 de enero de 2000.

Los antecedentes.

Los computadores de primera generación (1938-1953) eran herramientas de cálculo de 1 ó 2 Kbytes. El ingreso del computador al plano comercial no fue hasta los equipos de segunda generación (1952-1963) con apenas 8KB o menos, pero el éxito y difusión del computador para aplicaciones comerciales fue marcado por los equipos de tercera generación (1962-1975) que era vendido regularmente con 64 ó 128 Kbytes. Con tales cantidades de memoria es fácil entender que el espacio en memoria era preciado. Ahorrar la mayor cantidad posible era una prioridad y eliminar datos innecesarios una necesidad. Ya sea por esto, por limitantes en el manejo de las longitudes de registros en los primeros sistemas de archivo o también por conservar espacio en cinta o disco, el almacenamiento de fechas en almacenamiento secundario fue objeto del mismo ahorro.

Uno de estos datos innecesarios era el siglo (los dos primeros dígitos del año). Encontrándose en la mitad del siglo XX, la mayoría de los cálculos relacionados con fechas para procesos cotidianos involucraba fechas del mismo siglo, de aquí que este dato no fuera necesario conservarlo. Conforme la tecnología evolucionaba, nuevos computadores de mayor capacidad aparecían. Poco después las primeras aplicaciones de software de gran escala comenzaron a ser modeladas. La automatización de procesos por medios informáticos dejó de verse como el desarrollo de programas de cómputo aislados para pasar a ser visto como el desarrollo de sistemas de información. Para entonces ya era demasiado tarde, ya existían procesos y datos mantenidos en equipo de cómputo con fechas representadas por dos dígitos, y una regla no escrita ya era

conocida por los programadores: “en fechas sólo conserva el año”. El desarrollo de nuevos sistemas de información heredó datos y reglas de desarrollo.

Ciertamente no se esperaba que muchos de los sistemas estuvieran en funcionamiento para las fechas actuales, después 10, 20 ó hasta 30 años. Pero tampoco se consideró que la compatibilidad con datos históricos y programas anteriores llevara consigo una bomba de tiempo.

Por supuesto, no debe entenderse que el manejo de años en computadores estaba limitado a dos dígitos. Algoritmos de calendarios perpetuos y aplicaciones especiales destinadas al almacenamiento de datos históricos o efemérides estaban obligadas a manejar años con cuatro dígitos o más. El problema en cuestión se debe a una convención y necesidad del momento, fomentada por algunos factores ambientales (recordemos que la mayoría de los formatos de fecha usados por el ser humano desde hace siglos contempla sólo los años del siglo o la posición del año a manera de los dígitos menos significativos).

Este problema también ha sido denominado como *El Bug del Milenio*, por ya estar acostumbrados a experimentar defectos (errores de programación o diseño) en el software y que son conocidos en el medio informático como *bugs*. Es importante recalcar que el problema no se debe a la llegada del nuevo milenio. El año 2000 es el último año del siglo XX; la llegada del nuevo milenio (el siglo XXI) será hasta el 1º de enero de 2001. Si el término “milenio” ha sido usado este debe ser interpretado como relativo al siglo XX.

Finalmente, resulta asombroso que exista un precedente al problema y que éste no se haya presentado con un cambio de siglo sino de década; más asombroso es que haya sido en plena era moderna y paradójicamente que se haya dado en el inicio de la Era de la Información. El popular equipo IBM 360 no podía manejar fechas más allá del 31 de diciembre de 1969. Pocos se dieron cuenta de esto, hasta que por todo el mundo los sistemas comenzaron a caer conforme se aproximaban al límite en sus cálculos. IBM corrigió el problema con un parche de "largo plazo". Nadie imaginaba lo que vendría después.

Notificación del problema: psicosis y mercadotecnia.

Nuestro manejo y representación de fechas nos ha traído dolores de cabeza. Hoy la programación y funcionamiento de todo dispositivo, proceso y mecanismo basado en tiempo debe ser revisado; y si tomamos en cuenta que el resolver los problemas diarios de producción y operación es de por sí el reto de cada día, veremos que el arreglar todo lo necesario para evitar problemas con el manejo de fechas, sin tomar en cuenta la necesidad de cambio e innovación, es una tarea descomunal.

Los primeros efectos del problema comenzaron a hacerse sentir desde el comienzo de la presente década al entrar al panorama de acción el año 2000 dentro de un periodo de 10 años que es importante en muchos procesos legales y administrativos. Era de esperarse que los problemas aumentarían conforme se acortaría el tiempo para la llegada para el año 2000, por lo que las primeras señales de alerta se dieron en la primera mitad de la década y los primeros trabajos de conversión iniciaron al comienzo de la segunda mitad.

