Errores de Software Significativos

Vicente Benjumea García

Introducción a la Programación
Departamento de Lenguajes y Ciencias de la Computación.
E.T.S.I. Informática. Univ. de Málaga.

Errores de Software Significativos

Este documento presenta una breve descripción de algunos errores de software que han tenido una gran repercusión. De ellos se pueden extraer algunas conclusiones interesantes:

- Es muy difícil tener la seguridad de que el software es correcto, incluso para proyectos de miles de millones de euros.
  - Cuanto más complejo es un sistema software, más difícil es tener la seguridad de su corrección.
- Incluso los mejores programadores y analistas cometen errores.
  - Cuanto más complejo es un determinado problema, más complejo es el software que lo resuelve.
- Incluso los errores más simples (conversión de tipos, desbordamientos aritméticos, divisiones por cero, etc) hacen incorrecto al software.
- Las consecuencias del software erróneo pueden ser muy costosas, tanto en vidas humanas, como monetarias, tiempo y esfuerzo.
- Debemos seguir **metodologías de programación y estrategias adecuadas** para desarrollar **software de calidad**, que minimice las posibilidades de introducir errores en el software, y facilite la depuración (detección y corrección) de los errores existentes.

La información reflejada en este documento ha sido tomada de **Wikipedia**, a partir del siguiente enlace: `https://en.wikipedia.org/wiki/List_of_software_bugs`, así como de los enlaces que aparecen en la descripción de cada caso.

Errores de Software Significativos

Espacio

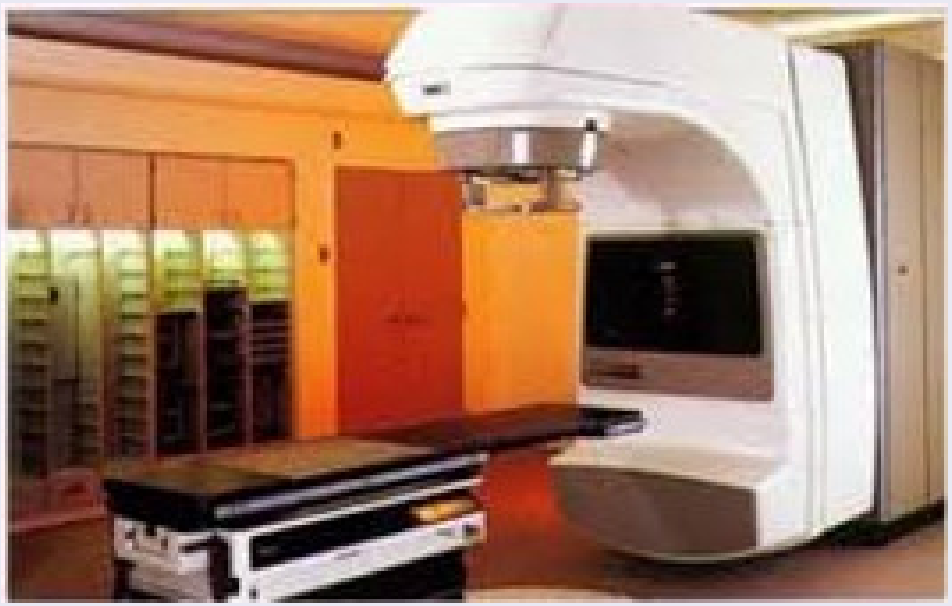
En 1962, durante el lanzamiento del **Mariner-1** de la NASA, falló el motor de propulsión auxiliar, con el resultado de la destrucción de la nave. El fallo fue debido a la codificación incorrecta de una **fórmula matemática**.

`https://en.wikipedia.org/wiki/Mariner_1`

Errores de Software Significativos

Salud

En la decada de 1980, la máquina de radioterapia **Therac-25** estuvo involucrada en varios accidentes que causaron el fallecimiento de varias personas. Debido a un error en el software de control de la máquina, los pacientes fueron sometidos a niveles de radiación erróneos.



`https://en.wikipedia.org/wiki/Therac-25`

Errores de Software Significativos

Espacio

En 1988, la nave rusa **Phobos-1**, desactivó sus propulsores de posición, y por lo tanto no podía orientar sus paneles solares adecuadamente, por lo que agotó sus baterías y quedó inutilizada. El error fue debido a la incorrecta ejecución de un **código de comprobación** que no debería haberse ejecutado durante la actividad de la nave, pero que fue activado por error desde la Tierra.

