



Verificación y Validación de Software

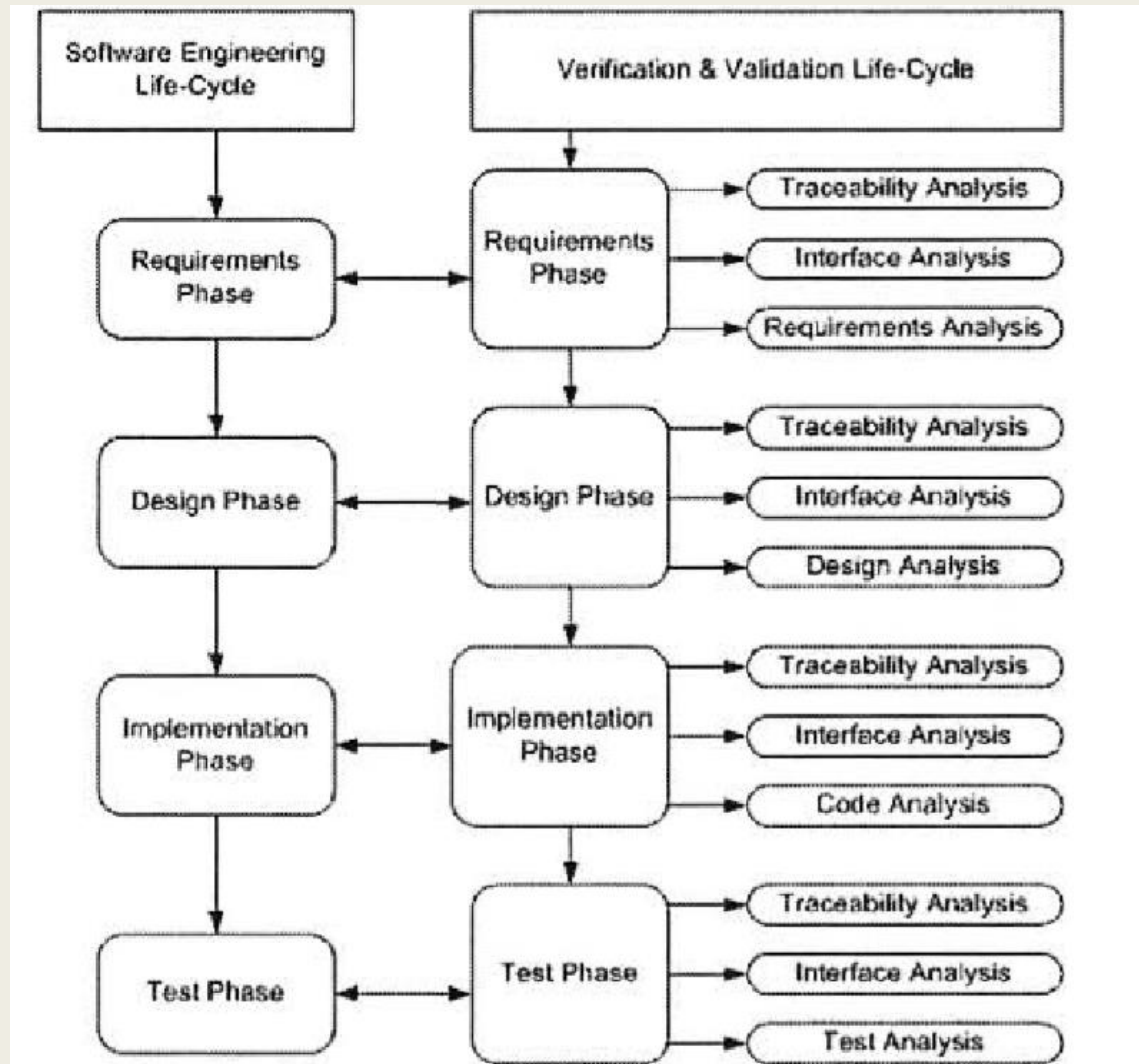
MIS. ELIZABETH MURRIETA SANGABRIEL



Verificación y validación en el proceso de desarrollo

- Modelo de ciclo de vida de verificación y validación (V&V) en relación con un ciclo de vida tradicional de ingeniería de software.
 - *Por ejemplo, la fase de requisitos de V&V está en línea con la fase de requisitos de ingeniería de software en la que el equipo de V&V cumple los requisitos de análisis de trazabilidad, análisis de interfaz y análisis de requisitos.*