Tal vez hasta ahí debió quedar el asunto, pero la mercadotecnia y los medios entraron en escena. El problema fue sobre analizado y las consecuencias exageradas. Comenzó a haber pánico y desconfianza. Unos a otros comenzaron a vigilarse para ver si estaban haciendo algo. Comenzaron a aparecer certificaciones, los productos comenzaron a mostrar etiquetas de estar a prueba de problemas con la llegada del último año del siglo, se crearon organismos, áreas y departamentos de control de calidad en la transformación al año 2000. Una enorme cantidad de información comenzó a fluir a través de Internet, correo y faxes para saber y mostrar quien estaba haciendo qué y cómo.

Cientos de empresas comenzaron a brindar asesoría sobre el tema. Artículos, conferencias y proyectos fueron escritos, impartidos y establecidos. Se desenterraron manuales, los programadores con dominio en COBOL, PL/I, Fortran y algunas “lenguas muertas” fueron llamados a líneas (con mejores salarios que en sus respectivas épocas). Empresas se creaban y muchas otras se especializaban sólo en el problema del año 2000.

Los comentarios no se hicieron esperar. En un extremo teníamos a los analistas y expertos que advertían sobre un desastre y las empresas que exageraban sus acciones. Por ejemplo, British Airways declaró que tendrá equipos y tripulaciones en tierra durante el primer día del año 2000 porque no tiene garantías de que todo funcionaría como debe. El temor aumentado propició que proveedores y consumidores preguntaran continuamente que es lo que la gente y empresas estaban haciendo para salvarse de un armagedón informático, y esto comenzaron a hacerlo de manera sistemática a través de encuestas, entrevistas y cuestionarios agregando una variable de costo más a la titánica empresa. Ahora no sólo debe llevarse a cabo la transformación y responder a los problemas de operación sino además responder este tipo de sondeos con

el fin de mantener contratos, clientes y participación en el mercado y que llevó en muchos casos a la creación de equipos dedicados ex profeso. Al final, la aparición de los organismos oficiales no se hizo esperar. Las grandes compañías y gobiernos han iniciado toda una industria del Y2K-compliant (léase *certificación*).

El otro extremo es el de la opinión de muchos otros analistas y expertos que consideran esto no es más que una broma y, aunque es de esperarse algún problema, el fin de la civilización como la conocemos está muy distante por el mero manejo de fechas. No habrá fallas masivas de servicios, nadie se quedará sin luz, los elevadores continuarán funcionando y sólo un porcentaje muy reducido de los sistemas de cómputo podría verse afectado en su funcionamiento.

Muy cercano a este punto, y caso de estudio, son los países asiáticos (principalmente China y Japón) quienes han sido severamente criticados por su aparente impasividad al problema. Claro que si consideramos que en realidad estas naciones se rigen por otros sistemas de fechas podremos ver el porqué no se ven tan alarmados por el calendario de occidente. Para mejorar las relaciones públicas se han tomado algunas acciones, algunas algo severas. Por ejemplo, el gobierno chino declaró que obligará volar la noche del 31 de diciembre de 1999 a todos sus ejecutivos que no trabajen sobre la corrección del problema.

La definición del problema.

Brevemente, este problema abarca lo siguiente:

- La representación del año usando sólo dos dígitos provoca que al manipular fechas se presenten fallas aritméticas, de comparación, de ordenación, y de entrada y salida a bases de datos y archivos. Erróneamente se tomará el valor 00 menor al 99 con lo que criterios de ordenación y comparación harán que datos relacionados a las fechas del 2000 sean tomadas como previas a los de 1900. Los cálculos aritméticos darán resultados negativos.
- Uso de algoritmos incorrectos que consideran años bisiestos al ser divisibles por 400.
- El manejar valores constantes empotrados en el código relacionados con fechas, como los dígitos "19", y el uso de valores especiales (llamados "números mágicos") como el "00" o "99".
- Problemas relacionados a desbordamiento en variables y registros destinados a fechas.

Dentro de las particularidades de este problema, que lo distinguen de cualquier otro de mantenimiento de software podemos citar

- Posee una fecha límite inamovible y común sin importar la naturaleza del sistema o giro de la empresa.
- En mucho casos el mantenimiento deberá ser a elementos obsoletos con tecnología y conocimientos en desuso.
- Se conoce la existencia del problema pero no su lugar.

