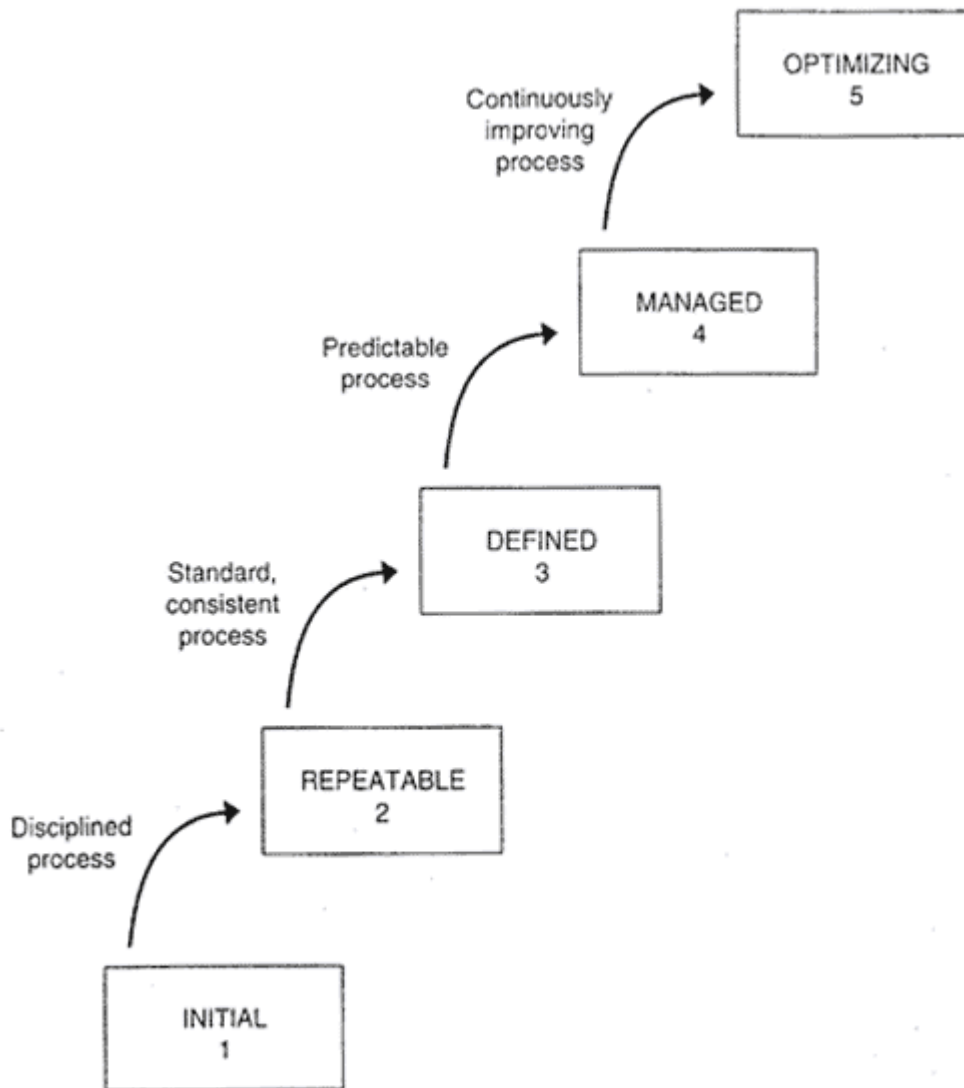


## Capabilty Maturity Model (CMM) -

**1.- Conceptos Fundamentales** La mejora continua de los procesos está basada en pasos pequeños evolutivos más que en innovaciones revolucionarias. El CMM provee un entorno para organizar estos pasos evolutivos en cinco niveles de madurez que establecen las bases para la mejora continua de los procesos. Estos cinco niveles de madurez definen una escala ordinal para medir la madurez del proceso de software de una organización y para evaluar la capacidad de proceso de software. Además ayuda a una organización a priorizar sus esfuerzos para la mejora.

