

**1970s**

[Computer Associates](https://en.wikipedia.org/wiki/Computer_Associates)**Panvalet®** (también conocido como **CA-Panvalet** ) es un [control de revisión](https://en.wikipedia.org/wiki/Revision_control) y un sistema de administración de código fuente para computadoras mainframe como [IBM System z](https://en.wikipedia.org/wiki/IBM_System_z) e [IBM System / 370 que](https://en.wikipedia.org/wiki/IBM_System/370) ejecutan los sistemas operativos [z / OS](https://en.wikipedia.org/wiki/Z/OS) y [z / VSE](https://en.wikipedia.org/wiki/VSE_(operating_system)) . A diferencia de las soluciones de código abierto como [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) , [SVN](https://en.wikipedia.org/wiki/Apache_Subversion) o [Mercurial](https://en.wikipedia.org/wiki/Mercurial) , Panvalet es un sistema patentado de código cerrado para el control de versiones y el código fuente como [Microsoft Visual SourceSafe](https://en.wikipedia.org/wiki/Microsoft_Visual_SourceSafe) en computadoras personales. Utiliza un [cliente-servidor](https://en.wikipedia.org/wiki/Client-server)modele donde los usuarios revisan los archivos para cambiarlos y revíselos en el depósito una vez finalizado. [[1]](https://en.wikipedia.org/wiki/Panvalet#cite_note-1)

Panvalet se puede usar para administrar el código fuente del programa, el [JCL](https://en.wikipedia.org/wiki/Job_Control_Language) y los [archivos del módulo de objetos](https://en.wikipedia.org/wiki/Object_file) . Admite controles de acceso granulares que incluyen el registro y la salida por ID de usuario de mainframe específicos. [[2]](https://en.wikipedia.org/wiki/Panvalet#cite_note-2)

En 1978, se informó que Panvalet, en ese momento un producto de Pansophic Systems, Inc, estaba en uso en más de 3.000 sitios.

|  |  |
| --- | --- |
| **CA Panvalet** | |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Computer Associates](https://en.wikipedia.org/wiki/CA_Technologies) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [z / OS](https://en.wikipedia.org/wiki/Z/OS) , [z / VSE](https://en.wikipedia.org/wiki/VSE_(operating_system)) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | Propiedad |
| **Sitio web** | [www .ca .com / us/ products / detail / ca-panvalet .aspx](http://www.ca.com/us/products/detail/ca-panvalet.aspx) |

**CA Harvest Software Change Manager** (originalmente conocido como **CCC / Harvest** ) es una herramienta de [software](https://en.wikipedia.org/wiki/Computer_software) para la [gestión de configuración](https://en.wikipedia.org/wiki/Configuration_management) ( [control de revisión](https://en.wikipedia.org/wiki/Revision_control) , [SCM](https://en.wikipedia.org/wiki/Software_configuration_management) , etc.) del [código fuente](https://en.wikipedia.org/wiki/Source_code) y otros activos de desarrollo de software.

El primer producto CCC (acrónimo de 'Cambio y control de configuración') se lanzó a principios de los años 70 y fue diseñado como un proyecto para un contratista del Departamento de Defensa en Santa Barbara CA. (La compañía en ese momento era [Hughes Aircraft](https://en.wikipedia.org/wiki/Hughes_Aircraft) , ahora el Centro de Investigación de Santa Bárbara para [Raytheon](https://en.wikipedia.org/wiki/Raytheon" \o "Raytheon) ). Se convirtió en la primera herramienta de CM disponible comercialmente.

CCC fue diseñado para gestionar todos los componentes que entraron en un [motor de avión](https://en.wikipedia.org/wiki/Aircraft_engine) , y ya que el mismo motor fue utilizado tanto por la [Fuerza Aérea de Estados Unidos](https://en.wikipedia.org/wiki/United_States_Air_Force) y de [la marina de guerra de Estados Unidos](https://en.wikipedia.org/wiki/United_States_Navy) (para el [F-14 Tomcat](https://en.wikipedia.org/wiki/F-14_Tomcat) y [F-15 Eagle](https://en.wikipedia.org/wiki/F-15_Eagle) ) que requieren robusto y fiable desarrollo paralelo.

La primera versión de **CCC / Harvest** fue desarrollada comercialmente por Softool Corporation, una compañía de software enfocada en CM fundada en 1977 en Goleta, California. Otras herramientas de CCC incluyen CCC / Manager, CCC / DM Turnkey y CCC / QuickTrak.

Softool fue adquirida a fines de 1995 por [Platinum Technology](https://en.wikipedia.org/wiki/Platinum_Technology" \o "Tecnología Platinum) , que luego fue adquirida en mayo de 1999 por Computer Associates (ahora conocida como [CA Technologies](https://en.wikipedia.org/wiki/CA_Technologies) ) que agregó CCC / Harvest a su suite AllFusion. En 2002, la parte 'CCC' del nombre se eliminó, y se agregó 'Change Manager', por lo que se hizo conocido como **AllFusion Harvest Change Manager** . Más tarde esto fue cambiado a **CA Harvest Software Change Manager** .

Funciones distintivas.

* **Cambie los paquetes:** Harvest puede proporcionar control de versiones y administración de cambios. El desarrollador realiza cambios en Harvest contra un paquete de cambio (creando un "conjunto de cambios"). Los paquetes de cambio consistirán inicialmente en una cantidad de archivos que el desarrollador ha creado o modificado. Este es el componente de control de versiones de Harvest.
* **Ciclos de vida: una** vez que el desarrollador está satisfecho con sus cambios, los cambios progresan a través de un ciclo de vida predefinido (es decir, en un número de etapas de PRUEBA secuenciales y finalmente en PRODUCCIÓN). En todas estas etapas de este "ciclo de vida", el paquete debe tener aprobaciones de los usuarios o grupos de usuarios apropiados. Estas aprobaciones se registran permanentemente para fines de auditoría. Por ejemplo, un gerente de pruebas puede tener que aprobar paquetes antes de pasar a la etapa de PRUEBA, y el equipo de administración de cambios de producción puede tener que aprobar paquetes antes de pasar al estado PROD.
* **Proyectos (Ambientes):** La filosofía de Central to Harvest es el concepto de un "proyecto" de cosecha. Los proyectos son totalmente personalizables según las necesidades de la aplicación, la organización o el equipo. El término proyecto se refiere a todo el marco de control en Harvest e incluye:
  + Una rama o línea de desarrollo separada donde los cambios pueden aislarse (el componente de control de versiones)
  + La definición de procesos y cómo los cambios progresan a través del ciclo de vida promocional
  + Control de acceso para procesos y archivos.

1972

**Source Code Control System** (**SCCS**) es un [sistema de control de versiones](https://en.wikipedia.org/wiki/Version_control) diseñado para rastrear los cambios en el [código fuente](https://en.wikipedia.org/wiki/Source_code) y otros archivos de texto durante el desarrollo de una pieza de software. Esto le permite al usuario recuperar cualquiera de las versiones anteriores del código fuente original y los cambios que se almacenan. Originalmente fue desarrollado en [Bell Labs](https://en.wikipedia.org/wiki/Bell_Labs) en 1972 por [Marc Rochkind](https://en.wikipedia.org/wiki/Marc_Rochkind) para una [computadora IBM System / 370](https://en.wikipedia.org/wiki/IBM_System/370) que ejecuta [OS / 360](https://en.wikipedia.org/wiki/OS/360) .

|  |  |
| --- | --- |
| **fuente** | |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | Marc J. Rochkind |
| **Versión inicial** | 1972 ; Hace 46 años |
| [**Repositorio**](https://en.wikipedia.org/wiki/Repository_(version_control)) | [Página web oficial](http://pubs.opengroup.org/) |
| **Escrito en** | [do](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Unix-like](https://en.wikipedia.org/wiki/Unix-like) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de versiones](https://en.wikipedia.org/wiki/Version_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | licencias de propiedad, licencia de[desarrollo común y distribución](https://en.wikipedia.org/wiki/Common_Development_and_Distribution_License) |
| **Sitio web** | [Las especificaciones de la base Open Group Edición 7, IEEE Std 1003.1-2008, edición de 2016](http://pubs.opengroup.org/onlinepubs/9699919799/utilities/sccs.html) |

Se considera el primer sistema de control de versiones oficial de la historia, ya que es la herramienta de control de versiones más antigua conocida. Sin rival hasta la llegada de RCS. SCCS basaba su gestión de las versiones en el procesamiento de ficheros individuales, además de cara a la concurencia, requería que cada uno de los desarrolladores del proyecto realizasen cambios en un momento determinado. Las versiones era protegidas mediante protecciones de seguridad a modo de bloqueo para el acceso del resto de desarrolladores, lo que originaba problemas de administración por olvido de retirar estos bloqueos. En 1972, [Marc Rochkind](https://en.wikipedia.org/wiki/Marc_Rochkind) desarrolló SCCS en [SNOBOL4](https://en.wikipedia.org/wiki/SNOBOL) en [Bell Labs](https://en.wikipedia.org/wiki/Bell_Labs) para una [computadora IBM System / 370](https://en.wikipedia.org/wiki/IBM_System/370) que ejecuta [OS / 360](https://en.wikipedia.org/wiki/OS/360)[MVT](https://en.wikipedia.org/wiki/Multiprogramming_with_a_Variable_number_of_Tasks) . [[1]](https://en.wikipedia.org/wiki/Source_Code_Control_System#cite_note-r1-1) Más tarde fue reescrito por él en [C](https://en.wikipedia.org/wiki/C_(programming_language)" \o "C (lenguaje de programación))para [UNIX](https://en.wikipedia.org/wiki/UNIX) , y luego se ejecuta en un [PDP-11](https://en.wikipedia.org/wiki/PDP-11) .

La primera versión lanzada al público fue SCCS versión 4 del 18 de febrero de 1977. [[3]](https://en.wikipedia.org/wiki/Source_Code_Control_System#cite_note-3) Estaba disponible con la edición [Programmer's Workbench](https://en.wikipedia.org/wiki/PWB/UNIX" \o "PWB / UNIX) (PWB) del [sistema operativo](https://en.wikipedia.org/wiki/Operating_system) . La versión 4 de SCCS fue la primera versión que utilizó un formato de archivo de historial basado en texto, las versiones anteriores sí utilizaron formatos de archivo de historial binario. La versión 4 ya no fue escrita o mantenida por Mark Rochkind. Posteriormente, SCCS se incluyó en las distribuciones comerciales [System III](https://en.wikipedia.org/wiki/UNIX_System_III" \o "Sistema UNIX III) y [System V de](https://en.wikipedia.org/wiki/UNIX_System_V" \o "Sistema UNIX V)[AT & T.](https://en.wikipedia.org/wiki/AT%26T)No tenía licencia con [32V](https://en.wikipedia.org/wiki/UNIX/32V) , el antecesor de [BSD](https://en.wikipedia.org/wiki/Berkeley_Software_Distribution) . [[4]](https://en.wikipedia.org/wiki/Source_Code_Control_System#cite_note-4) El conjunto de comandos de SCCS ahora es parte de la [especificación de UNIX único](https://en.wikipedia.org/wiki/Single_UNIX_Specification) .

SCCS fue el sistema de control de versiones dominante para Unix hasta que [los](https://en.wikipedia.org/wiki/Version_control) sistemas de [control de versiones](https://en.wikipedia.org/wiki/Version_control) posteriores , especialmente el [RCS](https://en.wikipedia.org/wiki/Revision_Control_System) y [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) posterior , obtuvieron una adopción más generalizada. Hoy en día, estos primeros sistemas de control de versiones generalmente se consideran obsoletos, particularmente en la comunidad de [código abierto](https://en.wikipedia.org/wiki/Open_source) , que en gran medida ha adoptado sistemas de [control de versiones distribuidas](https://en.wikipedia.org/wiki/Distributed_version_control) . Sin embargo, el formato de archivo SCCS todavía se usa internamente en algunos programas de control de versiones más nuevos, incluidos *[BitKeeper](https://en.wikipedia.org/wiki/BitKeeper" \o "BitKeeper)* y *[TeamWare](https://en.wikipedia.org/wiki/TeamWare" \o "TeamWare)* . Este último es un frontend de SCCS. *[Sablime](http://sablime.alcatel-lucent.com/)* se ha desarrollado a partir de una versión modificada de SCCS [[5]](https://en.wikipedia.org/wiki/Source_Code_Control_System#cite_note-5)pero usa un formato de archivo de historial que es incompatible con SCCS. El formato de archivo SCCS utiliza una técnica de almacenamiento llamada [deltas intercalados](https://en.wikipedia.org/wiki/Interleaved_deltas) (o el tejido [[6]](https://en.wikipedia.org/wiki/Source_Code_Control_System#cite_note-6) ). Esta técnica de almacenamiento ahora es considerada por muchos desarrolladores de sistemas de [control de](https://en.wikipedia.org/wiki/Version_control) versiones como fundamentales para las técnicas avanzadas de [fusión](https://en.wikipedia.org/wiki/Merge_(version_control)) y control de versiones, [[7]](https://en.wikipedia.org/wiki/Source_Code_Control_System#cite_note-7) como la [fusión](https://en.wikipedia.org/wiki/Merge_(version_control)) "Precise [Codeville](https://en.wikipedia.org/wiki/Codeville" \o "Codeville) " ("pcdv").

