**2. Una visión hegeliana del pasado de la ingeniería de software**

El filósofo Hegel planteó la hipótesis de que el aumento de la comprensión sigue un camino de tesis (es por eso que las cosas suceden de la manera que lo hacen); Antítesis (la tesis falla en algunos aspectos importantes; Aquí hay una mejor explicación); Y síntesis (la antítesis rechazada demasiado de la tesis original; Aquí hay un híbrido que captura el mejor de ambos evitando sus defectos). A continuación intentaré aplicar esta hipótesis para explicar la evolución de la ingeniería de software de los años cincuenta hasta el presente.

**2.1 Tesis de los 50’: Ingeniería de Software como Ingeniería de Hardware**

Cuando entré en el campo del software en 1955 en General Dynamics, la tesis predominante fue: "Software de ingeniería como ingeniería de Hardware ". Todos en la organización de software de GD eran

Un ingeniero de hardware o un matemático, y el software

Desarrollado era el apoyo a la ingeniería de aviones o cohetes. Gente

Mantienen cuadernos de ingeniería y practicaron tales preceptos de hardware como

"Medir dos veces, cortar una vez", antes de ejecutar su código en el

computadora.

Este comportamiento también fue consistente con la computación de 1950

ciencias económicas. En mi primer día en el trabajo, mi supervisor me mostró

La computadora GD ERA 1103, que llenaba una gran sala. Él dijo,

"Ahora escucha. Estamos pagando $ 600 por hora por esta computadora y $ 2

Una hora para ti, y quiero que actúes en consecuencia. "Esto inculcó

Una serie de buenas prácticas como la comprobación de

Comprobar y ejecutar manualmente mis programas antes de ejecutar

ellos. Pero también me dejó con un sesgo hacia el ahorro de microsegundos

Cuando el equilibrio económico comenzó a ir en el otro sentido.

El proyecto de procesamiento de información más ambicioso de la década de 1950

Fue el desarrollo del Semi-Automated Ground Environment

(SAGE) para la defensa aérea estadounidense y canadiense. Reunió

Principales ingenieros de radar, ingenieros de comunicaciones,

Ingenieros y ingenieros de software nacientes para desarrollar un sistema que

Detectar, rastrear y evitar que los aviones enemigos bombardearan la

EE.UU. y Canadá homelands.

La Figura 1 muestra el proceso de desarrollo de software desarrollado por

Ingenieros de hardware para su uso en SAGE [1]. Muestra que las secuencias

Modelos de cascada se han utilizado en el desarrollo de software para

largo tiempo. Además, si uno organiza los pasos en una forma V con Codificación

En la parte inferior, este proceso de 1956 es equivalente al V-modelo para

desarrollo de software. SAGE también desarrolló el Lincoln Labs

Utility System para ayudar a los miles de programadores que participan en

Desarrollo de software SAGE. Incluía un ensamblador, una biblioteca y

Sistema de gestión de la construcción, una serie de programas de utilidad y

Pruebas y depuración. El sistema SAGE resultante se ha cumplido satisfactoriamente

Sus especificaciones con un deslizamiento de un año aproximadamente. Benington

El comentario de fondo sobre el éxito fue "Es fácil para mí solteros

Un factor que nos llevó a nuestro relativo éxito: todos éramos

Ingenieros y había sido entrenado para organizar nuestros esfuerzos

Líneas de ingeniería ".

Otra indicación de la orientación de la ingeniería de hardware de la

De los años 1950 está en los nombres de las principales sociedades

Profesionales del software: la Asociación para la Maquinaria Informática

Y la IEEE Computer Society.