Un nivel de madurez es un esfuerzo evolutivo bien definido para alcanzar un proceso de software maduro. Cada nivel de madurez está compuesto por un conjunto de objetivos de proceso que, cuando se satisfacen, establecen un componente importante del proceso de software. Alcanzar cada nivel del entorno de madurez establece un componente diferente en el proceso de software, resultando en un aumento de la capacidad de los procesos de la organización.



### Los Cinco Niveles de Madurez

Organizar el CMM en los cinco niveles mostrado en la figura a continuación, prioriza las acciones de mejora para incrementar la madurez de los procesos de software. Los arcos etiquetados indican el tipo de capacidad de proceso que es institucionalizada por la organización en cada nivel.

Los cinco niveles pueden ser descriptos brevemente como:

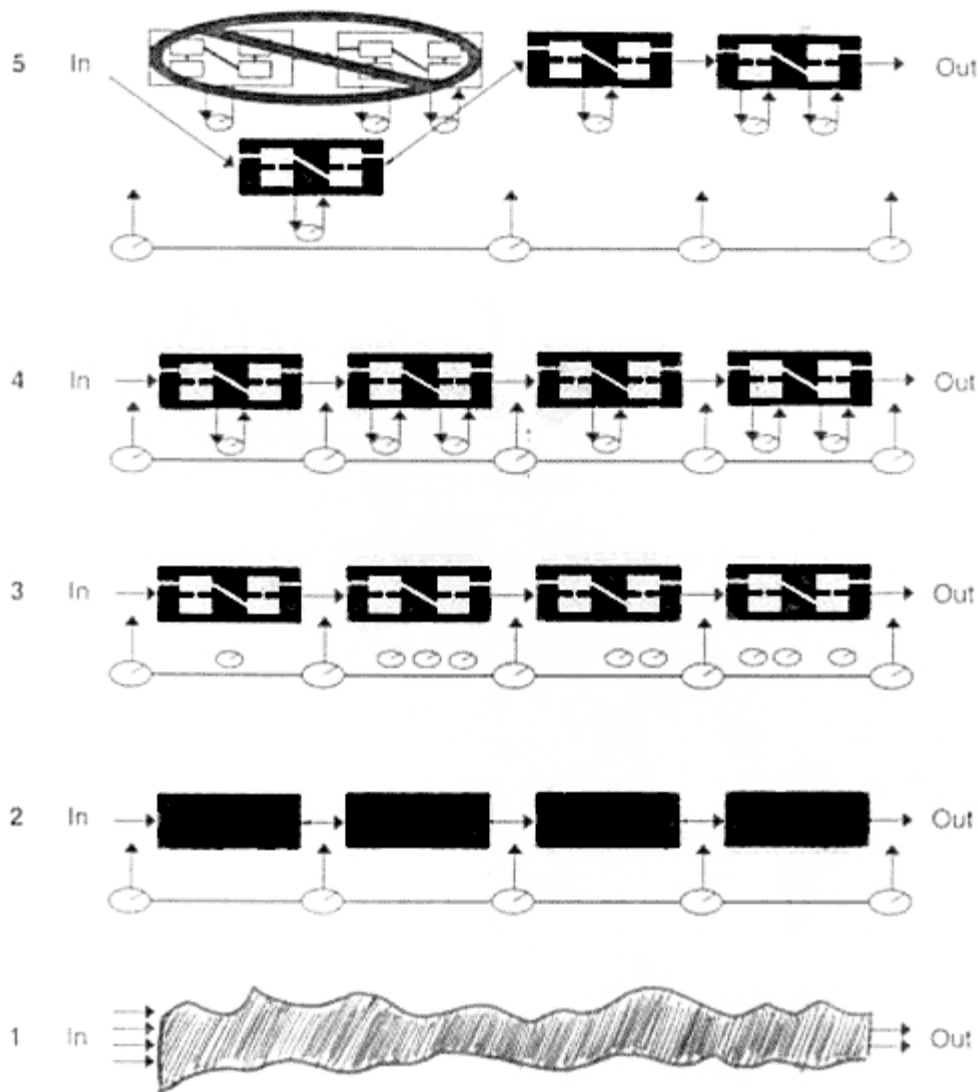
1. **Inicial:** el proceso de software está caracterizado como *ad hoc*, y a veces caótico. Pocos procesos están definidos y el éxito depende de esfuerzos individuales y heroicos.
2. **Repetible:** se establecen procesos administrativos básicos en los proyectos para el seguimiento de costos, tiempos y funcionalidades.

