

**INSTITUTO TECNOLÓGICO**

**de Tuxtepec**

**S.E.P. TECNOLÓGICO NACIONAL DE MÉXICO**

ACTIVIDAD:

**Investigación sobre CVS**

PRESENTAN:

**CASTILLO RODRIGUEZ URIEL (21350239)**

**CERVANTES GUATEMALA SERGIO (21350486)**

**GARCIA GARCIA JESUS ALBERTO (21350249)**

**VITAL BENITEZ IRVING ENRIQUE (21350311)**

MATERIA:

**Sistemas de control de Versiones**

DOCENTE:

**Olivia Guadalupe López Ruíz**

CARRERA:

**INGENIERIA EN SISTEMAS COMPUTACIONALES**

TUXTEPEC, OAX, A 18 de Marzo del 2025

INTRODUCCIÓN

¿QUÉ ES CVS?

CREADORES

EMPRESA QUE LO DESARROLLÓ

HISTORIA

**1986: Los inicios de CVS**

* Dick Grune, un profesor e investigador en ciencias de la computación, desarrolló CVS como un conjunto de scripts en shell (lenguaje de comandos de Unix).
* El objetivo principal era gestionar el código fuente de sus proyectos personales y facilitar la colaboración con otros desarrolladores.
* En esta etapa, CVS era una herramienta básica, pero sentó las bases para lo que sería un sistema de control de versiones más robusto.

**1989: La reescritura en C**

* Brian Berliner, un ingeniero de software, reescribió CVS en el lenguaje de programación C.
* Esta reescritura permitió que CVS fuera más eficiente, escalable y fácil de usar.
* Berliner también agregó funcionalidades clave, como el soporte para operaciones concurrentes (de ahí el nombre "Concurrent Versions System").
* CVS comenzó a ganar popularidad en la comunidad de software libre.

**1990: Lanzamiento público**

* CVS se lanzó públicamente como una herramienta de código abierto.
* Su adopción creció rápidamente, especialmente entre desarrolladores que trabajaban en proyectos colaborativos.
* En esta época, CVS se convirtió en una alternativa popular a sistemas de control de versiones más antiguos, como RCS (Revision Control System).

**1991-1995: Expansión y adopción masiva**

* CVS se convirtió en el sistema de control de versiones predilecto para proyectos de software libre y de código abierto.
* Proyectos emblemáticos, como el desarrollo del kernel de Linux, utilizaron CVS en sus primeras etapas.
* La comunidad de software libre contribuyó a mejorar CVS, agregando nuevas funcionalidades y corrigiendo errores.

**1996: CVS 1.0**

* Se lanzó la versión 1.0 de CVS, que incluía mejoras significativas en cuanto a estabilidad y rendimiento.
* Esta versión consolidó a CVS como una herramienta confiable para la gestión de versiones en proyectos de mediana y gran escala.

**1998-2000: Dominio en la industria**

* CVS alcanzó su punto máximo de popularidad durante estos años.
* Fue ampliamente adoptado por empresas, instituciones educativas y proyectos de software libre.
* Sin embargo, también comenzaron a surgir críticas sobre sus limitaciones, como el manejo ineficiente de archivos binarios y la falta de soporte para operaciones atómicas.

**2001: Competencia emergente**

* Subversion (SVN), un sistema de control de versiones diseñado para superar las limitaciones de CVS, fue lanzado como proyecto de código abierto.
* Subversion ofrecía ventajas significativas, como operaciones atómicas y un mejor manejo de archivos binarios.
* Aunque CVS seguía siendo popular, muchos proyectos comenzaron a migrar a Subversion.

**2005: El auge de Git**

* Git, un sistema de control de versiones distribuido creado por Linus Torvalds, fue lanzado.
* Git ofrecía una arquitectura distribuida, lo que permitía a los desarrolladores trabajar sin depender de un servidor central.
* Aunque Git no reemplazó a CVS de inmediato, marcó el comienzo de un cambio en la industria hacia sistemas más modernos.

**2010: Declive de CVS**

* Para esta década, CVS había perdido gran parte de su relevancia.
* La mayoría de los proyectos nuevos optaban por Git o Subversion.
* Sin embargo, algunos proyectos legacy continuaron utilizando CVS debido a la complejidad de migrar a un nuevo sistema.

**Actualidad (2020s): Legado de CVS**

* CVS ya no es ampliamente utilizado, pero su legado perdura.
* Fue uno de los primeros sistemas de control de versiones en facilitar la colaboración entre desarrolladores.
* Muchas de las funcionalidades y conceptos introducidos por CVS se han incorporado en sistemas modernos como Git y Subversion.

CARACTERÍSTICAS DE CVS

VENTAJAS

EMPRESAS QUE LO UTILIZAN

CONCLUSIÓN

REFERENCIAS BIBLIOGRÁFICAS

* Prezi, A. E. J. O. (s. f.). CVS(Sistema Control Versiones). prezi.com. <https://prezi.com/tuqifpcirncw/cvssistema-control-versiones/>
* Rgomez. (2015, 27 mayo). Historia (informal) de los sistemas de control de versiones. Blog Irontec. <https://blog.irontec.com/historia-informal-de-los-sistemas-de-control-de-versiones/>