`https://en.wikipedia.org/wiki/Phobos_1`

Errores de Software Significativos

Militar

En 1991, un error de software en la **precisión** del mecanismo de control del tiempo de los misiles **MIM-104 Patriot** causó que no fuese capaz de interceptar un misil iraquí, que llegó a impactar en una base militar, causando el fallecimiento de decenas de personas.

`https://en.wikipedia.org/wiki/MIM-104_Patriot#Failure_at_Dhahran`

Errores de Software Significativos

Espacio

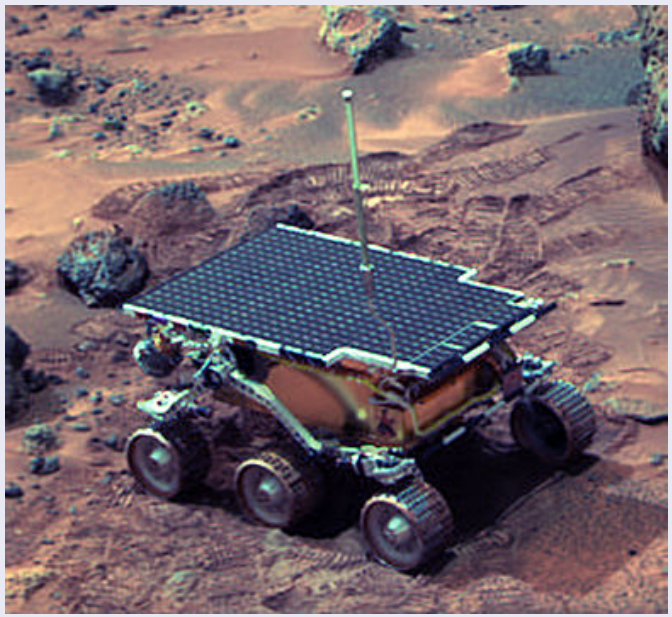
En 1996, la nave **Ariane-5** de la ESA, en el vuelo 501, se destruyó 40 segundos después del lanzamiento debido a un error en el software de guiado. El error fue debido a la ejecución de **código reutilizado** del software para el *Ariane-4*, que estaba diseñado para *magnitudes menores*, de tal forma que una conversión de un número de punto flotante de 64 bits a un número entero de 16 bits causó un **desbordamiento aritmético**.

`https://en.wikipedia.org/wiki/Cluster_(spacecraft)`

Errores de Software Significativos

Espacio

En 1997, la misión del vehículo **Sojourner** del proyecto **Mars-Pathfinder** de la NASA fue puesta en peligro debido a un error de **inversión de prioridad** en el software concurrente que controlaba el vehículo, provocando continuas *reinicializaciones* del sistema, debido a que la tarea más prioritaria era bloqueada por tareas de menor prioridad. Finalmente, el error de software pudo ser corregido y cargado desde la Tierra.



```
https://en.wikipedia.org/wiki/Mars_Pathfinder
https://en.wikipedia.org/wiki/Sojourner_(rover)
```

Errores de Software Significativos

Militar

En 1997, el buque **USS Yorktown** quedó inoperativo durante varias horas debido a un error software de **división por cero**.

`https://en.wikipedia.org/wiki/USS_Yorktown_(CG-48)#Smart_ship_testbed`

Errores de Software Significativos

Espacio

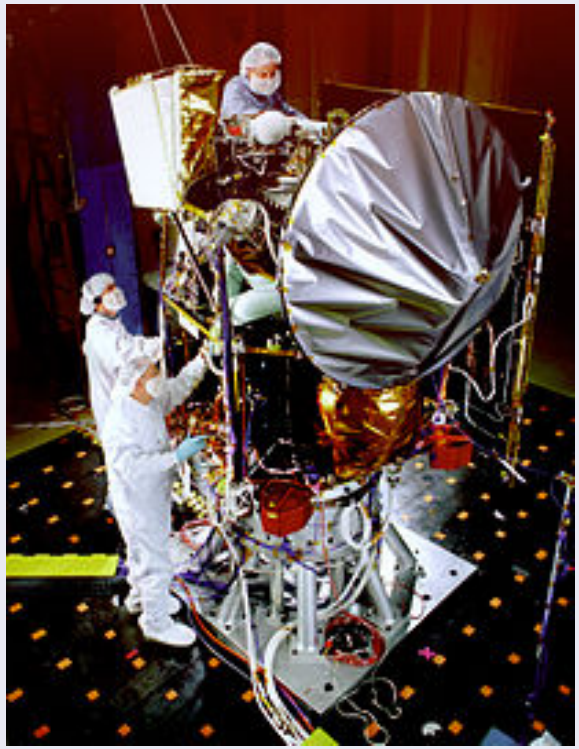
En 1999, la nave **Mars Polar Lander** de la NASA fue destruida debido a que el software de control de vuelo interpretó erróneamente las vibraciones producidas por las turbulencias, creyendo que la nave ya había llegado a la superficie, por lo que desconectó los motores a 40 metros de altura de la superficie de Marte.