Además de solucionar algunos [problemas del año 2000](https://en.wikipedia.org/wiki/Year_2000_problem) en 1999, no existe un desarrollo activo en las diversas versiones SCCS específicas del proveedor de UNIX. [[8]](https://en.wikipedia.org/wiki/Source_Code_Control_System#cite_note-8) En 2006, [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems" \o "Sun Microsystems) (hoy parte de [Oracle](https://en.wikipedia.org/wiki/Oracle_Corporation) ) lanzó su versión [Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)) de SCCS como [código abierto](https://en.wikipedia.org/wiki/Open_source) bajo la [licencia CDDL](https://en.wikipedia.org/wiki/Common_Development_and_Distribution_License) como parte de sus esfuerzos para abrir Solaris.

1980

**Dimensions CM** es un producto de [gestión de configuración y cambio de software](https://en.wikipedia.org/wiki/Software_configuration_management)[[4]](https://en.wikipedia.org/wiki/Dimensions_CM#cite_note-4) desarrollado por [Serena Software](https://en.wikipedia.org/wiki/Serena_Software) . Incluye [control de revisiones](https://en.wikipedia.org/wiki/Revision_control) , [cambio](https://en.wikipedia.org/wiki/Change_management) , [construcción](https://en.wikipedia.org/wiki/Build_management)[[5]](https://en.wikipedia.org/wiki/Dimensions_CM#cite_note-5) y capacidades de [gestión de versiones](https://en.wikipedia.org/wiki/Release_management) . Nombres de productos anteriores:

* Dimensiones PCMS (Software SQL) [[7]](https://en.wikipedia.org/wiki/Dimensions_CM#cite_note-7)
* Dimensiones PVCS (Merant, Intersolv)

|  |  |
| --- | --- |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Serena Software](https://en.wikipedia.org/wiki/Serena_Software) |
| **Versión inicial** | 1980 ; Hace 38 años [[*citación necesitada*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 14.2 [[1]](https://en.wikipedia.org/wiki/Dimensions_CM#cite_note-1) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Multiplataforma](https://en.wikipedia.org/wiki/Cross-platform)[[2]](https://en.wikipedia.org/wiki/Dimensions_CM#cite_note-2)[[3]](https://en.wikipedia.org/wiki/Dimensions_CM#cite_note-3) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Gestión de configuración de software](https://en.wikipedia.org/wiki/Software_configuration_management) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Serena Software](https://en.wikipedia.org/wiki/Serena_Software)[EULA](https://en.wikipedia.org/wiki/End-user_license_agreement) |
| **Sitio web** | [serena.com/index.php/en/products/featured-products/dimensions-cm](http://www.serena.com/index.php/en/products/featured-products/dimensions-cm/) |

**Endevor** es una herramienta de [administración de lanzamiento](https://en.wikipedia.org/wiki/Release_management) y [administración de código fuente](https://en.wikipedia.org/wiki/Source_code_management) para [computadoras mainframe que](https://en.wikipedia.org/wiki/Mainframe_computer) ejecutan [z / OS](https://en.wikipedia.org/wiki/Z/OS) . [[1]](https://en.wikipedia.org/wiki/Endevor#cite_note-1) Es parte de una familia de herramientas de administración de [CA Technologies](https://en.wikipedia.org/wiki/CA_Technologies) (anteriormente Computer Associates), que se utiliza para mantener aplicaciones de software y rastrear sus versiones. [[2]](https://en.wikipedia.org/wiki/Endevor#cite_note-2)

La palabra ENDEVOR es un acrónimo que originalmente significaba Medio ambiente para desarrolladores y operaciones, pero que ahora es el nombre formal del producto principal de control de fuente de [Application Lifecycle Management](https://en.wikipedia.org/wiki/Application_Lifecycle_Management" \o "Gestión del ciclo de vida de la aplicación) de CA mainframe . También compite contra otra herramienta de gestión de código fuente de CA, [Panvalet](https://en.wikipedia.org/wiki/Panvalet" \o "Panvalet) .

**IBM Software Configuration and Library Manager** (SCLM) es un producto de software de [IBM](https://en.wikipedia.org/wiki/IBM) que es un componente de [ISPF](https://en.wikipedia.org/wiki/ISPF) .

Se introdujo por primera vez con la versión 3 de ISPF y fue una actualización de un componente anterior llamado LMF (Facilidad de administración de bibliotecas) que se introdujo en la versión 2 de ISPF. [[1]](https://en.wikipedia.org/wiki/IBM_Software_Configuration_and_Library_Manager#cite_note-1) Es un componente de [MVS](https://en.wikipedia.org/wiki/MVS) , [OS / 390](https://en.wikipedia.org/wiki/OS/390) y [z / OS](https://en.wikipedia.org/wiki/Z/OS)[sistemas operativos](https://en.wikipedia.org/wiki/Operating_system) / [plataformas](https://en.wikipedia.org/wiki/Computing_platform) . El SCLM consta de dos productos: un administrador de biblioteca y un [administrador de configuración](https://en.wikipedia.org/wiki/Software_configuration_management) y proporciona las siguientes funciones:

1. Administra los cambios en los datos de la aplicación
2. Realiza auditorías y [versiones](https://en.wikipedia.org/wiki/Version_control)
3. Controla el movimiento de los datos de la aplicación de un conjunto de bibliotecas de etapas al siguiente (conocido como Promover en SCLM)
4. Rastrea los componentes de la aplicación
5. Proporciona una función de compilación inteligente
6. Proporciona una promoción segura
7. Se integra con [Tivoli Information Management](https://en.wikipedia.org/w/index.php?title=IBM_Tivoli_Framework&action=edit&redlink=1" \o "IBM Tivoli Framework (la página no existe)) para z / OS y [WebSphere](https://en.wikipedia.org/wiki/WebSphere" \o "WebSphere) Studio Asset Analyzer para z / OS

El recurso IBM Software Configuration and Library Manager (SCLM) for z/OS es un producto de gestión de configuración de software (SCM). SCLM for z/OS consta de dos productos: un gestor de bibliotecas y un gestor de configuración para la plataforma IBM z/OS. SCLM for z/OS es una característica de ISPF (Interactive System Productivity Facility), que se incluye con el sistema operativo z/OS.

SCLM for z/OS incluye:

* **Gestor de bibliotecas**: gestiona los cambios en los datos de las aplicaciones, lleva a cabo las auditorías y el mantenimiento de versiones y controla el traspaso de la aplicación de un conjunto de bibliotecas intermedias al siguiente.
* **Gestor de configuración**: sabe cómo encajan todas las piezas de una aplicación, incluidos los módulos de origen, de objetos y de carga. Permite especificar relaciones adicionales en SCLM para asociar otros datos, como casos de prueba, publicaciones y JCL, con la aplicación.
* **Gestión de cambios**: permite asociar un código de cambio con un miembro y, a continuación, crear, promover y crear informes según los códigos de cambio. También proporciona una interfaz de muestra para IBM Tivoli Information Management for z/OS, que realiza el seguimiento de los registros de problemas y cambios hasta la finalización.
* **Análisis del impacto**: lleva a cabo análisis hipotéticos de las posibles consecuencias de un cambio antes de que se realice una compilación o una promoción. Genera informes personalizados del contenido de las aplicaciones controladas por SCLM e informes que muestran la relación entre las partes de la aplicación.
* **Características de ISPF**: proporciona funciones de visualización y exploración de SCLM que son iguales a sus equivalentes de ISPF. La función de edición de SCLM se crea sobre la edición de ISPF y admite las macros y los mandatos de edición de ISPF, la edición sensible al idioma y la herramienta Edit Compare. Una opción de tabla de configuración de ISPF determina si se puede acceder a un miembro desde fuera de SCLM.

1982 **Revision Control System** **(RCS)** es un [sistema de control de versiones](https://en.wikipedia.org/wiki/Version_control_system) inicial ( [VCS](https://en.wikipedia.org/wiki/Version_control_system) ). Se puede considerar como un conjunto de comandos de [UNIX](https://en.wikipedia.org/wiki/Unix) que permiten que varios usuarios desarrollen y mantengan código de programa o documentos. Con RCS, los usuarios pueden hacer sus propias revisiones de un documento, realizar cambios y fusionarlos. RCS fue desarrollado originalmente para programas, pero también es útil para documentos de texto o archivos de configuración que se revisan con frecuencia.

|  |  |
| --- | --- |
| **Sistema de control de revisión** | |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Walter F. Tichy](https://en.wikipedia.org/wiki/Walter_F._Tichy) |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Proyecto GNU](https://en.wikipedia.org/wiki/GNU_Project) |
| **Versión inicial** | 1982 ; Hace 36 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 5.9.4 (22 de enero de 2015 ; hace 3 años [[1]](https://en.wikipedia.org/wiki/Revision_Control_System#cite_note-1) )[[±]](https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Revision_Control_System&action=edit) |
| [**Repositorio**](https://en.wikipedia.org/wiki/Repository_(version_control)) | * <http://git.savannah.gnu.org/cgit/rcs.git/?h=p>   [Edita esto en Wikidata](https://www.wikidata.org/wiki/Q705851#P1324) |
| **Escrito en** | [do](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Unix-like](https://en.wikipedia.org/wiki/Unix-like) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de versiones](https://en.wikipedia.org/wiki/Version_Control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Licencia pública general de GNU](https://en.wikipedia.org/wiki/GNU_General_Public_License) |
| **Sitio web** | [www .gnu .org / s / rcs /](https://www.gnu.org/s/rcs/) |

### Modo de operación

RCS opera solo en archivos individuales. No tiene forma de trabajar con un proyecto completo, por lo que no admite [confirmaciones atómicas que](https://en.wikipedia.org/wiki/Atomic_commit#Revision_control) afecten a múltiples archivos. Aunque proporciona ramificación para archivos individuales, la sintaxis de la versión es engorrosa. En lugar de utilizar ramas, muchos equipos simplemente usan el mecanismo de bloqueo incorporado y trabajan en una sola rama de la *cabeza*.

### Uso

RCS gira en torno al uso de "grupos de revisión" o conjuntos de archivos que se han registrado a través de los comandos "co" (pago) y "ci" (check-in). De forma predeterminada, un archivo registrado se elimina y se reemplaza con un archivo ", v" (por lo que foo.rb cuando se activa se convierte en foo.rb, v) que luego cualquier persona que tenga acceso al grupo de revisión puede revisar. Los archivos RCS (nuevamente, los archivos con la extensión ", v") reflejan el archivo principal con metadatos adicionales en sus primeras líneas. Una vez registrado, RCS almacena las revisiones en una estructura de árbol que se puede seguir para que un usuario pueda revertir un archivo a un formulario anterior si es necesario. [[2]](https://en.wikipedia.org/wiki/Revision_Control_System#cite_note-:1-2)

### Ventajas

### Estructura simple y fácil de trabajar con

* El ahorro de revisión no depende de un repositorio central

### Desventajas

* Hay poca seguridad, en el sentido de que el historial de versiones puede ser editado por los usuarios.
* Solo un usuario puede trabajar en un archivo a la vez.

1984 DSEE

1985 **PVCS Version Manager** (originalmente llamado **Polytron Version Control System** ) es un paquete de software de [Serena Software](https://en.wikipedia.org/wiki/Serena_Software)Inc., para el [control](https://en.wikipedia.org/wiki/Version_control) de [versiones](https://en.wikipedia.org/wiki/Version_control) de archivos de [códigos fuente](https://en.wikipedia.org/wiki/Source_code) .

PVCS sigue el enfoque de "bloqueo" para el control de concurrencia; no tiene un operador de [*fusión*](https://en.wikipedia.org/wiki/Merge_(version_control)) incorporado (pero, sin embargo, tiene un comando de fusión separado). Sin embargo, PVCS también se puede configurar para admitir a varios usuarios al mismo tiempo que intentan editar el archivo; en este caso, el segundo committer cronológico tendrá una rama creada para él / ella, de modo que ambas modificaciones, en lugar de estar en conflicto, aparecerán como historias paralelas para el mismo archivo. Esto es diferente de [Concurrent Versions System](https://en.wikipedia.org/wiki/Concurrent_Versions_System" \o "Sistema de versiones concurrentes) (CVS) y [Subversion](https://en.wikipedia.org/wiki/Subversion_(software)" \o "Subversion (software)) donde el segundo committer necesita *fusionar* primero los cambios a través del comando de *actualización* y luego resolver los conflictos (cuando existen) antes de comprometerse.

Originalmente desarrollado por Don Kinzer y publicado por [Polytron](https://en.wikipedia.org/wiki/Polytron_(software)" \o "Polytron (software)) en 1985, a través de una historia de adquisiciones y fusiones, el producto a veces fue propiedad de Sage Software de Rockville (1989), Maryland (no relacionado con [Sage Software](https://en.wikipedia.org/wiki/Sage_Software" \o "Sage Software) del Reino Unido), [Intersolv](https://en.wikipedia.org/w/index.php?title=Intersolv&action=edit&redlink=1" \o "Intersolv (la página no existe)) 1992, [Micro Focus International](https://en.wikipedia.org/wiki/Micro_Focus_International) 1998 y [Merant PLC](https://en.wikipedia.org/wiki/Merant_PLC" \o "Merant PLC) 2001. Este último fue adquirido por Serena Software en 2004, que luego fue adquirido por [Silver Lake Partners](https://en.wikipedia.org/wiki/Silver_Lake_Partners" \o "Socios de Silver Lake) en 2006.