Las soluciones.

Ante todo, lo primero era adoptar un standard en la representación de las fechas y para el cual ya existía uno: el formato ISO 8601. Bajo esta especificación la fecha es expresada en el formato CCYYMMDD, donde CC corresponden a la centuria, YY al año en esa centuria, MM y DD son el mes y día del año respectivamente. El formato no sólo soluciona el problema de incluir el siglo en la fecha sino que también ordena los elementos de la fecha de manera que la centuria y el año sean los dígitos más significativos y los días los menos significativos, con lo que se consigue efectuar comparaciones de orden directas.

Los problemas relacionados con números mágicos, años bisiestos y desbordamiento de registros o variables puede ser corregido de manera limpia y directa. Al final es sólo un asunto de corrección de algoritmos y redefinición de las estructuras de datos.

Los problemas relacionados con fechas son otra cosa e involucran el estudio del código y propósito de la aplicación para determinar el método más adecuado a seguir. Las posibilidades de solución adoptadas han sido:

Expansión.

Esta solución implica el cambio del formato de los dos dígitos usados para el año a cuatro dígitos (YY □ YYYY), incluyendo todo aquello a lo que se haga referencia: estructuras de datos, variables, registros en archivos (actuales e históricos), bases de datos, variables de ambiente del sistema operativo, etcétera. La solución no sólo es la más completa y la mejor, sino que asegura que sus aplicaciones funcionarán adecuadamente para los próximos 2,600 años.

Codificación.

La información del siglo se codifica en un espacio de seis dígitos, que puede ser hecho de cinco formas distintas:

- Codificación de fechas Julianas completas en un campo binario de 48 bits, i.e. usando 6 bytes (YYMMDD ☐ BBBB).
- Codificación del campo de fecha como un desplazamiento (offset) o tiempo transcurrido en días desde una fecha específica, e.g. desde el 1o de enero de 1900 y que dicho sea de paso funciona también para los próximos 2,600 años (YYMMDD ☐ DDDDD).
- Codificación del siglo con un dígito (1 para 1900, 2 para 2000 y así sucesivamente) y reemplazo del mes y el día con el día del año (YYMMDD ☐ CYYDDD).
- Codificación del siglo con un dígito y expresión del mes en base 12 (0-9, A-B representado como M', de tal manera que YYMMDD ☐ CYYM'DD).
- Codificación del año completo usando dos bytes; que permite representar 65,536 años en un rango de -32767 a 32768 o de 0 a 65,535 (YYMMDD ☐ BBMMDD).

Desplazamiento.

Si el problema sólo consistía en la aritmética podía recurrirse a algunos trucos con números. Por ejemplo, mientras que la diferencia 99-93=6 es correcta, 00-93=-93 no lo es en términos de años entre las dos fechas. Sin embargo, $(00+7)-(93+7)=07-00=7$ nos da el resultado esperado (i.e. aritmética en módulo 100). Por supuesto la constante a usar debe ser correctamente elegida si se desea tener disponibles y válidas otras propiedades de la fecha resultante del cálculo como el día de la semana. Esencialmente hay dos ciclos que determinan los días de la semana: cuatro semanas por mes y siete días en una semana. El producto del periodo de estos dos ciclos, $4 \times 7 = 28$, nos da una constante que posee las propiedades de coincidencia en días de la semana. De aquí que usar 28 para éste método sea mucho más lógico.

Ventanas de tiempo.

Esta solución sólo funciona si la aplicación se mantiene dentro de un período de 100 años. La ventaja es que no requiere cambio en el formato de datos, sólo en los programas.

Puentes.

Para los casos en los que no se tiene el código fuente de la aplicación, por lo que no es posible efectuar modificaciones, sólo se cuenta con la alternativa de usar puentes o programas de interfaz. Estos programas son externos a la aplicación o sistema y actúan de intermediarios con el entorno operacional. Pueden ser implementados con el método de ventanas de tiempo, o modificando el reloj del sistema y sumando o restando 28 años a la fecha de entrada o salida del programa.

Substitución o retiro.

En muchos casos la mejor alternativa fue el de "borrón y cuenta nueva". La substitución de sistemas o equipo (el definitivo apagado o eliminación) fue muy bien vista si se debía a razones organizacionales (renovación de equipo, fusiones, ventas, cambios de domicilio, etcétera).

Abstención.