`https://en.wikipedia.org/wiki/Mars_Polar_Lander`

Errores de Software Significativos

Espacio

En 1999, la nave **Mars Climate Orbiter** de la NASA fue destruida debido a que el software de la estación de control de la Tierra generaba los comandos en un determinada unidad de fuerza (*lbf*), mientras que el software de la estación orbital los esperaba en otra unidad de fuerza (*newtons*).



`https://en.wikipedia.org/wiki/Mars_Climate_Orbiter`

Errores de Software Significativos

Espacio

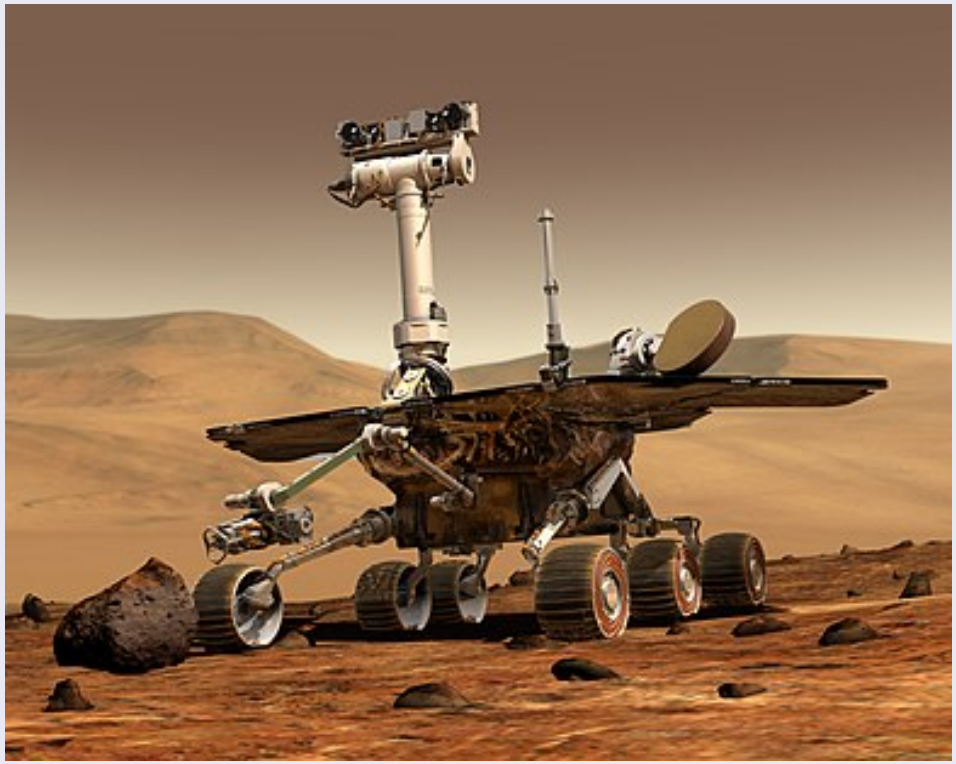
En 2000, el lanzamiento de una nave rusa **Zenit-3SL** falló debido a un error del software que causó un cierre prematuro de la segunda etapa, por lo que no pudo llegar a la órbita prevista para la colocación del satélite que transportaba.

`https://en.wikipedia.org/wiki/Zenit-3SL`

Errores de Software Significativos

Espacio

En 2004, el vehículo **Spirit** de la NASA quedó inoperativo unas pocas semanas después de su llegada a la superficie de Marte, debido a un error en el software de gestión del sistema de ficheros. Finalmente pudo ser corregido desde la Tierra.