[Synergex](https://en.wikipedia.org/w/index.php?title=Synergex&action=edit&redlink=1) portó tanto el Gestor de versiones de PVCS como el Constructor de configuraciones de PVCS (una [utilidad de](https://en.wikipedia.org/wiki/Make_(software)) creación extendida , que incluye una variante de la herramienta de línea de comandos que hace [[2]](https://en.wikipedia.org/wiki/PVCS#cite_note-2)[[*mejor fuente necesaria*](https://en.wikipedia.org/wiki/Wikipedia:NOTRS)] ) a varias plataformas [Unix](https://en.wikipedia.org/wiki/Unix) y [OpenVMS](https://en.wikipedia.org/wiki/OpenVMS" \o "OpenVMS) . [[3]](https://en.wikipedia.org/wiki/PVCS#cite_note-3)

En 2009, Serena Software aclaró que continuará invirtiendo en PVCS y brindará soporte a los clientes de PVCS en el futuro previsible.   
La versión de PVCS Version Manager 8.5 (2014) presenta tanto la nueva característica como el nuevo soporte de plataforma. [[1]](https://en.wikipedia.org/wiki/PVCS#cite_note-blog-1)

En 2016, [Micro Focus International](https://en.wikipedia.org/wiki/Micro_Focus_International) anunció la adquisición de Serena Software para volver a ser los custodios de PVCS.

|  |  |
| --- | --- |
| **Gerente de versión de PVCS** | |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Polytron](https://en.wikipedia.org/wiki/Polytron) |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Serena Software](https://en.wikipedia.org/wiki/Serena_Software) |
| **Versión inicial** | 1985 ; Hace 33 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 8.5 / 2 de enero de 2014 ; Hace 4 años [[1]](https://en.wikipedia.org/wiki/PVCS#cite_note-blog-1) |
| **Escrito en** | [C](https://en.wikipedia.org/wiki/C_(programming_language)) , [C ++](https://en.wikipedia.org/wiki/C%2B%2B) , [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [DOS](https://en.wikipedia.org/wiki/DOS) , [[*cita requerida*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)][Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) , tipo [Unix](https://en.wikipedia.org/wiki/Unix-like) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de versiones](https://en.wikipedia.org/wiki/Version_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Software propietario](https://en.wikipedia.org/wiki/Proprietary_software) |
| **Sitio web** | [www .microfocus.com / products / pvcs /](https://www.microfocus.com/products/pvcs/) |

1986 **Concurrent Versions System** (**CVS**) también conocido como el **Sistema Concurrent Versioning** , es un sistema de [control de revisiones](https://en.wikipedia.org/wiki/Revision_control)[cliente-servidor de](https://en.wikipedia.org/wiki/Client-server)[software libre](https://en.wikipedia.org/wiki/Free_software) en el campo del [desarrollo](https://en.wikipedia.org/wiki/Software_development) de [software](https://en.wikipedia.org/wiki/Software_development) . Un sistema de control de versiones realiza un seguimiento de todo el trabajo y todos los cambios en un conjunto de archivos, y permite que varios desarrolladores (potencialmente muy separados en el espacio y el tiempo) [colaboren](https://en.wikipedia.org/wiki/Collaboration) . [Dick Grune](https://en.wikipedia.org/wiki/Dick_Grune) desarrolló CVS como una serie de [guiones](https://en.wikipedia.org/wiki/Shell_scripts) de [shell](https://en.wikipedia.org/wiki/Shell_scripts" \o "Shell scripts) en julio de 1986. [[1]](https://en.wikipedia.org/wiki/Concurrent_Versions_System#cite_note-1)

Además de los desarrolladores de software comercial, CVS se hizo popular con el mundo del [software de código abierto](https://en.wikipedia.org/wiki/Open_source_software) y fue lanzado bajo la [Licencia Pública General de GNU](https://en.wikipedia.org/wiki/GNU_General_Public_License) . Si bien hubo un desarrollo regular para agregar funciones y corregir errores en el pasado, [[2]](https://en.wikipedia.org/wiki/Concurrent_Versions_System#cite_note-2) incluidas compilaciones regulares y resultados de pruebas, [[3]](https://en.wikipedia.org/wiki/Concurrent_Versions_System#cite_note-3) no ha habido nuevas versiones desde 2008.

|  |  |
| --- | --- |
| **Sistema de versiones concurrentes** | |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [El equipo de CVS](http://savannah.nongnu.org/project/memberlist.php?detailed=1&group=cvs) |
| **Versión inicial** | 19 de noviembre de 1990 ; Hace 27 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 1.11.23 / 8 de mayo de 2008 ; Hace 9 años |
| [**Lanzamiento de vista previa**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 1.12.13 / 26 de julio de 2006 ; Hace 11 años |
| [**Repositorio**](https://en.wikipedia.org/wiki/Repository_(version_control)) | * cvs: //anonymous@cvs.sv.gnu.org/sources/cvs   [Edita esto en Wikidata](https://www.wikidata.org/wiki/Q467252#P1324) |
| **Escrito en** | [do](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Unix](https://en.wikipedia.org/wiki/Unix-like) , [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Licencia pública general de GNU](https://en.wikipedia.org/wiki/GNU_General_Public_License) |
| **Sitio web** | [savannah .nongnu .org / projects/ cvs](http://savannah.nongnu.org/projects/cvs) |

CVS utiliza una arquitectura [cliente-servidor](https://en.wikipedia.org/wiki/Client%E2%80%93server) : un servidor almacena la (s) versión (es) actual (es) de un [proyecto](https://en.wikipedia.org/wiki/Project) y su historial, y los clientes se conectan al servidor para "verificar" una copia completa del proyecto, trabajar en esta copia y luego más tarde "registrar" sus cambios. Normalmente, el cliente y el servidor se conectan a través de una [LAN](https://en.wikipedia.org/wiki/Local_area_network) o de [Internet](https://en.wikipedia.org/wiki/Internet) , pero tanto el cliente como el servidor pueden ejecutarse en la misma máquina si CVS tiene la tarea de realizar un seguimiento del historial de versiones de un proyecto solo con desarrolladores locales. El software del servidor normalmente se ejecuta en [Unix](https://en.wikipedia.org/wiki/Unix) (aunque al menos el servidor [CVSNT](https://en.wikipedia.org/wiki/CVSNT) también es compatible con varios sabores de [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) ), mientras que los clientes de CVS pueden ejecutarse en cualquierplataforma del [sistema operativo](https://en.wikipedia.org/wiki/Operating_system) .

Varios desarrolladores pueden trabajar en el mismo proyecto al mismo tiempo, cada uno editando archivos dentro de su propia "copia de trabajo" del proyecto y enviando (o *registrando* ) sus modificaciones al servidor. Para evitar conflictos, el servidor solo acepta los cambios realizados en la versión más reciente de un archivo. Por lo tanto, se espera que los desarrolladores mantengan actualizada su copia de trabajo incorporando los cambios de otras personas de forma regular. El cliente de CVS maneja esta tarea en su mayoría automáticamente, y requiere intervención manual solo cuando surge un [conflicto de edición](https://en.wikipedia.org/wiki/Edit_conflict) entre una modificación registrada y la versión local aún no verificada de un archivo.

Si la operación de verificación se realiza correctamente, los números de versión de todos los archivos involucrados se incrementan automáticamente, y el servidor CVS escribe una línea de descripción proporcionada por el usuario, la fecha y el nombre del autor en sus archivos de [registro](https://en.wikipedia.org/wiki/Data_logging) . CVS también puede ejecutar scripts externos de procesamiento de registro especificados por el usuario después de cada confirmación. Estas secuencias de comandos se instalan mediante una entrada en el archivo de inicio de sesión de CVS , que puede desencadenar notificaciones por correo electrónico o convertir los datos de registro en un formato basado en web.

Los clientes también pueden comparar versiones, solicitar un historial completo de cambios o consultar una instantánea histórica del proyecto a partir de una fecha determinada o de un número de revisión.

Los servidores CVS pueden permitir el "acceso de lectura anónimo", [[5]](https://en.wikipedia.org/wiki/Concurrent_Versions_System#cite_note-anoncvs-5) en el que los clientes pueden verificar y comparar versiones con una contraseña en blanco o simple publicada (por ejemplo, "anoncvs"); solo el registro de cambios requiere una cuenta personal y contraseña en estos escenarios.

Los clientes también pueden usar el comando "actualizar" para actualizar sus copias locales con la versión más reciente en el servidor. Esto elimina la necesidad de descargas repetidas de todo el proyecto.

CVS también puede mantener diferentes "ramas" de un proyecto. Por ejemplo, una versión publicada del proyecto de software puede formar una rama, utilizada para corregir errores, mientras que una versión bajo desarrollo actual, con cambios importantes y nuevas características, puede formar una rama separada.

CVS usa [compresión delta](https://en.wikipedia.org/wiki/Delta_compression) para un almacenamiento eficiente de diferentes versiones del mismo archivo. Esto funciona bien con archivos de texto grandes con pocos cambios de una versión a la siguiente. Este suele ser el caso de los archivos de código fuente. Por otro lado, cuando se le dice a CVS que almacene un archivo como binario, mantendrá cada versión individual en el servidor. Almacenar archivos como binarios es importante para evitar la corrupción de archivos binarios.

En el mundo del software de código abierto, el Concurrent Version System (CVS) ha sido durante mucho tiempo la herramienta de elección para el control de versiones. Y con razón. CVS en sí mismo es software libre, y su *modus operandi* no restrictivo y soporte para el funcionamiento en red, que permite a docenas de programadores geográficamente dispersos compartir su trabajo, encaja muy bien con la naturaleza colaborativa del mundo de código abierto. CVS y su modelo de desarrollo semi-caótico se han convertido en piedras angulares de código abierto.

-  Collins-Sussman, Control de versiones con Subversion para Subversion 1.1, 2005

1990 **Rational Synergy**  es una herramienta de software que proporciona capacidades de [gestión de configuración de software](https://en.wikipedia.org/wiki/Software_configuration_management) (SCM) para todos los artefactos relacionados con el desarrollo de software, incluidos el [código fuente](https://en.wikipedia.org/wiki/Source_code) , los documentos y las imágenes, así como las bibliotecas ejecutables y el software final. Rational Synergy también proporciona el repositorio de la herramienta de [administración de cambios](https://en.wikipedia.org/wiki/Change_management) conocida como Cambio Racional. Juntas, estas dos herramientas forman un entorno de [gestión de configuración](https://en.wikipedia.org/wiki/Configuration_management) y [gestión de configuración](https://en.wikipedia.org/wiki/Configuration_management) integrado que se utiliza en organizaciones de desarrollo de software que necesitan procesos controlados de SCM y una comprensión de lo que hay en una compilación de su software.

El nombre *Synergy se* refiere a su integración a nivel de base de datos con Change Management que proporciona vistas de lo que está en una construcción en términos de defectos.

|  |  |
| --- | --- |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | CaseWare, Inc. |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Software racional](https://en.wikipedia.org/wiki/Rational_Software) |
| **Versión inicial** | 1990 ; Hace 28 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 7.2.1.5 [[1]](https://en.wikipedia.org/wiki/Rational_Synergy#cite_note-1) / 23 de marzo de 2016 ; Hace 21 meses |
| **Escrito en** | Acento, [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [AIX](https://en.wikipedia.org/wiki/IBM_AIX" \o "IBM AIX) , [HP-UX](https://en.wikipedia.org/wiki/HP-UX" \o "HP-UX) , [Linux](https://en.wikipedia.org/wiki/Linux" \o "Linux) ,[Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)" \o "Solaris (sistema operativo)) , [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows" \o "Microsoft Windows) |
| **[Tipo](https://en.wikipedia.org/wiki/Software_categories" \l "Broad_categories" \o "Categorías de software)** | [Gestión de configuración de software](https://en.wikipedia.org/wiki/Software_configuration_management) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | IBM [EULA](https://en.wikipedia.org/wiki/Software_license) |
| **Sitio web** | [www .ibm .com/ software / productos / es / ratisyne](http://www.ibm.com/software/products/en/ratisyne) |

1990 **Sun WorkShop TeamWare** más tarde **Forte TeamWare** , luego **Forte Code Management Software** ) es un sistema distribuido de [control de revisión de](https://en.wikipedia.org/wiki/Revision_control)[código fuente](https://en.wikipedia.org/wiki/Source_code) creado por [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems" \o "Sun Microsystems) . Último disponible como parte del producto [Forte Developer 6 actualización 2](https://en.wikipedia.org/wiki/Oracle_Solaris_Studio) , TeamWare ya no se ofrece para la venta, [[1]](https://en.wikipedia.org/wiki/Sun_WorkShop_TeamWare#cite_note-1) y no es parte del producto [Sun Studio](https://en.wikipedia.org/wiki/Sun_Studio_(software)" \o "Sun Studio (software)).

La implementación más grande de TeamWare se realizó dentro de Sun, donde (salvo algunas excepciones) en un momento dado fue el único [VCS](https://en.wikipedia.org/wiki/Version_Control_System) utilizado. TeamWare se utilizó para gestionar los árboles fuente más grandes de Sun, incluidos los de [Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)) y [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) , pero como parte del proceso de conversión de esas bases de [código a](https://en.wikipedia.org/wiki/Open_source) comunidades de [código abierto](https://en.wikipedia.org/wiki/Open_source) , se trasladaron a sistemas de control de revisiones más nuevos como [Mercurial](https://en.wikipedia.org/wiki/Mercurial_(software)) .