Verificación y validación en el proceso de desarrollo



Verificación y validación en el proceso de desarrollo - Requerimientos y análisis -

- El equipo de V&V realiza el análisis de requisitos para garantizar que se hayan identificado los requisitos de software correctos, que satisfagan las necesidades del sistema, que sean consistentes y que puedan verificarse mediante pruebas.
- Es durante la fase de requisitos que el proyecto de desarrollo establecerá lo que se supone que debe hacer el sistema de software.
- Esta fase del ciclo de vida es la más crítica y V&V puede tener el mayor impacto al cumpliendo los requisitos de V&V para esta fase.
- Hay varios enfoques que V&V puede adoptar para cumplir con estos requisitos.
- Desde la revisión manual de las especificaciones de requisitos, la realización de análisis estáticos en los documentos, el modelado y la ejecución de las especificaciones y la demostración matemática de la existencia u omisión de propiedades específicas.

Verificación y validación en el proceso de desarrollo - Requerimientos y análisis -

- Durante esta fase del ciclo de vida, el equipo de V&V ya ha establecido las características del sistema de interés para el esfuerzo de V&V y ha confirmado que se han asignado al software a través del análisis de trazabilidad.
- El equipo de V&V también ha asegurado que los requisitos de la interfaz son adecuados para satisfacer las necesidades del sistema.
- Ahora es el momento de determinar el mérito técnico de los requisitos de software.

Verificación y validación en el proceso de desarrollo - Requerimientos y análisis -

- V&V asegurará que se hayan identificado los requisitos de software correctos.
- V&V asegurará que los requisitos de software satisfagan los requisitos del sistema.
- V&V asegurará que los requisitos de software estén completamente definidos.
- V&V asegurará que cada requisito de software se use de manera consistente.
- V&V asegurará que los requisitos de software cumplan con las normas y prácticas de ingeniería apropiadas.
- V&V asegurará que la precisión lógica y computacional satisfaga las necesidades del sistema.
- V&V asegurará que todos los requisitos de software son comprobables.

Verificación y validación en el proceso de desarrollo - Requerimientos y análisis -

- Los cuatro enfoques son **análisis manual**, **análisis estático**, **análisis dinámico** y **análisis formal**.
- **Análisis manual:** Durante la fase de planificación, el equipo de V&V debe ser diligente en la planificación de los enfoques que se utilizarán para cada análisis que se realice. Esto no es algo que se decida más adelante en el proyecto.
- El análisis manual es simplemente un experto en el dominio que se presenta para revisar los requisitos de software.
- Se basa únicamente en su experiencia y conocimiento. Se estudia los artefactos bajo escrutinio y hace observaciones basadas en esa experiencia y conocimiento.
- **Desventajas:**
- Este es un enfoque costoso porque los expertos en dominios no son baratos.
- Adoptar este enfoque es que no tiene una comprensión clara de si el experto en el dominio, a través del estudio de los requisitos, cumplió con los requisitos de V&V para esta fase.
- Será difícil repetir el análisis y lograr los mismos resultados con analistas diferentes.