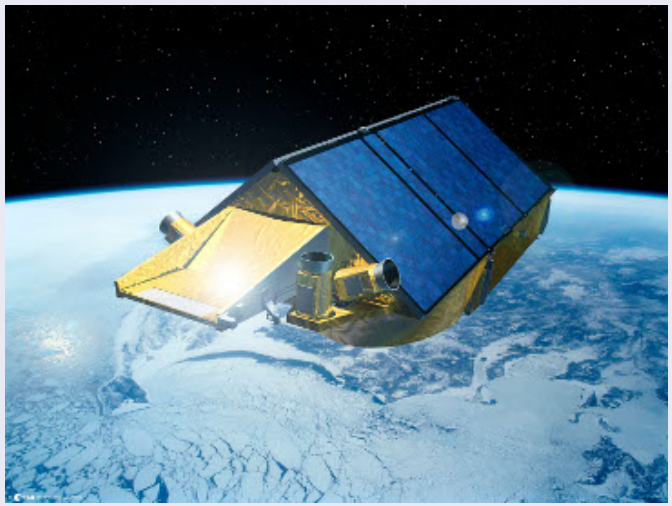


`https://en.wikipedia.org/wiki/Spirit_(rover)`

Errores de Software Significativos

Espacio

En 2005, se perdió el satelite **CryoSat-1** de la ESA, ya que el motor principal de la segunda etapa del cohete no se apagó adecuadamente debido a un error de programación, causando un error catastrófico en el sistema de control de vuelo del cohete **Rokot**.



`https://en.wikipedia.org/wiki/Rokot`
`https://en.wikipedia.org/wiki/CryoSat-1`

Errores de Software Significativos

Espacio

En 2006, una actualización de los parametros del software enviada desde la estación de control de la Tierra causó que el software de control de la nave **Mars Global Surveyor** de la NASA asumiera incorrectamente que un motor había fallado y que debía orientar a sus baterías hacia el Sol, causando su sobrecalientamiento, y la pérdida final de la nave.

`https://en.wikipedia.org/wiki/Mars_Global_Surveyor`

Errores de Software Significativos

Salud

En 2008, un equipo de seguridad fue capaz de tomar el control remoto de un dispositivo de **implante cardiaco Medtronic**,

`https://en.wikipedia.org/wiki/Medtronic#Technology_safety`
`https://www.nytimes.com/2008/03/12/business/12heart-web.html?_r=0`

Errores de Software Significativos

Transporte

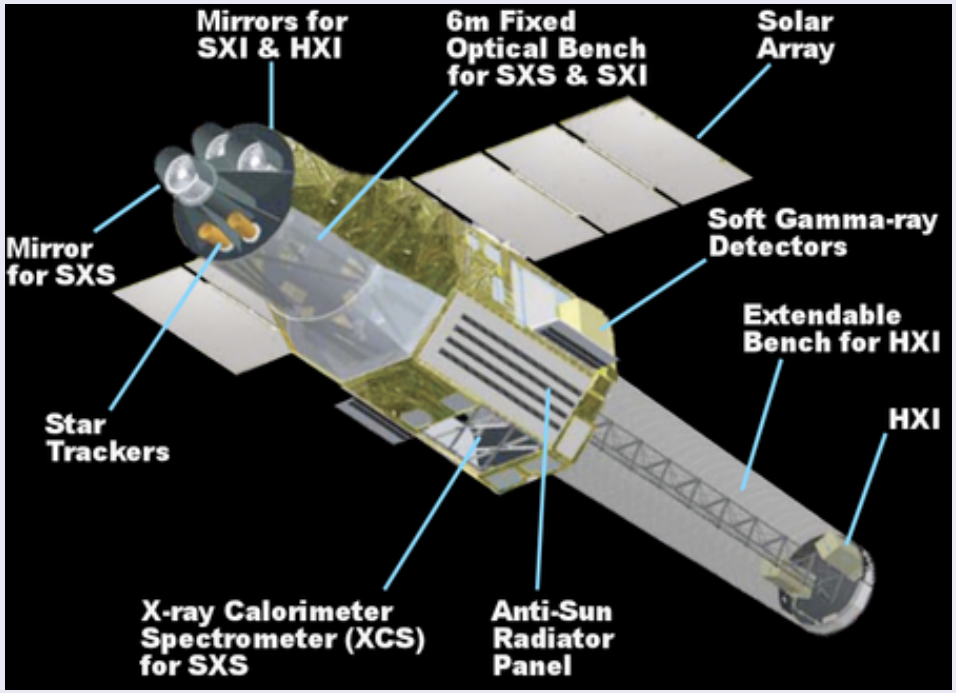
En 2015, en el avión **Boeing 787 Dreamliner**, se detectó un error de software de **desbordamiento aritmético** que apagaría todos los generadores eléctricos si el avión hubiese estado operativo por más de 248 días.

`https://en.wikipedia.org/wiki/Boeing_787_Dreamliner`
`https://www.engadget.com/2015/05/01/boeing-787-dreamliner-software-bug/`

Errores de Software Significativos

Espacio

En 2016, el satelite japonés **Hitomi** fue destruido debido a un error de software en el sistema de control de la orientación, causando que el satelite girase a máxima velocidad, en vez de estabilizarlo.