TeamWare presenta una serie de características avanzadas que no se encuentran en sistemas de control de versiones anteriores como [RCS](https://en.wikipedia.org/wiki/Revision_Control_System) y [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) . En particular, presenta una jerarquía de repositorios de origen y permite actualizaciones atómicas de múltiples archivos, características que se encuentran en sistemas de control de versiones posteriores, como [Subversion](https://en.wikipedia.org/wiki/Subversion_(software)" \o "Subversion (software)) y [Perforce](https://en.wikipedia.org/wiki/Perforce" \o "Forzosamente) . TeamWare permite el [desarrollo distribuido](https://en.wikipedia.org/wiki/Distributed_development) copiando un repositorio a otro que podría residir en otra máquina o red. Los desarrolladores pueden luego realizar cambios en la copia local del repositorio, integrando periódicamente los cambios acumulados en el repositorio local en el repositorio original.

TeamWare se implementa como una capa sobre el [SCCS](https://en.wikipedia.org/wiki/Source_Code_Control_System) anterior , que se utiliza para rastrear los cambios en los archivos individuales. TeamWare solo funciona mediante un sistema de archivos a los que acceden [los programas cliente](https://en.wikipedia.org/wiki/Client_(computing)) (que interactúan sin un [servidor](https://en.wikipedia.org/wiki/Server_(computing)) ) y la mayoría de los usuarios distribuidos de un repositorio acceden a él mediante un [sistema de archivos en red](https://en.wikipedia.org/wiki/Distributed_file_system) montado como [NFS](https://en.wikipedia.org/wiki/Network_File_System_(protocol)) .

Evan Adams fue el líder arquitectónico de TeamWare. [[2]](https://en.wikipedia.org/wiki/Sun_WorkShop_TeamWare#cite_note-2) Glenn Skinner fue el inventor de *smoosh*[[3]](https://en.wikipedia.org/wiki/Sun_WorkShop_TeamWare#cite_note-3)[se[*necesita una fuente mejor*](https://en.wikipedia.org/wiki/Wikipedia:NOTRS)] [[*discutir*](https://en.wikipedia.org/wiki/Talk:Sun_WorkShop_TeamWare#smoosh?)] y [Larry McVoy](https://en.wikipedia.org/wiki/Larry_McVoy) escribió *smoosh* , una herramienta para fusionar archivos SCCS, [[4]](https://en.wikipedia.org/wiki/Sun_WorkShop_TeamWare#cite_note-4) que se dice que es un precursor de TeamWare. [[5]](https://en.wikipedia.org/wiki/Sun_WorkShop_TeamWare#cite_note-5) El sistema de control de versiones [BitKeeper](https://en.wikipedia.org/wiki/BitKeeper" \o "BitKeeper) , diseñado por McVoy, comparte una serie de conceptos de diseño con TeamWare anterior.

1991 **QVCS** (Quma Version Control System) es una familia de productos de [sistemas de control de versiones](https://en.wikipedia.org/wiki/Version_control_system) históricamente publicados por Quma Software, Inc.

QVCS se publicó por primera vez en 1991 como un conjunto de utilidades de línea de comandos para [Amiga](https://en.wikipedia.org/wiki/Amiga) .

Quma llevó ese producto original de Amiga a la plataforma de [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) en 1996. En 2000, se introdujo QVCS-Pro con características adicionales, incluida la compatibilidad con [entornos de desarrollo integrados que](https://en.wikipedia.org/wiki/Integrated_development_environment)[cumplen](https://en.wikipedia.org/wiki/SCC_compliant) con la norma [SCC de](https://en.wikipedia.org/wiki/SCC_compliant)Microsoft . [[4]](https://en.wikipedia.org/wiki/QVCS#cite_note-4) En 2004, se introdujo el producto QVCS-Enterprise [multiplataforma](https://en.wikipedia.org/wiki/Cross-platform) . [[5]](https://en.wikipedia.org/wiki/QVCS#cite_note-5) QVCS y QVCS-Pro están escritos en [C ++](https://en.wikipedia.org/wiki/C%2B%2B) . QVCS-Enterprise está escrito en [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) .

El 1 de enero de 2012, QVCS-Pro y QVCS se convirtieron en software gratuito. QVCS-Enterprise hizo lo mismo el 16 de junio de 2012, y luego fue abierta bajo la licencia Apache 2.0 el 2 de marzo de 2014.

En algún momento después del 23 de junio de 2013, el nombre de dominio qumasoft.com fue abandonado. Ahora (junio de 2015) es propiedad de una compañía de pérdida de peso. El contenido del sitio web cuando fue operado por el autor de QVCS está disponible en [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine" \o "Wayback Machine) .

|  |  |
| --- | --- |
| **QVCS** | |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | Jim Voris |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | Software Quma |
| **Versión inicial** | 1991 ; Hace 27 años |
| [**Último lanzamiento**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 3.10.20 [[1]](https://en.wikipedia.org/wiki/QVCS#cite_note-1) / 5 de abril de 2010 ; hace 7 años |
| **Estado del desarrollo** | sitio web abandonado[[2]](https://en.wikipedia.org/wiki/QVCS#cite_note-2) |
| **Escrito en** | [C ++](https://en.wikipedia.org/wiki/C%2B%2B) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | Freeware |

QVCS y QVCS-Pro se dirigen a equipos de desarrollo de Windows más pequeños. QVCS-Enterprise es una aplicación [cliente-servidor](https://en.wikipedia.org/wiki/Client%E2%80%93server), es multiplataforma y está diseñada para ser utilizada por equipos de desarrollo distribuidos.

**1992 Rational ClearCase** es una familia de herramientas de [software](https://en.wikipedia.org/wiki/Computer_software) que admite [la administración de configuración de software](https://en.wikipedia.org/wiki/Software_configuration_management) (SCM) del [código fuente](https://en.wikipedia.org/wiki/Source_code) y otros activos de [desarrollo de software](https://en.wikipedia.org/wiki/Software_development) . También es compatible con la gestión de datos de diseño de artefactos de diseño electrónico, lo que permite el codesarrollo de hardware y software. ClearCase incluye [control de revisiones](https://en.wikipedia.org/wiki/Revision_control) y forma la base para la administración de la configuración en grandes y medianas empresas, acomodando proyectos con cientos o miles de desarrolladores. Está desarrollado por [IBM](https://en.wikipedia.org/wiki/IBM) .

ClearCase admite dos modelos de gestión de configuración: [UCM](https://en.wikipedia.org/wiki/Rational_ClearCase_UCM) (Unified Change Management) y [ClearCase base](https://en.wikipedia.org/w/index.php?title=Base_ClearCase&action=edit&redlink=1" \o "Base ClearCase (la página no existe)) . UCM proporciona un modelo listo para usar mientras que ClearCase base proporciona una infraestructura básica (UCM se basa en ClearCase base). Ambos se pueden configurar para admitir una amplia variedad de necesidades.

ClearCase puede acomodar grandes archivos binarios, grandes cantidades de archivos y grandes tamaños de repositorio. Admite ramificación, etiquetado y control de versiones de directorios.

El sistema de [base de datos](https://en.wikipedia.org/wiki/Database) que utiliza ClearCase es [RDM Embedded](https://en.wikipedia.org/wiki/RDM_Embedded) from [Raima](https://en.wikipedia.org/wiki/Raima) . En la terminología ClearCase, una base de datos individual se llama un *VOB* ( **V** ersioned **O**bject **B** ASE). En esta capa, el mantenimiento se lleva a cabo utilizando herramientas de Raima. Alrededor de esta capa, un conjunto de interfaces con las herramientas que lo acompañan se utilizan para administrar el sistema de base de datos física, que requiere habilidades específicas de [administrador de la base de datos](https://en.wikipedia.org/wiki/Database_administrator) .

El servicio más importante es la ubicación de Atria Broker Daemon (ALBD), que gestiona todas las comunicaciones (LAN) entre computadoras. A partir de la versión 7, la plataforma del servidor ejecuta Websphere Application Server con una aplicación de servidor llamada Change Management Server (CM Server), que presta servicios a los clientes de ClearCase a través del protocolo HTTP. (Antes de la versión 7 existía un servicio web mediante el cual los usuarios podían acceder a ClearCase a través de sus navegadores.) Desde entonces, el servidor CM ha sido reemplazado por el servidor de red de área amplia ClearCase Remote Client (servidor WAN CCRC), que sigue basándose en la aplicación Websphere. Servidor.

|  |  |
| --- | --- |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Atria Software](https://en.wikipedia.org/wiki/Atria_Software) |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [IBM](https://en.wikipedia.org/wiki/IBM) |
| **Versión inicial** | 1992 ; Hace 26 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 9.0.1.2 [[1]](https://en.wikipedia.org/wiki/Rational_ClearCase#cite_note-1) / 18 de diciembre de 2017 ; Hace 37 días |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [AIX](https://en.wikipedia.org/wiki/IBM_AIX) , [HP-UX](https://en.wikipedia.org/wiki/HP-UX) , [Linux](https://en.wikipedia.org/wiki/Linux) ,[Linux en z Systems](https://en.wikipedia.org/wiki/Linux_on_z_Systems) ,[Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)) , [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) , [z / OS](https://en.wikipedia.org/wiki/Z/OS) (cliente limitado) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Gestión de configuración de software](https://en.wikipedia.org/wiki/Software_configuration_management) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | IBM [EULA](https://en.wikipedia.org/wiki/Software_license) |
| **Sitio web** | [www .ibm .com/ software / productos / en / clearcase](http://www.ibm.com/software/products/en/clearcase) |

1994 **Configuration Management Version Control (CMVC)** es un paquete de [software](https://en.wikipedia.org/wiki/Software) que sirve como depósito de objetos y realiza [control de versiones de](https://en.wikipedia.org/wiki/Version_control) software , administración de configuraciones y funciones de administración de cambios.

|  |  |
| --- | --- |
| **Configuration Management** | |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [IBM](https://en.wikipedia.org/wiki/IBM) |
| **Estado del desarrollo** | Inactivo |
| **Escrito en** | [C ++](https://en.wikipedia.org/wiki/C%2B%2B) , [C](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | Linux, Mac, Windows, z / OS, AIX ([multiplataforma](https://en.wikipedia.org/wiki/Cross-platform) ) |
| **Disponible en** | Plurilingüe |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Desarrollo de software](https://en.wikipedia.org/wiki/Software_development) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Propiedad](https://en.wikipedia.org/wiki/Proprietary_software) |
| **Sitio web** | [www .ibm .com](http://www.ibm.com/) |

* El seguimiento de defectos se integró en el sistema de control de origen.
* Posibilidad de configurar un repositorio para exigir que se asocien defectos con cada registro de archivo.
* Los niveles se utilizan para gestionar conjuntos de defectos, lo que permite a los administradores pensar en términos de conjuntos de cambios en lugar de conjuntos de archivos.
* La administración distribuida permite la delegación de autoridad. Esto es particularmente útil en equipos grandes.
* Tiene la capacidad de rastrear historiales de archivos incluso después de cambiar el nombre. Esto se debe a que el nombre del archivo en el disco era un número y el nombre del archivo en la base de datos podría cambiarse.
* Los archivos pueden ser compartidos entre el código fuente

1994 **Microsoft Visual SourceSafe** (**VSS**) es un [programa de](https://en.wikipedia.org/wiki/Computer_program)[control de fuente](https://en.wikipedia.org/wiki/Source_control) descontinuado , orientado a pequeños proyectos de desarrollo de software. Como la mayoría de los sistemas de control de fuente, SourceSafe crea una *biblioteca virtual* de archivos de computadora. Aunque se usa más comúnmente para el [código fuente](https://en.wikipedia.org/wiki/Source_code) , SourceSafe puede manejar cualquier tipo de archivo en su [base de datos](https://en.wikipedia.org/wiki/Database) , pero las versiones anteriores se mostraron inestables [[1]](https://en.wikipedia.org/wiki/Microsoft_Visual_SourceSafe#cite_note-1)[[2]](https://en.wikipedia.org/wiki/Microsoft_Visual_SourceSafe#cite_note-2) cuando se usaban para almacenar grandes cantidades de datos no textuales, como imágenes, y compilaban ejecutables.

|  |  |
| --- | --- |
| **Visual SourceSafe** | |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Microsoft](https://en.wikipedia.org/wiki/Microsoft) |
| **Versión inicial** | 1994 ; Hace 24 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 2005 (8.0.50727.42) / octubre de 2005 ; Hace 12 años |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Software comercial](https://en.wikipedia.org/wiki/Commercial_software)[propietario](https://en.wikipedia.org/wiki/Proprietary_software) |
| **Sitio web** | [www .microsoft .com](http://www.microsoft.com/) |

Originalmente, SourceSafe no era una [gestión de código fuente](https://en.wikipedia.org/wiki/Source_Code_Management) cliente / servidor , sino más bien un sistema SCM *local* único. Arquitectónicamente, esto sirve como una fortaleza y una debilidad del diseño, dependiendo del entorno en el que se utiliza. Permite configurar un solo sistema de usuario con menos configuración que la de otros sistemas [SCM](https://en.wikipedia.org/wiki/Source_Code_Management) . Además, el proceso de copia de seguridad puede ser tan simple como copiar todos los contenidos de un único árbol de directorios. Para [entornos multiusuario](https://en.wikipedia.org/wiki/Multi-user) , sin embargo, carece de muchas características importantes que se encuentran en otros productos [SCM](https://en.wikipedia.org/wiki/Source_Code_Management) , incluido el soporte para [confirmaciones atómicas](https://en.wikipedia.org/wiki/Atomic_commit) de múltiples archivos ( [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) tiene el mismo problema que el original).[RCS](https://en.wikipedia.org/wiki/Revision_Control_System) ). SourceSafe hereda su funcionalidad *compartida* mediante el acceso directo al sistema de archivos remoto a todos los archivos en el repositorio. Esto, junto con errores históricos en la base de código, ocasionalmente condujo a la corrupción de la base de datos de SourceSafe, un problema observado por Microsoft. [[9]](https://en.wikipedia.org/wiki/Microsoft_Visual_SourceSafe#cite_note-9)

A partir de VSS 2005, Microsoft agregó un modo [cliente-servidor](https://en.wikipedia.org/wiki/Client%E2%80%93server) . En este modo, los clientes no necesitan [acceso de escritura](https://en.wikipedia.org/wiki/Write_access) a un [recurso](https://en.wikipedia.org/wiki/Server_Message_Block) compartido [SMB](https://en.wikipedia.org/wiki/Server_Message_Block) donde pueden dañar la base de datos SS. En su lugar, se debe acceder a los archivos a través de las herramientas del cliente de VSS: el cliente de Windows VSS, la herramienta de [línea de comandos de](https://en.wikipedia.org/wiki/Command-line)VSS o alguna aplicación que se integre o emule con estas herramientas de cliente.

