

Critical Date Testing – Leap Day and More

May 2012

As we described in [It's Official! Leap Day Caused the Windows Azure Outage](#), our companion article in this issue, Microsoft's Vice President of Windows Servers and Solutions described in a very transparent blog why the Windows Azure Cloud went down for a day and a half at the stroke of midnight GMT, February 28, 2012. It was, in fact, a Leap Day bug.

When a new virtual machine (VM) is created in Azure, the VM generates a security certificate that expires in one year. The faulty program accomplished this simply by adding one to the year. Thus, certificates created on February 29, 2012, expired on February 29, 2013 – an invalid date. This caused physical servers to be declared unhealthy, and all of their active VMs were moved to other physical servers. None of these VMs could now come up, and the failure of otherwise healthy servers cascaded through most of the cloud.

After the immense and largely successful effort expended to protect against the Y2K problem, one would think that date/time bugs would be a thing of the past. Evidently, not so. Azure was not the only fatality of Leap Day. Systems around the world felt its impact:

- A Leap Day bug prevented customers of Australia's Health Industry Claims and Payments Service (HICAPS) from paying for their health care services. HICAPS manages 18,500 terminals to manage health care cards for medical transactions.
- Users of the TomTom automobile GPS devices had to find their own way on Leap Day. Almost a week later, the Dutch manufacturer of the satellite navigation device determined that the problem was caused by a Leap Day bug in the GPS receiver software from a third-party supplier. The bug prevented TomTom devices from obtaining an accurate GPS location.
- Some of the lottery games run in Western Canada suffered a minor glitch due to Leap Day. Winning Lotto Max tickets must be redeemed within one year. Many with winning tickets bought in March and April of 2011 found that they could not cash in their tickets on the last allowable day because the program did not recognize the extra day in 2012.
- The ATM and EFTPOS networks of Commonwealth Bank of Australia were down for nine hours on or about Leap Day. The common conjecture was that it was a Leap Day problem, but the bank was quick to deny that, saying that it was caused by the bank's creaky 40-year-old IT system that it is in the process of replacing.¹

¹ [Australia's Painful Banking Outages](http://www.availabilitydigest.com/public_articles/0703/australian_bank_outages.pdf), *Availability Digest*, March 2012.
http://www.availabilitydigest.com/public_articles/0703/australian_bank_outages.pdf

- Even gaming sites were not spared. The Mobile Army game from Battlenet did not adjust properly for Leap Year. Starting with February 29th, the dates for all of its special events were off by one day.
- Sometimes, things work backwards. Two years ago, several models of Sony PlayStation 3 game consoles thought that 2010 was a leap year. The bug affected millions of users worldwide and threatened their ability to obtain trophies and other game achievements.

Among many others, two specific date/time testing products are available for systems that represent a bulk of mission-critical systems – Opta2000 from TANDsoft for HP NonStop servers and SoftDate from SoftBase for IBM zOS systems. We briefly review these products as examples of the capabilities of date/time simulators.

Opta2000

Opta2000 from TANDsoft (www.tandsoft.com) is a clock and time-zone simulator for HP NonStop servers. With Opta2000, each application can run under its own virtual system clock with no application modifications. Thus, applications can be easily tested for proper behavior for future dates. Equally important, existing production, backup and development systems can support consolidated worldwide applications running in different time zones.

No matter the application, the programming language, the system type, or the operating environment, all time calls generate a system call to an operating system procedure. By intercepting these calls, Opta2000 can manipulate the returned time to meet the simulated time requirements of the application group. For time-zone simulation, Opta2000 offsets the system's actual time by an amount expressed as a deviation from GMT time (Greenwich Mean Time). For clock simulation, the virtual system times can be offset to whatever time is required by the application.

Time simulation services are applied to flexibly defined application groups that can consist of a specified set of programs, processes, users, and so on. If an application is not defined within any group, it continues to operate off of the normal system time.

A system can host many application groups, and each group can specify its own time offset. Date/time offsets for the application groups are entered via a command-line interface.

Clock Simulation

Clock simulation has several uses. For one, the application can be easily tested for critical times. For instance, to test Leap Day, the application's clock would be set to just before midnight, February 28th, of the year to be tested. The application can be tested over and over again to monitor its behaviour as it transitions the critical time. This testing capability applies both to home-grown applications and to third-party applications.

Another use for clock simulation is to rerun a batch application that failed. The clock can be set to the time expected by the batch run, and the batch application rerun.

Opta2000 was used successfully in many Y2K testing projects to ensure proper operation as midnight, December 31, 1999 passed.

Time-Zone Simulation

With Opta2000's time-zone simulation, applications can be run on the same system as if they were located on different systems resident in various time zones. The various transitions between standard time and daylight savings time for all regions are supported. This allows a company to consolidate the

workloads of several geographically dispersed systems onto a single system located in some other foreign time zone.