La disciplina necesaria en los procesos es acorde para repetir éxitos anteriores de proyectos con aplicaciones similares.

3. **Definido:** el proceso de software, tanto administrativo como ingenieril, está documentado, estandarizado e integrado en un proceso estándar para la organización. Todos los proyectos usan una versión aprobada y ajustada del proceso estándar de software de la organización para desarrollar y [mantener software](#)
4. **Administrado:** se toman decisiones del proceso de software y de la calidad del producto. Tanto el proceso de software como los productos se conocen cuantitativamente y se controlan.
5. **Optimizado:** existe retroalimentación cuantitativa del proceso, lo que permite una mejora continua del proceso. Al mismo tiempo se manejan ideas y tecnologías innovadoras.

Estas cinco etapas reflejan el echo de que CMM es un modelo para mejorar la capacidad de las empresas de software. Las prioridades en el CMM, como lo expresan estos niveles, no están dirigidas a proyectos individuales. Un proyecto en peligro podría priorizar sus problemas de manera diferente a la que plantea CMM. Sus soluciones podrían ser de poco valor para la organización, dado que otros proyectos podrían tener problemas diferentes o no poder tomar ventajas de sus soluciones dado que les falta los basamentos necesarios para implementarlas. El CMM se enfoca en procesos que son de valor a lo largo de toda la organización.

**2.- Caracterización de los Niveles de Madurez** Los niveles de madurez del 2 al 5 pueden ser caracterizados a través de las actividades que la organización lleva a cabo para establecer o mejorar el proceso de software, por las actividades que desarrollan en cada proyecto, y por la capacidad de proceso resultante a través de los proyectos.



## Visibilidad en Cada Nivel de Madurez

### 2.1- Nivel 1: Nivel Inicial

En el Nivel Inicial, la organización típicamente no representa un ambiente estable para el desarrollo y mantenimiento de software. Típicamente las organizaciones de nivel 1 se plantean compromisos sobredimensionados, y frecuentemente tienen dificultad en plantear compromisos que es staff puede alcanzar con un proceso ingenieril ordenado, resultando en una serie de crisis. Durante una crisis, los proyectos típicamente abandonan los procedimientos planeados y vuelven a la codificación y el testeo. El éxito depende de administradores excepcionales y equipos de software efectivos. Ocasionalmente, administrados de software capaces y fuertes pueden sobreponerse a las

presiones que implican tomar atajos en el proceso, pero cuando abandonan el proyecto su influencia estabilizadora se va con ellos. Incluso un proceso fuerte no puede sobrellevar la inestabilidad creada por la ausencia de prácticas administrativas sólidas.

A pesar de los procesos caóticos, las organizaciones de nivel 1 frecuentemente desarrollan productos que funcionan, aunque excedan el presupuesto y el cronograma. El éxito en las organizaciones de nivel 1 depende de la competencia y el heroísmo de las personas de la organización y no puede ser repetido a menos que los mismos individuos sean asignados al próximo proyecto. Entonces, en el nivel 1, la capacidad es una característica de los individuos, no de las organizaciones.

## **2.2- Nivel 2: Nivel Repetible**

En el Nivel Repetible, se establecen políticas para administrar un proyecto de software y procedimientos para implementar esas políticas. El planeamiento y la administración de proyectos nuevos se basan en la experiencia con proyectos similares. La capacidad de los procesos se aumenta al establecer una disciplina básica de administración de procesos en cada proyecto. Los proyectos implementan procesos efectivos que son definidos, documentados, utilizados, entrenados, medidos, reforzados y mejorables.

Los proyectos en las organizaciones de nivel 2 tienen instalados controles básicos de administración software. Los compromisos realistas de proyecto son cumplidos basados en los resultados observados en proyectos previos y en los requerimientos del proyecto actual. El administrador de software para un proyecto controla los costos, cronogramas y funcionalidad del software, los problemas en el cumplimiento de los compromisos son identificados a medida que surgen. Los requerimientos de software y los productos de trabajo desarrollados para satisfacerlos poseen un lineamiento sólido y su integridad está controlada. Los estándares del proyecto de software están definidos, y la organización asegura que son fielmente seguidos. El proyecto de software trabaja con sus subcontratantes, si existiera alguno, para establecer una relación cliente proveedor efectiva.