Verificación y validación en el proceso de desarrollo - Requerimientos y análisis -

- **Análisis estático:** Ha habido avances prometedores en el área del análisis estático y el uso de herramientas para ayudar al experto en el dominio en esta evaluación.
- Una de esas áreas estaba en el método SCR basado en tablas para especificar los requisitos (Heitmeyer 1998).
- Este tipo de método proporciona la capacidad de garantizar la integridad y la coherencia con los requisitos hasta cierto punto.
- El método basado en tablas para especificar requisitos puede no ser adecuado para todos los tipos de requisitos (por ejemplo, depurador de memoria), sin embargo, parece directamente aplicable a los requisitos de tipo de control que enfrentamos en nuestras aplicaciones críticas (por ejemplo, temporizador de vigilancia).
- Una cosa a tener en cuenta aquí es que al cumplir con los requisitos de integridad, el enfoque de SCR se centra principalmente en los estados y las transiciones de estado representados en los requisitos.
- El mismo enfoque puede integrarse en la verificación del modelo discutido.

Verificación y validación en el proceso de desarrollo - Requerimientos y análisis -

- **Análisis estático:** La razón por la que sugiero un enfoque de análisis estático es que el soporte de herramientas aliviaría en gran medida la carga de un analista al verificar los requisitos en cuanto a si siguen un estándar particular o las mejores prácticas de ingeniería.
- Las herramientas que escanean un documento de requisitos y señalan las inconsistencias entre el estándar identificado y el documento bajo evaluación no deberían ser difíciles de construir.
- Las herramientas deben ser configurables según el estándar al que está utilizando el proyecto. Por lo tanto, se necesitaría un archivo de configuración para identificar el estándar con el que se compara el documento de requisitos.
- Existen otras herramientas que también realizan escaneos automáticos que identifican áreas de los requisitos que pueden o no cumplir con las "mejores prácticas de ingeniería".

Verificación y validación en el proceso de desarrollo - Requerimientos y análisis -

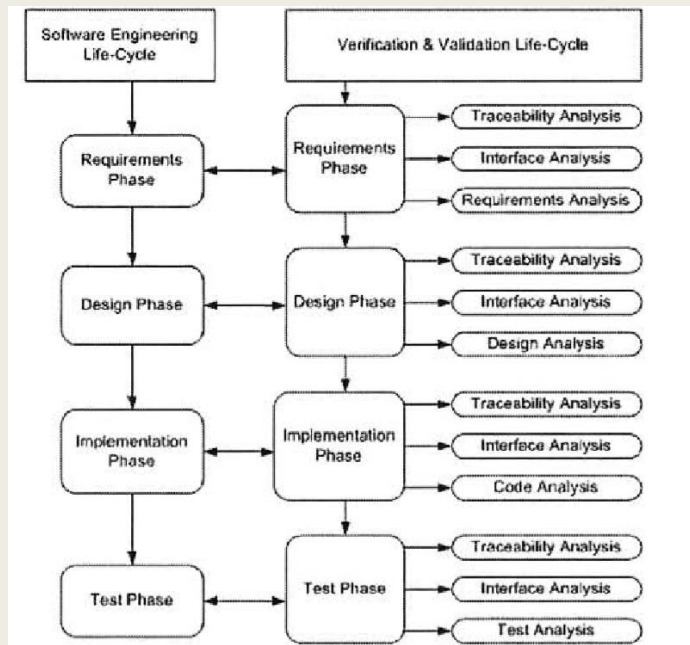
- **Análisis estático:** Se elige un enfoque de análisis estático para cumplir este requisito simplemente porque existen herramientas que analizan automáticamente una especificación de requisitos y generan casos de prueba.
- El analista podría tomar este conjunto e identificar para qué requisitos no se podrían generar casos de prueba.
- El soporte de esta herramienta se concentra en los requisitos que ya están definidos en alguna herramienta (por ejemplo, MATLAB).
- Cuando tiene requisitos que ya están definidos en un idioma en particular, debe buscar herramientas que admitan ese idioma.
- Para aquellos requisitos que se especifican en un documento de texto utilizando el lenguaje natural, la tarea se vuelve un poco más manual.