`https://en.wikipedia.org/wiki/Hitomi_(satellite)`

Errores de Software Significativos

Transporte

En 2018 y 2019, dos aviones **Boeing 737 MAX** sufrieron accidentes mortales cuya causa ha sido atribuida inicialmente a un error en el software del sistema de control de maniobras (MCAS). Actualmente se encuentra bajo investigación, y los aviones no pueden ser utilizados.

`https://en.wikipedia.org/wiki/Boeing_737_MAX`
`https://en.wikipedia.org/wiki/Boeing_737_MAX_groundings`

Errores de Software Significativos

Seguridad y criptografía

En 2014, la capa de seguridad SSL/TLS de *Apple* contenía un agujero de seguridad importante (CVE-2014-1266), conocido como **"goto fail bug"**. Debido a ello, la verificación de firmas digitales no se realizaba correctamente, y siempre resultaba con *éxito*.

```
static OSStatus
SSLVerifySignedServerKeyExchange(SSLContext *ctx, bool isRsa, SSLBuffer signedParams,
                                 uint8_t *signature, UInt16 signatureLen)
{
    OSStatus        err;
    ...

    if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0)
        goto fail;
    if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
        goto fail;
        goto fail;
    if ((err = SSLHashSHA1.final(&hashCtx, &hashOut)) != 0)
        goto fail;
    ...

fail:
    SSLFreeBuffer(&signedHashes);
    SSLFreeBuffer(&hashCtx);
    return err;
}
```

También se encontró un error similar en la capa de seguridad de GnuTLS, que permitía a los atacantes saltar la seguridad de SSL/TLS.

https://en.wikipedia.org/wiki/Unreachable_code#Examples
https://www.cve.org/CVERecord?id=CVE-2014-1266

Errores de Software Significativos

Seguridad y criptografía

En 2014, se descubrió una vulnerabilidad en la capa de seguridad *OpenSSL*, denominada *"Heartbleed"* (CVE-2014-0160), que fue introducida en 2012. La vulnerabilidad consistía en un error que permitía leer de un buffer más datos de los permitidos (*"buffer over-read"*), debido a un **error en la validación de los datos de entrada**, que no realizaba una comprobación de un valor dentro de los límites. Este error permitía a los atacantes acceder a información confidencial y privada, incluyendo contraseñas y claves privadas.

```
// Este código de comprobación hubiese evitado el error.
if (1 + 2 + payload + 16 > s->s3->rrec.length)
    return 0;
```

https://en.wikipedia.org/wiki/Heartbleed
https://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2014-0160

Errores de Software Significativos

Bibliografía

- https://en.wikipedia.org/wiki/List_of_software_bugs
- https://en.wikipedia.org/wiki/Mariner_1
- https://en.wikipedia.org/wiki/Therac-25
- https://en.wikipedia.org/wiki/Phobos_1
- https://en.wikipedia.org/wiki/MIM-104_Patriot#Failure_at_Dhahran
- https://en.wikipedia.org/wiki/Cluster_(spacecraft)
- https://en.wikipedia.org/wiki/Mars_Pathfinder
- https://en.wikipedia.org/wiki/Sojourner_(rover)
- https://en.wikipedia.org/wiki/USS_Yorktown_(CG-48)#Smart_ship_testbed
- https://en.wikipedia.org/wiki/Mars_Polar_Lander
- https://en.wikipedia.org/wiki/Mars_Climate_Orbiter
- https://en.wikipedia.org/wiki/Zenit-3SL
- https://en.wikipedia.org/wiki/Spirit_(rover)

Errores de Software Significativos

Bibliografía

- https://en.wikipedia.org/wiki/Rokot
- https://en.wikipedia.org/wiki/CryoSat-1
- https://en.wikipedia.org/wiki/Mars_Global_Surveyor
- https://en.wikipedia.org/wiki/Medtronic#Technology_safety
- https://www.nytimes.com/2008/03/12/business/12heart-web.html?_r=0
- https://en.wikipedia.org/wiki/Boeing_787_Dreamliner
- https://www.engadget.com/2015/05/01/boeing-787-dreamliner-software-bug/
- https://en.wikipedia.org/wiki/Hitomi_(satellite)
- https://en.wikipedia.org/wiki/Boeing_737_MAX_groundings
- https://en.wikipedia.org/wiki/Unreachable_code#Examples
- https://www.cve.org/CVERecord?id=CVE-2014-1266
- https://en.wikipedia.org/wiki/Heartbleed
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=cve-2014-0160