Los procesos podrían diferir entre proyectos en una organización de nivel 2. El requerimiento organizacional para alcanzar el nivel 2 es que existan políticas de nivel organizacional que guíen a los proyectos en el establecimiento de los procesos apropiados de administración.

La capacidad del proceso de software de las organizaciones de nivel 2 puede ser resumido como disciplinado porque la planificación y el control del proyecto de software es estable y los sucesos anteriores pueden ser repetidos. El proceso del proyecto está bajo un control efectivo del sistema de administración de proyecto. Siguiendo planes realistas basados en el desempeño de proyectos previos.

**2.3- Nivel 3: Nivel Definido** En el Nivel Definido el proceso (o los procesos) estándares para el desarrollo y mantenimiento de software está documentado y

usado por toda la organización. Referido en el CMM como el *proceso de software estándar de la organización*, este proceso estándar incluye ingeniería de software y procesos de administración y los integra en un todo coherente. Los procesos establecidos en el nivel 3 son usados (y cambiados, apropiadamente) para ayudar a los administradores de software y al staff técnico a realizarlas más efectivamente. La organización explota las prácticas efectivas de ingeniería de software cuando estandariza sus procesos de software. Un grupo dentro de la organización es responsable de las actividades de los procesos de software. Un programa de entrenamiento empresarial u organizacional es implementado para asegurar que el staff y los administradores tienen los conocimientos y destrezas requeridos para satisfacer sus roles asignados.

Los proyectos se ajustan a los procesos de software estándares de la organización para desarrollar su propio proceso de software definido, el cual se adapta a las características únicas del proyecto. Este proceso ajustado está referido en el CMM como el *proceso de software definido del proyecto*. Este es el proceso usado en la realización de las actividades de proyecto. Un proceso de software definido contiene un conjunto coherente e integrado de procesos de administración y de ingeniería de software bien definidos. Un proceso bien definido puede ser caracterizado como aquel que incluye legibilidad, criterio, entradas, estándares y procedimientos para realizar el trabajo, mecanismos de verificación, salidas, y criterios de completitud. Como el proceso de software está bien definido, la administración tiene una buena visión interna del progreso técnico del proyecto.

La capacidad del proceso de software de las organizaciones de nivel 3 puede ser resumida como estándar y consistente porque las actividades de ingeniería de software y de administración son estándares y repetibles. Dentro de la líneas de producción establecidas, el costo, cronograma y funcionalidad están bajo control, y la calidad del software es verificada. Esta capacidad de proceso está basada en un entendimiento común y organizacional de las actividades, roles, y responsabilidades en un proceso de software definido.

## **2.4- Nivel 4: Nivel Administrado**

En el Nivel Administrado, la organización establece objetivos de calidad cuantitativos para los productos y procesos de software. La productividad y calidad son medidas por actividades de proceso de software importantes a través de todos los proyectos como parte de un programa de medición organizacional. Una base de datos de procesos de software organizacional es usada para recolectar y organizar los datos disponibles de los procesos de software definidos de los proyectos. Los procesos de software son instrumentados con medidas bien definidas y consistentes. Estas medidas establecen los fundamentos cuantitativos para evaluar los procesos y productos de software de los proyectos.

Los proyectos alcanzan el control sobre sus productos y procesos disminuyendo la variación del desempeño de sus procesos para mantenerse dentro de los límites cuantitativos aceptables. Las variaciones cuantitativas del desempeño de proceso puede ser distinguida de la variación aleatoria (ruido), particularmente dentro de las líneas de

producto establecidas. Los riesgos involucrados al mover la curva de aprendizaje del dominio nuevo de una aplicación son conocidos y administrados cuidadosamente.

La capacidad del proceso de software de las organizaciones de nivel 4 puede ser resumida como cuantificable y predecible porque el proceso es medido y opera dentro de límites cuantitativos. Este nivel de capacidad de proceso permite a una organización predecir tendencias en la calidad del proceso y del producto dentro de estos límites cuantitativos. Como el proceso es estable y medido, cuando algunas circunstancias especiales suceden, la "causa especial" de la variación puede ser identificada y localizada. Cuando los límites predefinidos son excedidos, se toman acciones para entender y corregir la situación. Los productos de software son de una calidad predeciblemente alta.