Una solución después de todo. Consiste en hacer absolutamente nada. No se trataba de ignorar el problema sino que después de determinar los riesgos, costos y alternativas contra el hacer nada se veía que era mejor dejar que el mundo siguiera su curso natural, especialmente si no se tenían sistemas dependientes del tiempo.

Como puede verse el problema con el manejo de fechas fue el meollo del asunto, de lo que se desprende que ciertas fechas fueron objeto de especial atención y parámetros de prueba para las correcciones efectuadas; y de las que podían encontrarse ciertas dependencias. La siguiente tabla enumera y describe estas fechas.

Fecha	Observación
1998-01-01	Inicio de año en el que pudieron comenzarse trabajos de ajuste y conversión.
1999-01-01	Inicio de año en el que pudieron comenzarse trabajos de ajuste y conversión.
1999-09-09	Presencia de números mágicos.
1999-10-10	Primer fecha que ocuparía los 8 dígitos y para la que ya habría adecuaciones en producción.
1999-12-31	El día previo al desastre.
2000-01-01	El día del desastre.
2000-01-03	Primer día hábil en la era del desastre.

2000-02-29	Día que hace bisiesto al 2000.
2000-01-10	Primer fecha de nueve caracteres en la era del desbordamiento de los años de dos dígitos.
2000-10-10	Primer fecha de diez caracteres en la era del desbordamiento de los años de dos dígitos.
2000-12-31	Día 366 del año del armagedón informático.
2001-01-01	Inicio del siglo XXI.
2100-01-01	Año no bisiesto.
3000-01-01	Año no bisiesto.

Tabla 1.- Fechas de prueba.

El costo.

Desde su inicio la estimación del costo de los trabajos a realizar ha resultado difícil. Primero porque se desconocía la ubicación y alcance del problema. Segundo, porque no se contaba con las implicaciones mercadotécnicas y organizacionales a su alrededor.

Mitre Corporation elaboró un estudio detallado de los sistemas del US Department of Defense en 1995 examinando 5.4 millones de líneas de código de nueve estaciones de comando y control y de dos sistemas de información. El estudio reveló que solamente 1.16% (62, 200 líneas de código) involucraban manipulaciones de fechas y éste código estaba contenido en sólo el 10.03% de los sistemas. El estudio de Mitre estimaba que la corrección de los sistemas afectados costaría entre \$0.75 USD y \$1.79 USD por línea en los sistemas de información y de \$1.00 USD a \$8.52 USD en los sistemas de control y comando.

En 1997 Gartner Group estimaba que la conversión de los sistemas le costaría al US Department of Defense más de \$30 mil millones USD y que el impacto a nivel mundial sería de \$600 mil millones USD, únicamente en lo que a software se refiere. Tres estos estudios aparecieron mucho otros similares indicando que entre un 10% y 15 % del código en sistemas de información resultaba afectado.

En México, los grandes bancos como Banamex, Bancomer y Serfin reportaban un aproximado de 30 millones de líneas de código. Por lo que podemos darnos una muy buena idea de lo que este problema implicaba para el entorno nacional.

Para finales de 1998 Gartner Group realizó un sondeo a nivel mundial (87 países) con los siguientes resultados

- 5% de los presupuestos para tecnologías de información fue dedicado a conversiones para el año 2000 en 1997, estimando se incrementara en un 30% para 1998 y 44% para 1999.
- Las compañías de mediano tamaño gastarían entre \$3.6 y \$4.2 millones USD en costos de conversión (a un costo de \$450 a \$600 USD por programa, \$1 a \$1.5 USD por línea de código).
- La Asociación de América para Tecnologías de Información (ITAA Information Technology Association of America) estimaba que el costo de corrección sería de entre \$100 y 200 mil millones USD sólo en los Estados Unidos de América, siendo de \$300 a \$600 mil millones USD en todo el mundo.

Conclusiones.

Finalmente, ¿qué es lo que pasará? Bueno, llegará el año 2000 y continuaremos desarrollando problemas que resolver para el año 2100 ó 3000. Todos los problemas que se esperaría ocurrieran un poco antes y a partir del 1 de enero del 2000 ya se habrán presentado, habrán sido anticipados o habrán sido corregidos desde inicios de la década de 1990.

Lo que si es de esperarse es que llegue el 3 de enero del 2000 y nos encontremos con empresas, organismos, áreas, departamentos y personal que nos sobran. Veremos que el costo de la solución ha sobrepasado por mucho las expectativas iniciales (principalmente porque se invirtió más en gente, papeles y publicidad que en la solución al problema y por la competencia por ver quien se había dado cuenta primero del problema o quien tenía la solución definitiva). No será sorprendente que el costo total de este asunto sobrepase el billón de dólares.