1995 **Perforce Software** es un desarrollador de software basado en [Minneapolis, Minnesota,](https://en.wikipedia.org/wiki/Minneapolis,_Minnesota) utilizado para el desarrollo de aplicaciones, que incluye software de [control](https://en.wikipedia.org/wiki/Version_control) de versiones, gestión de repositorios basada en la web, colaboración de desarrolladores, [gestión del ciclo de vida de las aplicaciones](https://en.wikipedia.org/wiki/Application_lifecycle_management) y software de [planificación ágil](https://en.wikipedia.org/wiki/Agile_software_development) . El software se vende bajo las marcas Helix y Hansoft. [[2]](https://en.wikipedia.org/wiki/Perforce_Software#cite_note-pocketgamer-2)

En enero de 2018, la empresa fue adquirida por la firma de capital privado Clearlake Capital.

|  |  |
| --- | --- |
| **Perforce Software** | |
| [Perforce-Software-Logo.png](https://en.wikipedia.org/wiki/File:Perforce-Software-Logo.png) | |
| [**Tipo**](https://en.wikipedia.org/wiki/List_of_business_entities) | Privado |
| **Industria** | [El control de versiones](https://en.wikipedia.org/wiki/Version_control) de software, [gestión del ciclo de vida de aplicaciones](https://en.wikipedia.org/wiki/Application_lifecycle_management) , [Git](https://en.wikipedia.org/wiki/Git" \o "Git)administración de repositorios, herramientas de colaboración desarrollador y [la planificación ágil](https://en.wikipedia.org/wiki/Agile_software_development) de software |
| **Fundado** | 1995 |
| **Sede** | [Minneapolis, MN](https://en.wikipedia.org/wiki/Minneapolis,_MN) |
| **Área servida** | En todo el mundo |
| **Gente clave** | Janet Dryer, CEO |
| **Productos** | Helix Core, Helix TeamHub, Helix Swarm, suite Helix ALM, Hansoft |
| **Número de empleados** | 250+ (2014) [[1]](https://en.wikipedia.org/wiki/Perforce_Software#cite_note-ashimel-1) |
| [**Padre**](https://en.wikipedia.org/wiki/Parent_company) | Clearlake Capital |
| **Sitio web** | [www .perforce .com](http://www.perforce.com/) |

**1995 StarTeam** es un sistema de [control de revisiones](https://en.wikipedia.org/wiki/Revision_control) utilizado en el desarrollo de software, especialmente cuando un proyecto involucra múltiples equipos en diferentes ubicaciones. StarTeam es una aplicación de software [SCM](https://en.wikipedia.org/wiki/Source_Code_Management) y [SDLC](https://en.wikipedia.org/wiki/Software_development_lifecycle) , creada por Starbase Corporation, que fue adquirida por [Borland](https://en.wikipedia.org/wiki/Borland" \o "Borland) en enero de 2003 [[1]](https://en.wikipedia.org/wiki/StarTeam#cite_note-1) que fue adquirida por [Micro Focus](https://en.wikipedia.org/wiki/Micro_Focus_International) mientras tanto. La aplicación es cliente-servidor, respaldada por una base de datos relacional que conserva todos los cambios realizados en un proyecto durante su evolución, así como los requisitos del proyecto, las asignaciones de tareas, las discusiones y el seguimiento de errores. [Microsoft SQL Server](https://en.wikipedia.org/wiki/Microsoft_SQL_Server) y la [base de datos Oracle](https://en.wikipedia.org/wiki/Oracle_database) son servidores de bases de datos compatibles.

|  |  |
| --- | --- |
| **Borland StarTeam** | |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | Base estelar, [Borland](https://en.wikipedia.org/wiki/Borland" \o "Borland) |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Micro Focus International](https://en.wikipedia.org/wiki/Micro_Focus_International) |
| **Versión inicial** | 1995 ; Hace 23 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 16.2 / 20 de noviembre de 2017 ; Hace 2 meses |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Microsoft Windows 10](https://en.wikipedia.org/wiki/Microsoft_Windows_10" \o "Microsoft Windows 10) , [Microsoft Windows 8](https://en.wikipedia.org/wiki/Microsoft_Windows_8" \o "Microsoft Windows 8) ,[Microsoft Windows 7](https://en.wikipedia.org/wiki/Microsoft_Windows_7" \o "Microsoft Windows 7), [Windows XP Professional](https://en.wikipedia.org/wiki/Windows_XP_Professional" \o "Windows XP Professional) SP3,[Windows Vista](https://en.wikipedia.org/wiki/Windows_Vista" \o "Windows Vista) SP1,[Red Hat Enterprise Linux](https://en.wikipedia.org/wiki/Red_Hat_Enterprise_Linux" \o "Red Hat Enterprise Linux) (WS) 5.1 |
| **Disponible en** | Plurilingüe |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [SCM](https://en.wikipedia.org/wiki/Software_configuration_management) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | Opciones de licencia de usuario final denominadas y concurrentes |
| **Sitio web** | [www .microfocus.com / products / change-management/ starteam /](http://www.microfocus.com/products/change-management/starteam/) |

1997 **Code Co-op** es el sistema de [control de revisión](https://en.wikipedia.org/wiki/Revision_control)[punto a punto](https://en.wikipedia.org/wiki/Peer-to-peer) creado por [Reliable Software](https://en.wikipedia.org/w/index.php?title=Reliable_Software&action=edit&redlink=1" \o "Reliable Software (page does not exist)) .

Code Co-op es un sistema de [control de revisión](https://en.wikipedia.org/wiki/Revision_control) distribuido del tipo replicado.

Utiliza [la](https://en.wikipedia.org/wiki/Peer-to-peer) arquitectura [punto a punto](https://en.wikipedia.org/wiki/Peer-to-peer) para compartir proyectos entre desarrolladores y controlar los cambios en los archivos. En lugar de utilizar una [base de datos centralizada](https://en.wikipedia.org/wiki/Centralized_database) (el repositorio), replica su propia base de datos en cada computadora involucrada en el proyecto.

Las réplicas se sincronizan mediante el intercambio de *scripts* (diferenciales) . El intercambio de scripts puede continuar usando diferentes transportes, incluidos el correo electrónico (soporte para [SMTP](https://en.wikipedia.org/wiki/SMTP) y [POP3](https://en.wikipedia.org/wiki/POP3) , integración con clientes [MAPI](https://en.wikipedia.org/wiki/MAPI) , [Gmail](https://en.wikipedia.org/wiki/Gmail" \o "Gmail) ) y LAN.

Code Co-op tiene incorporado un sistema [wiki punto a punto](https://en.wikipedia.org/wiki/Peer-to-peer_wiki) , que se puede usar para integrar la documentación con un proyecto de software. También es posible crear [bases de datos Wiki](https://en.wikipedia.org/w/index.php?title=Wiki_database&action=edit&redlink=1) basadas en texto , que pueden consultarse utilizando [SQL](https://en.wikipedia.org/wiki/SQL) simplificado directamente desde páginas wiki.

|  |  |
| --- | --- |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | Software confiable |
| **Versión inicial** | 1996 ; Hace 22 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 5.2a / 4 de junio de 2011 ; Hace 6 años [[1]](https://en.wikipedia.org/wiki/Code_Co-op#cite_note-1) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) |
| **Disponible en** | [Inglés](https://en.wikipedia.org/wiki/English_language) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | Propiedad |
| **Sitio web** | [relisoft .com / co\_op /](http://relisoft.com/co_op/) |

1. Soporte de desarrollo distribuido a través [de correo electrónico](https://en.wikipedia.org/wiki/E-mail) , [LAN](https://en.wikipedia.org/wiki/Local_Area_Network) o [VPN](https://en.wikipedia.org/wiki/VPN)
2. Modificaciones basadas en el modelo: las modificaciones en varios archivos se registran como una [transacción](https://en.wikipedia.org/wiki/Database_transaction)
3. Las adiciones, eliminaciones, renombrados y movimientos de archivos se tratan en el mismo nivel que las ediciones; se pueden agregar en cualquier combinación a un conjunto de cambios de check-in.
4. Los cambios de archivos pueden revisarse antes de un check-in utilizando un built-in o definido por el usuario
5. Los cambios de sincronización pueden ser revisados ​​de la misma manera por los destinatarios
6. Fusión visual de tres vías
7. El historial del proyecto se replica en cada máquina. La versión histórica puede ser revisada, comparada o restaurada
8. Integración con clientes Microsoft SCC, incluido [Visual Studio](https://en.wikipedia.org/wiki/Visual_Studio)

1998 El **CVSNT** Sistema de control de versiones implementa un [sistema de control de versiones](https://en.wikipedia.org/wiki/Version_control_system) : se realiza un seguimiento de todos los cambios en un conjunto de archivos, por lo general la implementación de un [software de](https://en.wikipedia.org/wiki/Software)[proyecto](https://en.wikipedia.org/wiki/Project) , y permite que varios (potencialmente) separadas geográficamente a los desarrolladores [colaboran](https://en.wikipedia.org/wiki/Collaboration) . Es compatible y originalmente se basa en el [Sistema de Versiones Concurrentes](https://en.wikipedia.org/wiki/Concurrent_Versions_System) (CVS), que se ha hecho popular en el mundo de [código abierto](https://en.wikipedia.org/wiki/Open-source) .

CVSNT realiza un seguimiento del historial de versiones de un proyecto (o conjunto de archivos).

CVSNT se basa en la misma arquitectura [cliente-servidor](https://en.wikipedia.org/wiki/Client%E2%80%93server) que el [Sistema de Versiones Simultáneas](https://en.wikipedia.org/wiki/Concurrent_Versions_System) : un servidor almacena la (s) versión (es) actual (es) del proyecto y su historial, y los clientes se conectan al servidor para verificar una copia completa del proyecto. , trabaje en esta copia y luego registre sus cambios. Un servidor puede ser un servidor caché o proxy (un servidor de solo lectura que transfiere solicitudes de escritura a otro servidor) o un servidor de lectura y escritura (normal).

Normalmente, el cliente y el servidor se conectan a través de una [WAN](https://en.wikipedia.org/wiki/Wide_area_network) o [Internet](https://en.wikipedia.org/wiki/Internet) , pero tanto el cliente como el servidor pueden ejecutarse en la misma [LAN](https://en.wikipedia.org/wiki/Local_area_network) o incluso en la misma máquina. El software del servidor funciona igual de bien en [Unix](https://en.wikipedia.org/wiki/Unix) y [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) . Es compatible con los clientes de CVS y es compatible con una gran cantidad de metodologías de [gestión](https://en.wikipedia.org/wiki/Configuration_management) de [configuración,](https://en.wikipedia.org/wiki/Configuration_management) incluido el método sin reservas compatible con CVS y el método reservado similar a [VSS](https://en.wikipedia.org/wiki/Microsoft_Visual_SourceSafe) . [[2]](https://en.wikipedia.org/wiki/CVSNT#cite_note-2)[[3]](https://en.wikipedia.org/wiki/CVSNT#cite_note-3)

La naturaleza modular de CVSNT permite a los administradores del servidor habilitar y deshabilitar protocolos de forma dinámica, así como cargar funciones adicionales a través de activadores como Auditoría, Integración de seguimiento de defectos, Automatic Checkout (para servidores web) y más.