## **2.5- Nivel 5: Nivel Optimizado**

En el Nivel Optimizado, toda la organización está focalizada en la mejora continua de procesos. La organización tiene la intensión de identificar proactivamente debilidades y fortalezas del proceso, con el objetivo de prevenir los defectos. Los datos en la efectividad de los procesos de software son utilizados para realizar análisis de costo/beneficios de nuevas tecnologías y de cambios propuestos a los procesos de software de la organización. Las innovaciones que aprovechan las mejores prácticas de ingeniería de software están definidas y transferidas por toda la organización.

Los grupos de software en las organizaciones de nivel 5 analizan defectos para determinar su causa. Estos grupos evalúan los procesos de software para prevenir la recurrencia de tipos conocidos de defectos y distribuir lecciones aprendidas por toda la organización.

Pérdida crónica, en la forma de rehacer trabajo, puede ser encontrada en cualquier sistema simplemente por variación aleatoria. Esfuerzos organizados para eliminar esta pérdida da como resultado cambios en el sistema, esto es, mejorar los procesos cambiando las "causas comunes" de las ineficiencias para prevenir las ocurrencia de pérdidas. Esto es verdad para todos los niveles de madurez, pero es el objetivo del nivel 5.

La capacidad del proceso de software de las organizaciones de nivel 5 puede ser caracterizada como mejora continua porque las organizaciones de nivel 5 están continuamente interesadas en la mejora del rango de capacidad de sus procesos, incluso mejorando el desempeño de los procesos de sus proyectos. Las mejoras se logran por avances incrementales en el proceso existente y por innovaciones usando nuevas tecnologías y métodos. Las mejoras de tecnologías y métodos están planeadas y administradas como actividades de negocios ordinarias.

## **3.- Salteando Niveles de Madurez**

CMM identifica los niveles por los cuales una organización debe evolucionar para establecer una cultura de excelencia en la Ingeniería de

Software. Cada nivel de madurez del CMM tiene los fundamentos necesarios para construir las bases de siguiente nivel, saltar niveles generalmente es contraproducente. Las organizaciones pueden aplicar mejoras específicas en los procesos en cualquier momento, aún antes de estar preparados para avanzar al nivel en que dicha práctica es recomendada. Sin embargo, las organizaciones deben entender que la estabilidad de estas mejoras lleva un riesgo asociado dado que las bases para la implantación exitosa no se han conseguido. Los procesos que no tengan una base apropiada pueden fallar en el momento que más son necesarios (situaciones de mayor exigencia).

Debe reconocerse que los esfuerzos de mejora de los procesos deben focalizarse en las necesidades de la organización en el marco del entorno de sus negocios y que prácticas de alto nivel pueden satisfacer las necesidades actuales de una organización o proyecto.

**4.- Visibilidad en el Proyecto de Software** Cada nivel del CMM aumenta la visibilidad dentro del proceso del software para ambos, administrador y el staff de ingeniería. Los ingenieros de software tienen una visión detallada en el estado del proyecto porque son los primeros que reciben información del estado y del desempeño del proyecto. Sin embargo, en grandes proyectos la visión está usualmente delineada por su experiencia personal en su área de responsabilidad.

Aquellos que no poseen información temprana y que no están en contacto con el proyecto, como los administradores seniors, carecen de visibilidad dentro de los procesos del proyecto y por lo tanto confían únicamente en revisiones periódicas para la información que requieren para monitorear el proceso.

**4.1- Nivel 1** El proceso de software es una entidad amorfa, una caja negra, y la visibilidad en los procesos del proyecto es limitado. Dado que la secuencia de actividades no está bien definida, los administradores tienen una extrema dificultad para definir los tiempos y el estado del progreso del proyecto y sus actividades. Los requerimientos fluyen de forma descontrolada dentro del proceso de software. El desarrollo de software es ve frecuentemente como magia, especialmente por los administradores que no están familiarizados con el software. Los clientes pueden determinar si el producto cumplirá con los requerimientos cuando le sea entregado.