Verificación y validación en el proceso de desarrollo - Requerimientos y análisis -

- **Análisis dinámico y formal** : es precisamente donde V&V debería estar en la realización de la validación de requisitos.
- Es difícil distinguir entre los dos conceptos.
- Por ejemplo, si un modelo de requisitos utilizando el conjunto de herramientas SCR (Software Cost Reduction), adoptando un enfoque formal para modelar los requisitos.
- Sin embargo, uno de los beneficios de usar esta metodología y conjunto de herramientas es que tiene una capacidad de simulación que le permitirá ejecutar el modelo, un enfoque de análisis dinámico.
- La capacidad de simulación tiene la capacidad de identificar las diferencias entre lo que los analistas de V&V perciben que son los requisitos y el comportamiento capturado en los requisitos del software.

Verificación y validación en el proceso de desarrollo - Arquitectura y diseño-

- La fase de diseño es el período en el ciclo de vida de desarrollo donde se establecen soluciones a los requisitos de software.
- Los diseños, así como las construcciones de datos, se establecen y vinculan a los requisitos de software específicos en los que constituyen la solución.



Verificación y validación en el proceso de desarrollo - Arquitectura y diseño-

- El diseño comienza con el modelo de requisitos al que se transfiere en 4 niveles de detalles de diseño:
 - *La estructura de datos*
 - *La arquitectura del sistema*
 - *La representación de la interfaz*
 - *El detalle del nivel de componente*

Verificación y validación en el proceso de desarrollo - Arquitectura y diseño-

- Tradicionalmente, el diseño de dominio de datos de información, desde la fase de requisitos, en las estructuras de datos que se requerirán para implementar el sistema.
- Las entradas del equipo de desarrollo son el diccionario de datos y los diagramas de Entidad-Relación (ER).
- El diseño arquitectónico define las relaciones entre los principales elementos estructurales del sistema.
- La entrada para hacer esto es generalmente los diagramas de flujo de datos (DFD).
- El diseño de la interfaz describe cómo el software se comunica dentro de sí mismo, con otros sistemas y con los usuarios.

Verificación y validación en el proceso de desarrollo - Arquitectura y diseño-

- Las entradas para desarrollar esto son los DFD (data flow diagrams), las especificaciones de control (CSpecs) y los diagramas de transición de estado (STD).
- Por último, el diseño a nivel de componente transforma los elementos estructurales de la arquitectura en una descripción de procedimiento de los componentes de software.
- Las entradas a esto son especificaciones de proceso (PSpecs), STD y CSpecs. Es durante este tiempo que el equipo de V&V puede evaluar la posible solución para el sistema.
- En el pasado, el análisis de diseño ha sido en gran medida ad hoc.
- Pero no creo que sea principalmente culpa de V&V, también creo que la ingeniería de software no es tan avanzada en los diseños de edificios o debo decir que documentar sus diseños.
- Puede obtener ciertos documentos científicos que describen los algoritmos elegidos para un diseño en particular, pero no he visto un diseño de software completo para ningún proyecto del que haya formado parte.

Verificación y validación en el proceso de desarrollo - Arquitectura y diseño-

- Se presentan los requisitos de V&V para el análisis de diseño.
- V&V asegurará que se hayan identificado los elementos de diseño correctos.
- V&V asegurará que el elemento de diseño satisfaga los requisitos del software.
- V&V asegurará que los elementos de diseño estén completamente definidos.
- V&V asegurará que cada elemento de diseño se use de manera consistente.
- V&V asegurará que el elemento de diseño cumpla con las normas y prácticas de ingeniería apropiadas.
- V&V asegurará que la precisión lógica y computacional satisfaga las necesidades del sistema.
- V&V asegurará que todos los elementos de diseño sean comprobables.