Las características del servidor CVSNT incluyen: [[4]](https://en.wikipedia.org/wiki/CVSNT#cite_note-4)

* Control de acceso para asegurar proyectos y sucursales. [[5]](https://en.wikipedia.org/wiki/CVSNT#cite_note-5)
* Auditoría detallada y métricas registradas en una base de datos [SQL](https://en.wikipedia.org/wiki/SQL) . [[6]](https://en.wikipedia.org/wiki/CVSNT#cite_note-6)[[7]](https://en.wikipedia.org/wiki/CVSNT#cite_note-7)[[8]](https://en.wikipedia.org/wiki/CVSNT#cite_note-8)
* Autenticación con [Active Directory](https://en.wikipedia.org/wiki/Active_Directory) . [[9]](https://en.wikipedia.org/wiki/CVSNT#cite_note-9)
* El seguimiento de todo el cambio, incluso si se fusionó desde algún lugar, pertenece a un informe de problema o formaba parte de un conjunto de cambios.
* Un panel de control para administrar la notificación por correo electrónico de los cambios, la integración de seguimiento de defectos y más.
* Sincronización de repositorio integrada (para servidores de conmutación por error).
* Cambiar soporte de ajuste (cambios de grupo por número de defecto).
* Admite archivos Unicode [UTF-8](https://en.wikipedia.org/wiki/UTF-8) / [UCS-2](https://en.wikipedia.org/wiki/UCS-2) y nombres de archivos multilingües.
* Cuando se opera en modo UTF-8 (Unicode), puede traducir automáticamente la codificación del nombre de archivo para cualquier cliente.
* Complementos para notificaciones por correo electrónico.
* Ayuda a facilitar la fusión de sucursales con su función "Mergepoint".
* Servidores nativos disponibles para Mac OS X, Windows, Linux, Solaris, HP-UX.
* Admite metodologías de versiones reservadas y sin reservas.
* CVSAPI para la integración en productos de terceros.
* Interfaz Script, COM y 3GL para desencadenantes e integración en herramientas de terceros (como seguimiento de defectos)

1998 **QVCS Enterprise** (Quma Version Control System) es una familia de productos de [sistemas de control de versiones](https://en.wikipedia.org/wiki/Version_control_system) históricamente publicados por Quma Software, Inc.

QVCS se publicó por primera vez en 1991 como un conjunto de utilidades de línea de comandos para [Amiga](https://en.wikipedia.org/wiki/Amiga) .

Quma llevó ese producto original de Amiga a la plataforma de [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) en 1996. En 2000, se introdujo QVCS-Pro con características adicionales, incluida la compatibilidad con [entornos de desarrollo integrados que](https://en.wikipedia.org/wiki/Integrated_development_environment)[cumplen](https://en.wikipedia.org/wiki/SCC_compliant) con la norma [SCC de](https://en.wikipedia.org/wiki/SCC_compliant)Microsoft . [[4]](https://en.wikipedia.org/wiki/QVCS#cite_note-4) En 2004, se introdujo el producto QVCS-Enterprise [multiplataforma](https://en.wikipedia.org/wiki/Cross-platform) . [[5]](https://en.wikipedia.org/wiki/QVCS#cite_note-5) QVCS y QVCS-Pro están escritos en [C ++](https://en.wikipedia.org/wiki/C%2B%2B) . QVCS-Enterprise está escrito en [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) .

El 1 de enero de 2012, QVCS-Pro y QVCS se convirtieron en software gratuito. QVCS-Enterprise hizo lo mismo el 16 de junio de 2012, y luego fue abierta bajo la licencia Apache 2.0 el 2 de marzo de 2014.

En algún momento después del 23 de junio de 2013, el nombre de dominio qumasoft.com fue abandonado. Ahora (junio de 2015) es propiedad de una compañía de pérdida de peso. El contenido del sitio web cuando fue operado por el autor de QVCS está disponible en [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine" \o "Wayback Machine) .

|  |  |
| --- | --- |
| **QVCS Enterprise** | |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | Jim Voris |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | Software Quma |
| **Versión inicial** | 1998 ; hace 20 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 3.0.9 [[3]](https://en.wikipedia.org/wiki/QVCS#cite_note-3) / 14 de abril de 2014 ; hace 3 años |
| **Escrito en** | [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) , [OS X](https://en.wikipedia.org/wiki/OS_X) ,[Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Licencia Apache 2.0](https://en.wikipedia.org/wiki/Apache_2.0_License) |
| **Sitio web** | [github .com / jimv39 / qvcsos](https://github.com/jimv39/qvcsos) |

2000 **BitKeeper** es una herramienta de [software](https://en.wikipedia.org/wiki/Software) para [control](https://en.wikipedia.org/wiki/Distributed_revision_control) de [revisión distribuida](https://en.wikipedia.org/wiki/Distributed_revision_control) del [código fuente](https://en.wikipedia.org/wiki/Source_code) de la computadora . Originalmente, [el software propietario](https://en.wikipedia.org/wiki/Proprietary_software) , fue lanzado como [software de código abierto](https://en.wikipedia.org/wiki/Open-source_software) bajo la [licencia Apache 2.0](https://en.wikipedia.org/wiki/Apache_License#Version_2.0) el 9 de mayo de 2016. [[2]](https://en.wikipedia.org/wiki/BitKeeper#cite_note-2) BitKeeper es producido por BitMover Inc., una empresa privada con sede en [Los Gatos, California](https://en.wikipedia.org/wiki/Los_Gatos,_California)[[3]](https://en.wikipedia.org/wiki/BitKeeper#cite_note-3) y propiedad de su [CEO](https://en.wikipedia.org/wiki/CEO) , [Larry McVoy](https://en.wikipedia.org/wiki/Larry_McVoy) , quien previamente había diseñado [TeamWare](https://en.wikipedia.org/wiki/TeamWare" \o "TeamWare) .

|  |  |
| --- | --- |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | BitMover Inc. |
| **Versión inicial** | 4 de mayo de 2000 ; Hace 17 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 7.3.2 / 23 de septiembre de 2017 ; Hace 3 meses [[1]](https://en.wikipedia.org/wiki/BitKeeper#cite_note-1) |
| [**Repositorio**](https://en.wikipedia.org/wiki/Repository_(version_control)) | * <http://repos.bkbits.net/bk/dev/>   [Edita esto en Wikidata](https://www.wikidata.org/wiki/Q878697#P1324) |
| **Escrito en** | [do](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | AIX, FreeBSD, HP-UX, IRIX, Linux, Mac OS X, NetBSD, OpenBSD, Solaris, Windows |
| **[Tipo](https://en.wikipedia.org/wiki/Software_categories" \l "Broad_categories" \o "Categorías de software)** | [Control de revisión distribuido](https://en.wikipedia.org/wiki/Distributed_revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Apache 2.0](https://en.wikipedia.org/wiki/Apache_License#Version_2.0) |
| **Sitio web** | [www .bitkeeper .org](http://www.bitkeeper.org/) |

BitKeeper se mencionó por primera vez como una solución a algunos de los dolores de crecimiento que Linux estaba teniendo en septiembre de 1998. [[4]](https://en.wikipedia.org/wiki/BitKeeper#cite_note-4) Betas de acceso temprano estaban disponibles en mayo de 1999 [[5]](https://en.wikipedia.org/wiki/BitKeeper#cite_note-5) y el 4 de mayo de 2000 se puso a disposición el primer lanzamiento público de BitKeeper . [[6]](https://en.wikipedia.org/wiki/BitKeeper#cite_note-6)[[7]](https://en.wikipedia.org/wiki/BitKeeper#cite_note-7) BitMover solía proporcionar acceso al sistema para ciertos [proyectos de código abierto](https://en.wikipedia.org/wiki/Open_source) o [software libre](https://en.wikipedia.org/wiki/Free_software) , uno de los cuales era el código fuente del [kernel de Linux](https://en.wikipedia.org/wiki/Linux_kernel" \o "Kernel de Linux) . La licencia para la versión "comunitaria" de BitKeeper había permitido a los desarrolladores utilizar la herramienta sin costo alguno para proyectos de software libre o de código abierto, siempre que esos desarrolladores no participaran en el desarrollo de una herramienta competidora (como[Concurrent Versions System](https://en.wikipedia.org/wiki/Concurrent_Versions_System) , [GNU Arch](https://en.wikipedia.org/wiki/GNU_Arch) , [Subversion](https://en.wikipedia.org/wiki/Subversion_(software)" \o "Subversion (software)) o [ClearCase](https://en.wikipedia.org/wiki/ClearCase" \o "ClearCase) ) durante el uso de BitKeeper más un año. Esta restricción se aplicó independientemente de si la herramienta competidora era gratuita o patentada. Esta versión de BitKeeper también requería que cierta metainformación sobre cambios se almacenara en servidores de computadora operados por BitMover, una adición que imposibilitaba a los usuarios de la versión de la comunidad ejecutar proyectos de los cuales BitMover no era consciente.

2000 **Apache Subversion** (a menudo abreviado **SVN** , después de su nombre de comando *svn* ) es un [sistema de control de](https://en.wikipedia.org/wiki/Software_versioning)[revisiones](https://en.wikipedia.org/wiki/Revision_control) y [versiones de software](https://en.wikipedia.org/wiki/Software_versioning) distribuido como [código abierto](https://en.wikipedia.org/wiki/Open_source) bajo la [licencia de Apache](https://en.wikipedia.org/wiki/Apache_License) . [[2]](https://en.wikipedia.org/wiki/Apache_Subversion#cite_note-2) Los desarrolladores de software usan Subversion para mantener versiones actuales e históricas de archivos tales como [código fuente](https://en.wikipedia.org/wiki/Source_code) , páginas web y documentación. Su objetivo es ser un sucesor casi compatible del ampliamente utilizado [Sistema de Versiones Concurrentes](https://en.wikipedia.org/wiki/Concurrent_Versions_System) (CVS).

La comunidad de [código abierto](https://en.wikipedia.org/wiki/Open_source) ha utilizado ampliamente Subversion: por ejemplo, en proyectos como [Apache Software Foundation](https://en.wikipedia.org/wiki/Apache_Software_Foundation) , [Free Pascal](https://en.wikipedia.org/wiki/Free_Pascal) , [FreeBSD](https://en.wikipedia.org/wiki/FreeBSD" \o "FreeBSD) , [GCC](https://en.wikipedia.org/wiki/GNU_Compiler_Collection) y [SourceForge](https://en.wikipedia.org/wiki/SourceForge" \o "SourceForge) . [CodePlex](https://en.wikipedia.org/wiki/CodePlex" \o "CodePlex) ofrece acceso a Subversion, así como a otros tipos de clientes.

Subversion fue creada por [CollabNet](https://en.wikipedia.org/wiki/CollabNet" \o "CollabNet) Inc. en 2000, y ahora es un proyecto de Apache de alto nivel que está siendo construido y utilizado por una comunidad global de colaboradores.

|  |  |
| --- | --- |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | [CollabNet](https://en.wikipedia.org/wiki/CollabNet) |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Apache Software Foundation](https://en.wikipedia.org/wiki/Apache_Software_Foundation) |
| **Versión inicial** | 20 de octubre de 2000 ; Hace 17 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 1.9.7 (9 de agosto de 2017 ; hace 5 meses [[1]](https://en.wikipedia.org/wiki/Apache_Subversion#cite_note-1) ) [[±]](https://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Subversion&action=edit) |
| [**Lanzamiento de vista previa**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 1.10.0-alpha3 (26 de julio de 2017 ; 6 meses atrás ) [[±]](https://en.wikipedia.org/w/index.php?title=Template:Latest_preview_software_release/Subversion&action=edit) |
| [**Repositorio**](https://en.wikipedia.org/wiki/Repository_(version_control)) | * <https://svn.apache.org/viewvc/subversion/trunk/>   [Edita esto en Wikidata](https://www.wikidata.org/wiki/Q46794#P1324) |
| **Estado del desarrollo** | Activo |
| **Escrito en** | [do](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Multiplataforma](https://en.wikipedia.org/wiki/Cross-platform) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Licencia Apache](https://en.wikipedia.org/wiki/Apache_License) |
| **Sitio web** | [subversion .apache .org](https://subversion.apache.org/) |