**4.2- Nivel 2** Los requerimientos de clientes son controlados, y las prácticas básicas de la administración de proyecto han sido establecidas. Estos controles le dan mayor visibilidad dentro del proyecto al administrador en ciertas ocasiones. El proceso de construcción de software puede ser visto como una sucesión de cajas negras que permiten al administrador tener visibilidad en los hitos del proyecto mientras las actividades fluyen entre cajas. Aún así, el administrador puede no saber los detalles de lo que sucede en las cajas. Los productos finales del proceso y los puntos de control para confirmar el avance están identificados y son conocidos. La administración reacciona a los problemas a medida que ocurren. El cliente puede revisar el producto en puntos de control definidos durante el proceso de software.

**4.3- Nivel 3** La estructura interna de las cajas, es decir, las tareas en el proceso de

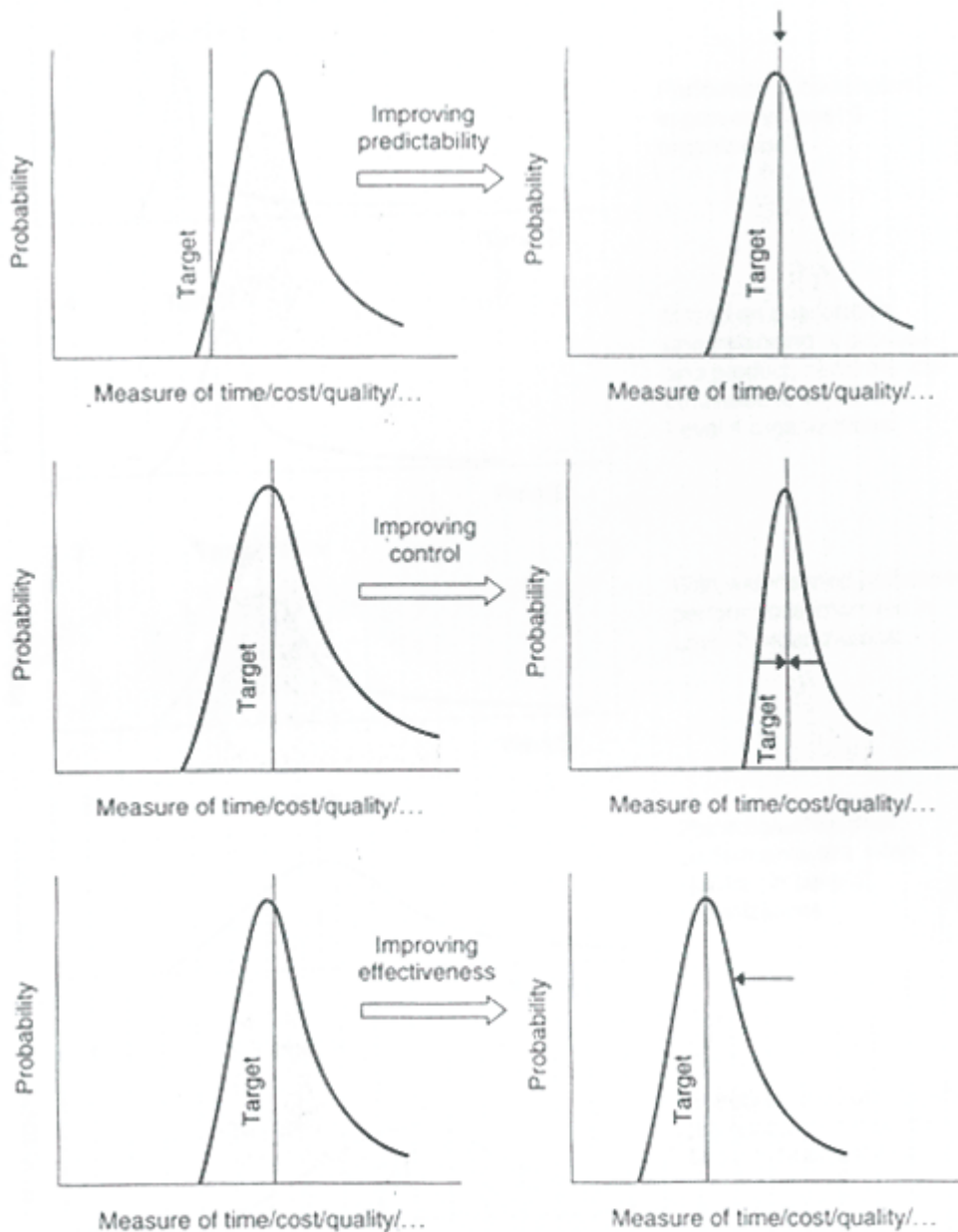


software del proyecto son visibles. La estructura interna representa la forma en que el proceso de software estándar de la organización se ha aplicado a proyectos específicos. Administradores e ingenieros entienden sus roles y responsabilidades en el proceso y como sus actividades interactúan en el nivel adecuado de detalle. La administración está preparada para riesgos que puedan ocurrir. El cliente puede obtener actualizaciones del estado del proyecto acertadas dado que los procesos definidos brindan gran visibilidad en las actividades del proyecto.