Verificación y validación en el proceso de desarrollo – Análisis de Código-

- El análisis de código es otra capacidad básica que V&V proporciona al proyecto de desarrollo.
- El proyecto de desarrollo normalmente no tiene los recursos para explorar todos los problemas potenciales que se pueden experimentar durante la ejecución del software.
- Además, normalmente se emplean pequeños equipos de programación que solo se centran en su área de desarrollo.
- Durante este tiempo, V&V se centra en determinar qué tan bien se ajusta el código a la especificación de diseño general y a los requisitos del sistema.
- Su objetivo es determinar la calidad general del código.

Verificación y validación en el proceso de desarrollo – Análisis de Código-

- Los requisitos durante el análisis de código son:
 - *Asegurará que se hayan identificado los elementos de código correctos.*
 - *Asegurará que el elemento de código satisfaga el elemento de diseño.*
 - *Asegurará que los elementos del código estén completamente definidos.*
 - *Asegurará que cada elemento del código se use de manera consistente.*
 - *Asegurará que el elemento del código cumpla con los estándares apropiados y las prácticas de ingeniería.*
 - *Asegurará que la precisión lógica y computacional satisfaga las necesidades del sistema.*
 - *Asegurará que todos los elementos del código sean comprobables.*

Verificación y validación en el proceso de desarrollo – Análisis de Código-

- Un paso más es que algunos adoptan un enfoque de análisis estático al escanear el código en busca de problemas de programación comunes (es decir, las variables no se inicializan antes de usarse).
- El equipo de V&V toma el código fuente y lo modela.
- Modelar en este sentido puede no significar traducir cada línea de código a otro formato.
- Simplemente instrumentar el código con aserciones podría ser un modelo.
- El punto es que se debe desarrollar alguna representación para poder evaluar su comportamiento.
- Nuevamente, estamos tratando de cumplir con los requisitos de análisis de código que son los dos primeros para asegurar que se hayan identificado los módulos de software correctos y que satisfagan los módulos de diseño.

Verificación y validación en el proceso de desarrollo – Análisis de Código-

- El analista de V&V podría usar el módulo de diseño para impulsar el desarrollo de afirmaciones que verifican las condiciones de entrada y salida.
- Esto puede considerarse un enfoque de caja negra.
- Otras aserciones podrían instrumentarse dentro del código para validar las restricciones de rendimiento identificadas por el diseño.
- Si el sistema identifica restricciones de memoria, los analistas pueden tomar el código ensamblado y analizarlo para identificar todos los empujes y estallidos en la pila para determinar la cantidad de memoria que se utiliza en casos específicos de tiempo.

Verificación y validación en el proceso de desarrollo – Análisis de Código-

- Existen numerosos métodos que pueden emplearse, así como herramientas para apoyarlos.
- No difieren totalmente de ese enfoque, pero son de naturaleza más manual y requieren un poco de discusión.
- La técnica es inspecciones de software.
- Las inspecciones han existido desde la década de 1970 y todavía se están utilizando de manera efectiva.
- Para ponerlo en una perspectiva V&V, las inspecciones pueden cumplir con todos los requisitos de análisis de código, pero diría que no pueden cumplirlos con el nivel de rigor deseado.

Verificación y validación en el proceso de desarrollo – Análisis de Código-

- Estos requisitos tratan de asegurar que el software cumpla con los estándares y las mejores prácticas y asegurar que el software sea comprobable.
- Las inspecciones pueden ser bastante útiles para cumplir estos dos requisitos.
- Durante una inspección de software, pequeños grupos de analistas estudian los productos de trabajo de forma independiente y luego se reúnen para examinar el trabajo en detalle.
- Los productos de trabajo son pequeños, pero completos, y los analistas suelen pasar de una a cuatro horas revisando el producto de trabajo y la información relacionada antes de la reunión de inspección.

Verificación y validación en el proceso de desarrollo – Análisis de Código-

- Fueron desarrollados originalmente en IBM a principios de la década de 1970 (Fagan 1976).
- El proceso de inspección tradicional consta de los siguientes pasos:
 - *Planeación*
 - *Visión General*
 - *Preparación*
 - *Examinación*
 - *Rehacer*
 - *Seguimiento*