In addition to production environments, time-zone simulation can be used to back up several diverse systems onto a single system. Though the various systems may be located in their own time zones, their backup images can be resident on a single system in some other time zone. Each backup image runs in a simulated time zone that matches that of its production system.

A major Canadian bank used Opta2000 to consolidate three production data centers and DR sites in two time zones (GMT-5 and GMT-8) into a single production site and a single DR site.

SoftDate

SoftDate from SoftBase (www.softbase.com) provides time and date simulation services for IBM zOS systems. SoftDate dynamically front-ends system routines to obtain the correct date and time. The front-ending occurs only in zOS regions where it is required. SoftDate can be turned on at the individual job step, TSO, CICS, or IMS user without impacting other jobs or users in the same system.

SoftDate confines its impact to only those regions and users that actually require it. There is no impact from SoftDate on regions which are not actively using it. There are no permanent changes to any system routines; and any region that is running without SoftDate active will execute only standard, unmodified system code.

Time simulation rules are defined via an intuitive ISPF (Interactive System Productivity Facility) interface. Thus, there is no need to make JCL changes in order to test time compliance. Notes can be attached to simulation rules to document them. Comprehensive security controls are provided to prevent time simulation where it is not intended. Exclusion lists allow users to specify programs that are never to use SoftDate.

SoftDate's rules can be assigned to groups of users or applications that can be differentiated by some combination of:

- job name
- step name
- procedure step name
- program name
- user ID
- subsystem name
- job class
- in the case of CICS, terminal ID, network ID, transaction ID, or user ID

Users have a great deal of flexibility in setting time simulation rules:

- They can set a clock value for all work run under their user ID.
- A clock value can be set for groups of users, transactions, and/or terminals.
- Rules can be activated and deactivated via batch.
- Rules can be automatically activated when CICS starts.

Users can set test dates and times within IMS without having to exit to ISPF.

Time-zone virtualization of zOS data centers supports data center consolidation. It allows local time to be presented to users who may be in widely dispersed geographic locations and time zones. For instance, a data center in New York may be hosting applications for users in Los Angeles, New York, London, Berlin, and Singapore. Each user will see his correct local time for his own time zone just as if the data center

were located in his city. Specific logical partitions (LPARs) are no longer needed to support branches, subsidiaries, or companies existing in different time zones.

SoftDate supports applications running under CICS, TSO, IMS, and WebSphere. CICS MRO (Multi-Region Operation) support ensures that a transaction started in one zOS region will carry the same time as its processing transitions other regions.

SoftDate supports Parallel Sysplex systems. Job, CICS, and IMS time simulation rules are synchronized across the Sysplex. MROs are supported to ensure time consistency for a transaction across regions.

All 64-bit zOS environments are supported, including support for 64-bit storage above the two gigabyte bar. 64-bit support is extended to the zOS DB2 10 database management system, including DB2 stored procedures.

The U. S. Social Security Administration has recently adopted SoftDate for time simulation purposes.

Other Time Simulation Products

There are many other date and time simulation products available in the marketplace. Though TANDsoft's Opta2000 is the only time simulator for HP NonStop systems, several others support IBM mainframes and Linux, Unix, and Windows systems. Some examples are:

- TICTOC from IBM supports a variety of IBM mainframes including zOS, OS/390, CICS, and DB2.
- Time Machine from Solution-Soft supports Windows, Linux, and Unix (HP-UX, Solaris, AIX) servers.
- SIM-Date from Cobbs Mill Consulting supports zOS, OS/390, MVS, and VSE.
- HourGlass from IBM supports zOS and CICS.
- HourGlass from Allegro supports Linux and HP-UX.
- Xpediter/Xchange from Compuware supports CICS, DB2, IMS/DC, and LE/370.

Time Testing Goes Beyond Y2K and Leap Years

Testing for proper Leap Day functionality is only one of many time-related challenges for application programmers. There are many other critical date transitions such as:

- the first or last business day of the week
- the first or last day of the month
- end of quarter or financial year
- public holidays such as Christmas and Independence Day
- clock rollover at midnight

In addition, many applications have critical dates that should be tested:

- Policy renewal and anniversary dates
- Bond and investment fund maturity dates
- Overdue payments
- License expiration

All of these cases and many more must be thoroughly tested before an application is deployed. Time simulation products are a necessary and important tool to ensure that an application will handle date/time-related functions properly.

Summary

Even with all the Y2K experience, there are still many failures due to date and time bugs. There is no reason for this. Products are available to thoroughly test applications to ensure that they handle critical dates and times successfully. These products are by and large noninvasive and require no program modifications. Simple means are provided to specify date simulation parameters for groups of applications or users based on a variety of parameters. These products can be used to offset times for testing or to simulate multiple time zones for system consolidation.

Time simulation products exist for most operating system environments, including those for IBM mainframes, HP NonStop servers, Linux, Windows, and most UNIX platforms.