**4.4- Nivel 4** Los procesos de software definidos se instrumentan y controlan cuantitativamente. Los administradores son capaces de medir el progreso y los problemas. Tiene un objetivo, una base cuantitativa para tomar decisiones. Su habilidad para predecir las salidas crece de forma estable y precisa mientras que la variabilidad en el proceso disminuye. El cliente puede establecer un entendimiento cuantitativo de la capacidad de los procesos y del riesgo antes de que el proyecto comience.

**4.5- Nivel 5** Nuevas y mejoradas formas de construir el software se prueban continuamente, de forma controlada, para mejorar la productividad y calidad. La disciplina y el cambio son una forma de vida y las actividades defectuosas son identificadas y reemplazadas o revisadas. La visibilidad se extiende más allá de la existencia de los procesos y dentro de los efectos de potenciales cambios a los mismos. Los administradores son capaces de estimar y de hacer un seguimiento cuantitativo del impacto y de la efectividad del cambio. El cliente y la organización de software continúan trabajando juntos para establecer una relación fuerte del tipo cliente-proveedor.

**5.- Predicción de la performance** La madurez del proceso de una organización de software ayuda a predecir la habilidad de un proyecto para alcanzar sus objetivos. Los proyectos en organizaciones de nivel 1 experimentan una amplia variación en alcanzar los costos, cronogramas de tiempos, funcionalidad y calidad.