* [Se compromete](https://en.wikipedia.org/wiki/Commit_(data_management)) como verdaderas [operaciones atómicas](https://en.wikipedia.org/wiki/Atomicity_(database_systems)) (las [operaciones de](https://en.wikipedia.org/wiki/Atomicity_(database_systems)) confirmación interrumpidas en CVS causarían incoherencia o corrupción del repositorio).
* Los archivos renombrados / copiados / movidos / eliminados retienen el historial completo de revisiones.
* El sistema mantiene el [control](https://en.wikipedia.org/wiki/Software_versioning) de [versiones](https://en.wikipedia.org/wiki/Software_versioning) para directorios, cambios de nombre y [metadatos de](https://en.wikipedia.org/wiki/Metadata) archivos (pero no para marcas de tiempo). Los usuarios pueden mover y / o copiar árboles de directorios completos muy rápidamente, conservando el historial completo de revisiones.
* Control de versiones de [enlaces simbólicos](https://en.wikipedia.org/wiki/Symbolic_link) .
* Soporte nativo para archivos binarios, con almacenamiento binario-diff con ahorro de espacio.
* [Apache HTTP Server](https://en.wikipedia.org/wiki/Apache_HTTP_Server) como servidor de red, [WebDAV](https://en.wikipedia.org/wiki/WebDAV" \o "WebDAV) / [Delta-V](https://en.wikipedia.org/wiki/WebDAV#Extensions_and_derivatives) para [protocolo](https://en.wikipedia.org/wiki/Protocol_(computing)) . También hay un [proceso de](https://en.wikipedia.org/wiki/Process_(computing)) servidor independiente llamado svnserve que usa un protocolo personalizado sobre [TCP / IP](https://en.wikipedia.org/wiki/Internet_Protocol_Suite) .
* [La ramificación](https://en.wikipedia.org/wiki/Branching_(software)) es una operación económica, independiente del tamaño del archivo (aunque Subversion por sí misma no distingue entre una rama y un directorio)
* Nativamente [cliente-servidor](https://en.wikipedia.org/wiki/Client%E2%80%93server_model) , diseño de [biblioteca en](https://en.wikipedia.org/wiki/Library_(computing))[capas](https://en.wikipedia.org/wiki/Abstraction_layer) .
* El protocolo cliente / servidor envía [diffs](https://en.wikipedia.org/wiki/Diff" \o "Diferencia) en ambas direcciones.
* Costos proporcionales al tamaño del cambio, no al tamaño de los datos.
* Salida [procesable](https://en.wikipedia.org/wiki/Parsing) , incluida la salida de registro [XML](https://en.wikipedia.org/wiki/XML) .
* licencia de [fuente abierta](https://en.wikipedia.org/wiki/Open_source) - [Apache License](https://en.wikipedia.org/wiki/Apache_License) desde la versión 1.7; las versiones anteriores usan una derivada de la Licencia de Software Apache 1.1.
* Mensajes de programas [internacionalizados](https://en.wikipedia.org/wiki/Internationalization_and_localization) .
* [Bloqueo](https://en.wikipedia.org/wiki/File_locking) de archivos para archivos no fusionables ("registros reservados").
* Autorización basada en ruta.
* [Enlaces de lenguaje](https://en.wikipedia.org/wiki/Language_binding) para [C #](https://en.wikipedia.org/wiki/C_Sharp_(programming_language)) , [PHP](https://en.wikipedia.org/wiki/PHP) , [Python](https://en.wikipedia.org/wiki/Python_(programming_language)" \o "Python (lenguaje de programación)) , [Perl](https://en.wikipedia.org/wiki/Perl) , [Ruby](https://en.wikipedia.org/wiki/Ruby_(programming_language)) y [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) .
* Soporte completo [MIME](https://en.wikipedia.org/wiki/MIME) : los usuarios pueden ver o cambiar el tipo MIME de cada archivo, y el software sabe qué tipos MIME pueden mostrar diferencias con respecto a las versiones anteriores.
* Seguimiento de fusión: se realizará un seguimiento de las fusiones entre sucursales, esto permite la fusión automática entre las sucursales sin decirle a Subversion qué necesita y qué no necesita fusionarse.
* Listas de cambios para organizar confirmaciones en grupos de compromiso.

2001 **PTC Integrity Lifecycle Manager** (anteriormente *MKS Integrity* ) es una plataforma de administración del ciclo de vida del sistema de software (SSLM) y de [administración del ciclo de vida de la aplicación](https://en.wikipedia.org/wiki/Application_lifecycle_management) (ALM) desarrollada por [MKS Inc.](https://en.wikipedia.org/wiki/MKS_Inc.) y lanzada por primera vez en 2001. El software es cliente / servidor, con ambas computadoras de escritorio ( [Java / swing](https://en.wikipedia.org/wiki/Swing_(java)) ) y las interfaces de cliente web. Proporciona a las organizaciones de desarrollo de software un entorno colaborativo en el que pueden gestionar los procesos de desarrollo de principio a fin, desde la [gestión de requisitos](https://en.wikipedia.org/wiki/Requirements_management) , la [gestión de cambios de ingeniería](https://en.wikipedia.org/wiki/Change_management_(engineering)) , el [control de revisiones](https://en.wikipedia.org/wiki/Revision_control) y la [gestión de compilación](https://en.wikipedia.org/wiki/Build_management) hasta la [gestión](https://en.wikipedia.org/wiki/Build_management) de [pruebas](https://en.wikipedia.org/wiki/Test_management) y[implementación de software](https://en.wikipedia.org/wiki/Software_deployment) , así como informes y métricas asociados.

|  |  |
| --- | --- |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [PTC, Inc.](https://en.wikipedia.org/wiki/PTC,_Inc.) |
| **Versión inicial** | Julio de 2001 ; Hace 16 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 11.1 / 30 de marzo de 2017 ; hace 9 meses |
| **Estado del desarrollo** | Activo |
| **Escrito en** | [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Unix](https://en.wikipedia.org/wiki/Unix-like) , [Windows](https://en.wikipedia.org/wiki/Windows) |
| **Disponible en** | Idiomas internacionales a través de IAN, localizaciones para inglés, alemán, japonés. |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [ALM](https://en.wikipedia.org/wiki/Application_Lifecycle_Management" \o "Gestión del ciclo de vida de la aplicación) , [RM](https://en.wikipedia.org/wiki/Requirements_Management" \o "Gestión de requerimientos) , [SCM](https://en.wikipedia.org/wiki/Software_Configuration_Management" \o "Gestión de configuración de software) ,[Calidad](https://en.wikipedia.org/wiki/Quality_Management" \o "Gestión de la calidad) , [Agile](https://en.wikipedia.org/wiki/Agile_software_development" \o "Desarrollo Ágil de Software) ,[SysEng](https://en.wikipedia.org/wiki/Systems_Engineering" \o "Ingeniería de Sistemas) |
| **[Licencia](https://en.wikipedia.org/wiki/Software_license" \o "Licencia de software)** | PTC [EULA](https://en.wikipedia.org/wiki/Software_license) |
| **Sitio web** | [www .ptc .com / application-lifecycle-management](http://www.ptc.com/application-lifecycle-management) |

MKS Integrity ahora es un producto de [PTC](https://en.wikipedia.org/wiki/Parametric_Technology_Corporation) desde la adquisición de MKS Inc. que se completó el 31 de mayo de 2011 por PTC. [[1]](https://en.wikipedia.org/wiki/PTC_Integrity#cite_note-1)

PTC Integrity Lifecycle Manager (Integrity LM o ILM) permite a los equipos de desarrollo de software realizar un seguimiento de todos los aspectos de su trabajo, incluidos los elementos de trabajo, el control de origen, los informes y la administración de compilación, en un solo producto. El producto consta de dos componentes: Integrity Configuration Management e Integrity Workflow & Documents. La parte de Configuration Management de PTC ILM se usa para manejar versiones de código fuente, ramas, etc. Se basa en la arquitectura cliente-servidor. El cliente de Thick Java no almacena ningún dato de gestión en el sistema local, por lo tanto, cualquier tarea realizada en los archivos fuente requiere una conexión de red. Esto significa que, a diferencia de [los sistemas distribuidos](https://en.wikipedia.org/wiki/Distributed_revision_control), este sistema requiere una conexión de red confiable, suficiente ancho de banda de red y suficiente poder de procesamiento en el lado del servidor. El otro componente (Workflow y documentos) consiste en un [sistema de seguimiento de problemas](https://en.wikipedia.org/wiki/Issue_tracking_system) , así como una solución de gestión de requisitos y requisitos.

Uno de los puntos fuertes en comparación con otras soluciones similares es la flexibilidad de PTC Integrity en términos de flujo de trabajo, campos, diseño de presentación, validación y capacidades de automatización. PTC Integrity Lifecycle Manager se basa en Java y utiliza una extensión de JavaScript para la creación de informes. Cualquier interacción se puede realizar en línea, en la CLI (Interfaz de línea de comando) o utilizando la API de Java del servidor o del cliente.

PTC Integrity Lifecycle Manager se basa en un único repositorio. Esta solución de repositorio único admite los tres pilares de la gestión del ciclo de vida: trazabilidad, automatización de procesos, informes y análisis [[2]](https://en.wikipedia.org/wiki/PTC_Integrity#cite_note-2) y algunas empresas pueden ver un valor adicional en este enfoque. [[3]](https://en.wikipedia.org/wiki/PTC_Integrity#cite_note-3)

La integración de PTC Integrity Lifecycle Manager con [IDEs](https://en.wikipedia.org/wiki/Integrated_development_environment" \o "Entorno de desarrollo integrado) y otras herramientas de desarrollo está, de fábrica, limitada a pocos productos. Los IDE admitidos incluyen [Eclipse](https://en.wikipedia.org/wiki/Eclipse_(software))[[4]](https://en.wikipedia.org/wiki/PTC_Integrity#cite_note-4) y [Visual Studio](https://en.wikipedia.org/wiki/Visual_Studio) . [[5]](https://en.wikipedia.org/wiki/PTC_Integrity#cite_note-5) También son compatibles [IBM i](https://en.wikipedia.org/wiki/IBM_i) y [Apache Maven](https://en.wikipedia.org/wiki/Apache_Maven) .

2001 **El**[software](https://en.wikipedia.org/wiki/Software)**GNU arch** es un sistema de [control de revisión distribuido](https://en.wikipedia.org/wiki/Distributed_revision_control) que es parte del [Proyecto GNU](https://en.wikipedia.org/wiki/GNU_Project) y está licenciado bajo la [Licencia Pública General de GNU](https://en.wikipedia.org/wiki/GNU_General_Public_License) . Se utiliza para realizar un seguimiento de los cambios realizados en un árbol fuente y para ayudar a los programadores a combinar y manipular los cambios realizados por varias personas o en diferentes momentos.

A partir de 2009, el estado oficial del arco de GNU es la desactivación, y solo se aplican correcciones de seguridad. [[1]](https://en.wikipedia.org/wiki/GNU_arch#cite_note-arch-status-2008-1) [Bazaar (o 'bzr')](https://en.wikipedia.org/wiki/GNU_bzr" \o "GNU bzr)también se ha convertido en un proyecto oficial de GNU y, por lo tanto, puede considerarse el reemplazo del arco de GNU. No es un tenedor de Arch.

|  |  |
| --- | --- |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | Thomas Lord |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | Andy Tai |
| [**Último lanzamiento**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 1.3.5 / 20 de julio de 2006 ; Hace 11 años |
| **Estado del desarrollo** | Solo correcciones de seguridad [[1]](https://en.wikipedia.org/wiki/GNU_arch#cite_note-arch-status-2008-1) |
| **Escrito en** | [do](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [GNU](https://en.wikipedia.org/wiki/GNU" \o "ÑU) / [Linux](https://en.wikipedia.org/wiki/Linux" \o "Linux) ,[Windows](https://en.wikipedia.org/wiki/Microsoft_Windows" \o "Microsoft Windows) , [Mac OS X](https://en.wikipedia.org/wiki/Mac_OS_X" \o "Mac OS X) |
| **[Tipo](https://en.wikipedia.org/wiki/Software_categories" \l "Broad_categories" \o "Categorías de software)** | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [GPL](https://en.wikipedia.org/wiki/GNU_General_Public_License) |
| **Sitio web** | [www .gnu .org/ software / gnu-arch /](https://www.gnu.org/software/gnu-arch/) |

Al tratarse de un sistema de control de versiones distribuido y descentralizado, cada revisión almacenada utilizando arch es única e identificable globalmente; dicho identificador se puede usar en una configuración [distribuida](https://en.wikipedia.org/wiki/Distributed_system) para fusionar fácilmente o "seleccionar cuidadosamente" cambios de fuentes completamente dispares.

Estar descentralizado significa que no hay necesidad de un servidor central para el cual los desarrolladores tengan que ser autorizados para contribuir. Al igual que con otros sistemas, se puede acceder a una copia completa de solo lectura de un proyecto en un repositorio "oficial" a través de [HTTP](https://en.wikipedia.org/wiki/HTTP) , [FTP](https://en.wikipedia.org/wiki/File_Transfer_Protocol) o [SFTP](https://en.wikipedia.org/wiki/SSH_file_transfer_protocol) ; pero luego, se alienta a los contribuyentes a realizar modificaciones y publicarlas en un archivo público (repositorio) propio, de modo que el desarrollador principal pueda fusionar manualmente los conjuntos de cambios en el repositorio oficial.

Para simular el comportamiento de los sistemas de control de revisiones centralizados, el desarrollador principal podría permitir el acceso al shell ( [SSH](https://en.wikipedia.org/wiki/Secure_Shell) ) o el acceso de escritura (FTP, SFTP, [WebDAV](https://en.wikipedia.org/wiki/WebDAV" \o "WebDAV) ) a un servidor, permitiendo a los usuarios autorizados comprometerse con un servidor central. Más a menudo, los proyectos archivados de GNU tienen un [dictador benévolo](https://en.wikipedia.org/wiki/Benevolent_dictator) líder que fusiona los cambios con los contribuyentes.

GNU arch tiene otras muchas características:

[**Compromisos atómicos**](https://en.wikipedia.org/wiki/Atomic_commit)

Los compromisos son todo o nada. El árbol debe estar en buenas condiciones antes de que comience el compromiso, y las confirmaciones no son visibles para el mundo hasta que se completen. Si la confirmación se interrumpe antes de esto, permanece invisible y debe revertirse antes de la siguiente confirmación. Esto evita la corrupción del archivo y las copias extraídas de otros usuarios.

**Orientado a los cambios**

En lugar de rastrear archivos individuales (como en [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) ), GNU arch rastrea los conjuntos de cambios, que son similares a los parches. Cada conjunto de cambios es una descripción de la diferencia entre un árbol fuente y otro, por lo que un conjunto de cambios puede usarse para producir una revisión de otra revisión. Se alienta a los autores a usar una confirmación por función o corrección de errores.

**Fácil ramificación**

La ramificación es eficiente y puede abarcar archivos. Una rama (o 'etiqueta') simplemente declara la revisión del ancestro, y el desarrollo continúa desde allí.