La nota final será que, del llamado armagedón informático tan anunciado, sólo se presentarán algunos incidentes. Y, es casi seguro, en la mayoría de éstos, un despreocupado e ingenuo operador solucionará el problema ajustando el reloj de la máquina o cambiando una variable de ambiente a algún año de 1900.

Bibliografía.

1. Dealing with dates: Solutions for the Year 2000, Robert A. Martin, *Computer*, IEEE, USA; March 1997, pp. 44-51.
2. Beyond Y2K, Philip J. Gill, *Oracle Magazine*, Oracle Corp., USA; January/February 1999, pp. 41-67.
3. Data elements and interchange Formats - Information interchange - Representation of dates and times, International Standard ISO 8601, International Organization for Standardization; Geneva, Switzerland; First Edition, 1988-06-15. URL: <http://www.iso.ch/market/8601.pdf>
4. Encapsulation Solutions for Year 2000 Compliance: A Summary, Don Estes, 2000 Technologies Corporation; USA, 1997. URL: <http://www.stsc.hill.af.mil/CrossTalk/1998/jan/y2kencapsulation.html>

Anexo 1

Información en la WWW.

Noticias, organismos, recursos y opiniones.

Bolsa Mexicana de Valores, URL: <http://www.bmv.com.mx/>
Información del año 2000, URL: <http://www.bmv.com.mx/html/2000.html>
Comisión Nacional Bancaria y de Valores, URL: <http://www.cnbv.gob.mx/>
Banco de México, URL: <http://www.banxico.org.mx/>
Instituto Nacional de Estadística Geografía e Informática, URL: <http://www.inegi.gob.mx/>
Comisión Nacional para la Conversión Informática Año 2000, URL: <http://www.y2k.gob.mx/>
Comisión Federal de Electricidad. Proyecto Año 2000, URL: <http://www.cfe.gob.mx/a2000/>
Asociación Mexicana de la Industria de Tecnología de Información (AMITI), URL: <http://www.amiti2000.org.mx/>
Secretaría de Contraloría y Desarrollo Administrativo, URL: <http://www.secodam.gob.mx/proy2000/>
Principia, URL: <http://spin.com.mx/~eravila>
ISO Online - International Organization for Standardization, URL: <http://www.iso.ch/>
MITRE/ESC Year 2000 Homepage, URL: <http://www.mitre.org/technology/y2k/>
U.S. Federal Government CIO Council Committee on Year 2000, URL:
<http://www.itpolicy.gsa.gov/mks/yr2000/y2khome.htm>
The Year 2000 Information Center, URL: <http://www.year2000.com/>
IBM Year 2000 Home Page, URL: <http://www.ibm.com/IBM/year2000/>
The Year 2000 and 2-Digits Dates: A Guide for Planning and Implementation, URL:
<http://www.s390.ibm.com/stories/tran2000.html>
Gartner Group, URL: <http://gartner5.gartnerweb.com/public/static/solutionseries/year2000/>
Netscape, URL: <http://www.netscape.com/y2k/index.html?cp=hom01cny2>
Gary North's Y2K Links and Forums, URL: <http://www.garynorth.com/>
Oracle, URL: <http://www.oracle.com/year2000/>
CNET News.com - Year 2000, URL: <http://www.news.com/Categories/Index/0,3,87,00.html?st.ne.nav..y2kidx>
Year 2000 National Education Taskforce, URL: <http://www.y2knet.com/>
Y2K News Magazine, URL: <http://www.y2knews.com/>
Information Technology Association of America - Year 2000 website, URL: <http://www.itaa.org/year2000/index.htm>
U.S. Federal Communications Commission Year 2000 Home Page, URL: <http://www.fcc.gov/year2000/>
U.S. Federal Reserve Board Y2K Page, URL: <http://www.bog.frb.fed.us/y2k/>
Year 2000 Liability, URL: <http://www.2000law.com/>
The National Bulletin Board for Year 2000, URL: <http://www.it2000.com/>

Humor y no creyentes.

Humor at The Year 2000 Information Center, URL: <http://www.year2000.com/y2khumor.html>
Year 2000 Computer Bug Hoax, URL: <http://www.angelfire.com/oh/justanumber/>
Woohoo! It's the Y2K Fun Page URL: <http://www.leonardsloan.com/about/y2k/index.htm>