**Fusión avanzada**

Debido al registro permanente de todos los antepasados ​​y las revisiones fusionadas, la fusión puede tener en cuenta qué rama contiene qué parche y puede realizar una combinación de tres vías basada en una revisión de antepasados ​​compartida.

[**Firmas criptográficas**](https://en.wikipedia.org/wiki/Cryptographic_signature)

Cada conjunto de cambios se almacena con un [hash](https://en.wikipedia.org/wiki/Cryptographic_hash_function) para evitar daños accidentales. Al usar un programa externo de firma de archivos (como [GnuPG](https://en.wikipedia.org/wiki/GnuPG" \o "GnuPG) u otro cliente [PGP](https://en.wikipedia.org/wiki/Pretty_Good_Privacy) ), estos valores hash también se pueden firmar opcionalmente, evitando modificaciones no autorizadas si el archivo está en peligro.

**Renombrar**

Todos los archivos y directorios pueden renombrarse fácilmente. Estos se rastrean mediante un ID único en lugar de por su nombre, por lo que se conserva el historial y los parches de los archivos se fusionan correctamente, incluso si los nombres de archivo son diferentes en las sucursales.

**Seguimiento de metadatos**

Los [permisos](https://en.wikipedia.org/wiki/File_system_permissions) de todos los archivos son rastreados. [Los enlaces simbólicos](https://en.wikipedia.org/wiki/Symbolic_link) son compatibles y son rastreados de la misma manera que los archivos y directorios.

2002 **Darcs** es un sistema de [control de versiones distribuidas](https://en.wikipedia.org/wiki/Distributed_version_control) creado por [David Roundy](https://en.wikipedia.org/wiki/David_Roundy) . Las características clave incluyen la capacidad de elegir qué cambios aceptar de otros repositorios, la interacción con otros repositorios locales (en disco) o repositorios remotos a través de [SSH](https://en.wikipedia.org/wiki/Secure_Shell) , [HTTP](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol) o correo electrónico, y una interfaz inusualmente interactiva. Los desarrolladores también enfatizan el uso de herramientas de software avanzadas para verificar la corrección: el [sistema](https://en.wikipedia.org/wiki/Type_system) de [tipo](https://en.wikipedia.org/wiki/Type_system) expresivo del [lenguaje de programación funcional](https://en.wikipedia.org/wiki/Functional_programming)[Haskell](https://en.wikipedia.org/wiki/Haskell_(programming_language)) impone algunas propiedades, y las pruebas aleatorias a través de [QuickCheck](https://en.wikipedia.org/wiki/QuickCheck" \o "Comprobación rápida) verifican muchas otras. [[2]](https://en.wikipedia.org/wiki/Darcs#cite_note-FOOTNOTERoundy20052-2) El nombre es un [acrónimo recursivo](https://en.wikipedia.org/wiki/Recursive_acronym)para el **Sistema de Control de Revisión Avanzado de Darcs** .

|  |  |
| --- | --- |
| [**Autor (es) original (es)**](https://en.wikipedia.org/wiki/Software_developer) | [David Roundy](https://en.wikipedia.org/wiki/David_Roundy) |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | Guillaume Hoffmann, y col. |
| **Versión inicial** | 3 de marzo de 2003 ; Hace 14 años [[1]](https://en.wikipedia.org/wiki/Darcs#cite_note-darcs-changelog-1) |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 2.12.5 / 11 de enero de 2017 ; Hace 12 meses [[1]](https://en.wikipedia.org/wiki/Darcs#cite_note-darcs-changelog-1) |
| [**Repositorio**](https://en.wikipedia.org/wiki/Repository_(version_control)) | * <http://hub.darcs.net/darcs/darcs-reviewed>   [Edita esto en Wikidata](https://www.wikidata.org/wiki/Q204377#P1324) |
| **Escrito en** | [Haskell](https://en.wikipedia.org/wiki/Haskell_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Unix](https://en.wikipedia.org/wiki/Unix" \o "Unix) , [Linux](https://en.wikipedia.org/wiki/Linux" \o "Linux) , [BSD](https://en.wikipedia.org/wiki/BSD" \o "BSD) , [Apple](https://en.wikipedia.org/wiki/Apple,_Inc." \o "Apple Inc.)[macOS](https://en.wikipedia.org/wiki/MacOS" \o "Mac OS) , [MS](https://en.wikipedia.org/wiki/Microsoft" \o "Microsoft)[Windows](https://en.wikipedia.org/wiki/Microsoft_Windows" \o "Microsoft Windows) |
| **[Tipo](https://en.wikipedia.org/wiki/Software_categories" \l "Broad_categories" \o "Categorías de software)** | [Control de versiones](https://en.wikipedia.org/wiki/Version_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [GNU](https://en.wikipedia.org/wiki/GNU)[GPL](https://en.wikipedia.org/wiki/GPL_license) |
| **Sitio web** | [darcs .net](http://darcs.net/) |

Darcs trata parches como [ciudadanos de primera clase](https://en.wikipedia.org/wiki/First-class_citizen) . Para el usuario, un repositorio puede verse como un conjunto de parches, donde cada parche no está necesariamente ordenado con respecto a otros parches, es decir, el conjunto de parches es solo un [conjunto parcialmente ordenado](https://en.wikipedia.org/wiki/Partially_ordered_set) . En muchos casos, los parches se pueden transmitir de forma independiente entre varios repositorios.

Muchas operaciones de bifurcación, [fusión](https://en.wikipedia.org/wiki/Merge_(version_control)) y selección que requerirían comandos adicionales con sistemas basados ​​en instantáneas como [Git](https://en.wikipedia.org/wiki/Git_(software)" \o "Git (software)) o [Mercurial se](https://en.wikipedia.org/wiki/Mercurial) pueden hacer directamente con Darcs con los comandos habituales "extraer" y "presionar". En términos de interfaz de usuario, esto significa que Darcs tiene menos comandos. Estos comandos son más interactivos: uno puede elegir con más precisión qué parches quieren intercambiar con repositorios remotos.

Los parches de un repositorio están ordenados linealmente. Darcs calcula automáticamente si los parches se pueden reordenar (una operación llamada conmutación) y cómo hacerlo. Estos cálculos implementan una llamada "teoría de parches".

Un parche de Darcs puede contener cambios de los siguientes tipos:

* cambios de línea,
* creación y eliminación de archivos y directorios,
* archivo y directorio en movimiento,
* sustitución de palabras (normalmente utilizada en la refactorización de códigos, por ejemplo, cambiar el nombre de todas las apariciones de "foo" a "bar" en un archivo determinado).

La noción de dependencia entre parches se define sintácticamente. Intuitivamente, un parche *B* depende de otro parche *A* si *A* proporciona el contenido que *B* modifica. Esto significa que los parches que modifican diferentes partes del código se consideran, por defecto, independientes. Para abordar casos cuando esto no es deseable, Darcs permite al usuario especificar dependencias explícitas entre parches.

Desde la versión 2.10, Darcs utiliza la [paciencia diff](https://en.wikipedia.org/w/index.php?title=Patience_diff&action=edit&redlink=1) por defecto.

2002 **Distributed Concurrent Versions System** (**DCVS**)  ( **DCVS** ) es un sistema de [control de revisiones distribuidas](https://en.wikipedia.org/wiki/Distributed_revision_control) que permite a los desarrolladores de software que trabajan en sitios distribuidos localmente colaborar de manera eficiente en un proyecto de software. DCVS se basa en el conocido sistema de control de [*versiones Sistema de versiones concurrentes*](https://en.wikipedia.org/wiki/Concurrent_Versions_System) . El código es libremente distribuible bajo las licencias de estilo GNU y BSD.

|  |  |
| --- | --- |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | elego Software Solutions GmbH |
| **Versión inicial** | Agosto de 2002 ; hace 15 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 1.0.3 / 25 de septiembre de 2006 ; Hace 11 años |
| **Estado del desarrollo** | Sin mantenimiento |
| **Escrito en** | [do](https://en.wikipedia.org/wiki/C_(programming_language)) |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Unix](https://en.wikipedia.org/wiki/Unix-like) , [Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Licencia pública general de GNU](https://en.wikipedia.org/wiki/GNU_General_Public_License) |
| **Sitio web** | [dcvs .elegosoft .com](http://dcvs.elegosoft.com/) |

DCVS proporciona toda la funcionalidad de CVS. Pero a diferencia de CVS, un sistema DCVS puede comprender un número arbitrario de repositorios distribuidos geográficamente cuyos contenidos se mantienen iguales en segundo plano por una versión extendida de *[CVSup](https://en.wikipedia.org/w/index.php?title=CVSup&action=edit&redlink=1" \o "CVSup (la página no existe))* , un programa desarrollado por John D. Polstra. La combinación del *repositorio DCVS* , el *servidor CVSup* extendido y el *programa del servidor DCVS* se denominará *servidor DCVS* en los párrafos siguientes.

Todos los contenidos de todas las líneas de desarrollo se pueden *extraer* de cualquiera de los servidores DCVS en un *espacio de trabajo DCVS* propiedad de un desarrollador. Todas las operaciones que no modifican el repositorio, como *[diff](https://en.wikipedia.org/wiki/Diff" \o "Diferencia)* , *[patch](https://en.wikipedia.org/wiki/Patch_(Unix)" \o "Parche (Unix))* , *log* , *annotate*etc., funcionan igual que en CVS, pero siempre usan el repositorio local y, por lo tanto, son mucho más rápidos en un escenario distribuido. Para evitar colisiones y pérdida de datos, a cada servidor DCVS se le asigna un conjunto de líneas de desarrollo (ramas DCVS) de las que es responsable. Las modificaciones en una sucursal solo pueden registrarse en el servidor responsable de la sucursal. La separación de modificaciones por líneas de desarrollo hace posible transferir y distribuir automáticamente los cambios en la red DCVS. Una persona que desee realizar cambios para una línea de desarrollo de la que su servidor DCVS local no es responsable puede crear una nueva línea de desarrollo (sucursal) y comprometer los cambios a la misma. El servidor local es automáticamente responsable de cualquier línea de desarrollo recién creada.

Por otro lado, cada desarrollador puede fusionar los cambios de las líneas de desarrollo que su servidor DCVS local no es responsable de las líneas de desarrollo locales. Por lo tanto, todos los cambios realizados en cualquier sitio de trabajo pueden ser aplicados a la rama original por los desarrolladores en el servidor DCVS responsable que realiza una operación de [*fusión*](https://en.wikipedia.org/wiki/Merge_(revision_control)) .

La funcionalidad de los [*conjuntos*](https://en.wikipedia.org/wiki/Changeset) de [*cambios*](https://en.wikipedia.org/wiki/Changeset) permite a los desarrolladores producir pequeños conjuntos de cambios relacionados con una característica o un defecto, que luego pueden ser aplicados por otros.

Un esquema de numeración específico garantiza que las líneas de desarrollo y los deltas se puedan identificar como pertenecientes a un determinado servidor DCVS. DCVS asigna un [*rango*](https://en.wikipedia.org/wiki/Range_(statistics)) único de números de ramificación a cada par (servidor / colección). Todos los rangos para todos los servidores y colecciones deben ser mutuamente exclusivos. Las definiciones de servidores, colecciones y rangos se leen desde un único archivo de configuración. Al consultar el contenido de este archivo, cada servidor DCVS puede decidir si es responsable de una determinada rama o delta de un archivo determinado. Si es así, se permiten todas las operaciones de modificación; si no, las operaciones de modificación solo son posibles en el servidor remoto apropiado.

Además, los nombres reales de las configuraciones, las [*etiquetas*](https://en.wikipedia.org/wiki/Tag_(metadata)) , se asignan de manera única a exactamente un servidor DCVS por medio de un sufijo específico del servidor que extiende todas las *etiquetas* (es decir, \_at\_dcvs\_mydomain\_org). Por lo tanto, no pueden surgir conflictos en el espacio del nombre de la etiqueta.

2002 **Surround SCM** es una aplicación de [administración de configuración de software](https://en.wikipedia.org/wiki/Software_configuration_management) desarrollada por [Seapine Software](https://en.wikipedia.org/wiki/Seapine_Software" \o "Seapine Software) . En 2008, Surround SCM ganó un [Premio Jolt](https://en.wikipedia.org/wiki/Jolt_Awards) en la categoría Gestión de cambios y configuración.

|  |  |
| --- | --- |
| [**Desarrollador (es)**](https://en.wikipedia.org/wiki/Software_developer) | [Seapine Software](https://en.wikipedia.org/wiki/Seapine_Software) |
| **Versión inicial** | Septiembre de 2002 ; hace 15 años |
| [**Lanzamiento estable**](https://en.wikipedia.org/wiki/Software_release_life_cycle) | 2017.1.0 / 24 de abril de 2017 ; hace 8 meses |
| [**Sistema operativo**](https://en.wikipedia.org/wiki/Operating_system) | [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) ,[Linux](https://en.wikipedia.org/wiki/Linux) , [Mac OS X](https://en.wikipedia.org/wiki/Mac_OS_X) |
| [**Tipo**](https://en.wikipedia.org/wiki/Software_categories#Broad_categories) | [Control de revisión](https://en.wikipedia.org/wiki/Revision_control) |
| [**Licencia**](https://en.wikipedia.org/wiki/Software_license) | [Propiedad](https://en.wikipedia.org/wiki/Proprietary_software) |
| **Sitio web** | [www .seapine .com](http://www.seapine.com/) |