### Mejora del Control, la Predictibilidad, y la Efectividad

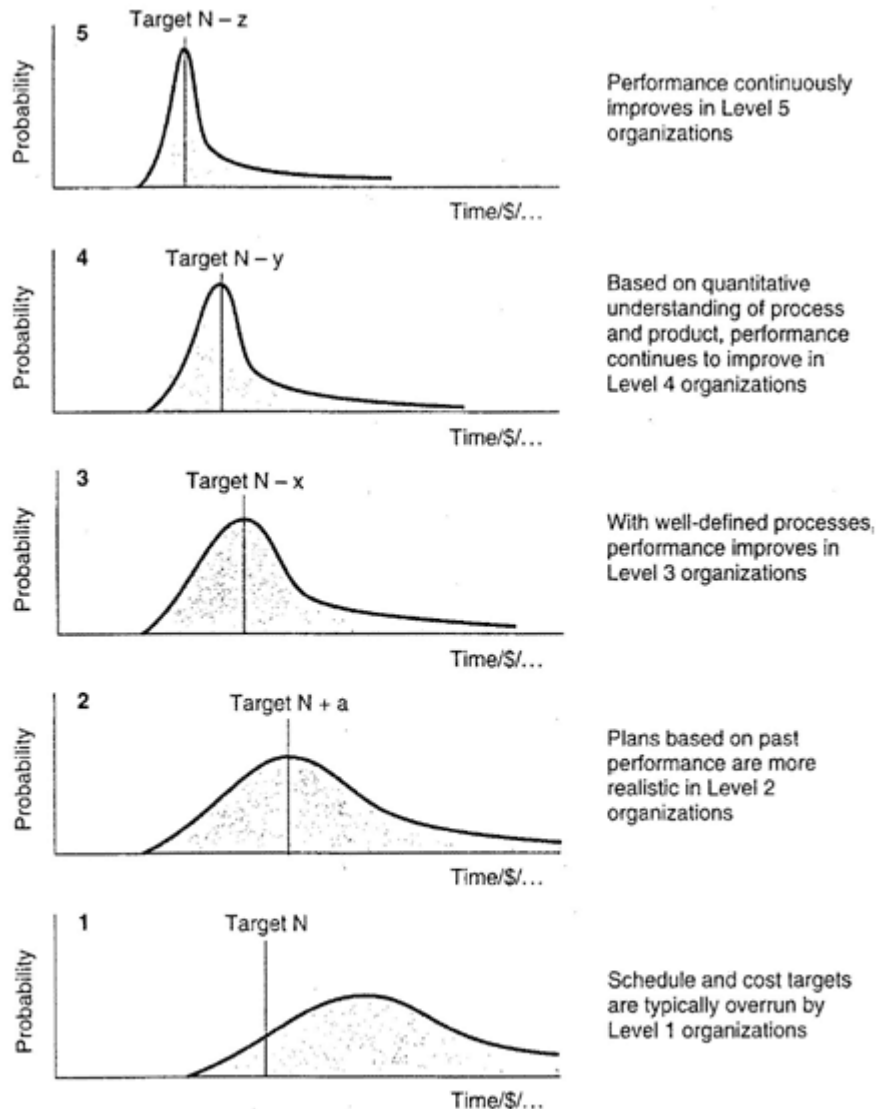
La figura ilustra los tipos de mejoras esperadas en predicción, control y efectividad en forma de una función de densidad de probabilidades para la performance probable para un proyecto en particular con respecto a sus objetivos que pueden ser cronograma de tiempo, costo, calidad, etc.

La primer mejora esperada durante el proceso de madurez de la organización es en predicción. Al incrementarse la madurez, la diferencia entre los resultados esperados y los resultado obtenidos disminuye en los

procesos. Por ejemplo, organizaciones de nivel 1 a menudo se exceden en los cronogramas y fechas de entrega por un amplio margen, mientras que organizaciones de niveles de madurez mayores son capaces de alcanzar las fechas esperadas con una gran precisión.

La segunda mejora es con respecto al control. Al incrementar la madurez la variabilidad de los resultados actuales entorno a los resultados esperados disminuye.

La tercer mejora es con respecto a la efectividad. Los resultados esperados mejoran al incrementarse la madurez de la organización. Estos es, al incrementarse la madurez de la organización de software, los costos disminuyen, los tiempos de desarrollo se tornan más cortos y la productividad y calidad aumentan. En una organización de nivel 1 el tiempo de desarrollo puede extenderse debido a la cantidad de errores que se deben corregir. En contraste organizaciones con niveles de maduración más alta tienen mayor efectividad en sus procesos y han reducido el costo de rehacer el trabajo, resultando en menores tiempo de desarrollo.



## Capacidad del Proceso Según el Nivel de Madurez

Las mejoras en predecir los resultados del proyecto, que se representa en la figura anterior, asume que las salidas del proyecto de software son más predecibles eliminado el ruido del proceso. Sistemas si precedentes complican el cuadro, dado que nuevas tecnologías y aplicaciones disminuyen la capacidad de un proceso incrementando la variabilidad. Aún en el caso de un sistema sin precedentes la administración y las prácticas de ingeniería características de organizaciones más maduras ayudan a identificar y tratar los problemas más temprano en el ciclo de desarrollo. En algunos casos un proceso maduro significa que proyectos que fallarán se identifican antes en el proceso de software.

## **6.- La estructura del CMM**

El CMM representa un marco de trabajo para realizar mejoras recomendadas para organizaciones de software que quieren incrementar la capacidad de sus procesos de software. El CMM es un modelo descriptivo en el sentido que describe atributos claves que se esperan para caracterizar una organización en un nivel de madurez particular. Es un modelo normativo en el sentido que prácticas detalladas caracterizan los tipos de comportamiento que se esperan de una organización que realiza proyectos a gran escala.

El CMM describe lo que normalmente esperamos en un proceso de software sin tener en cuenta como está implementado.

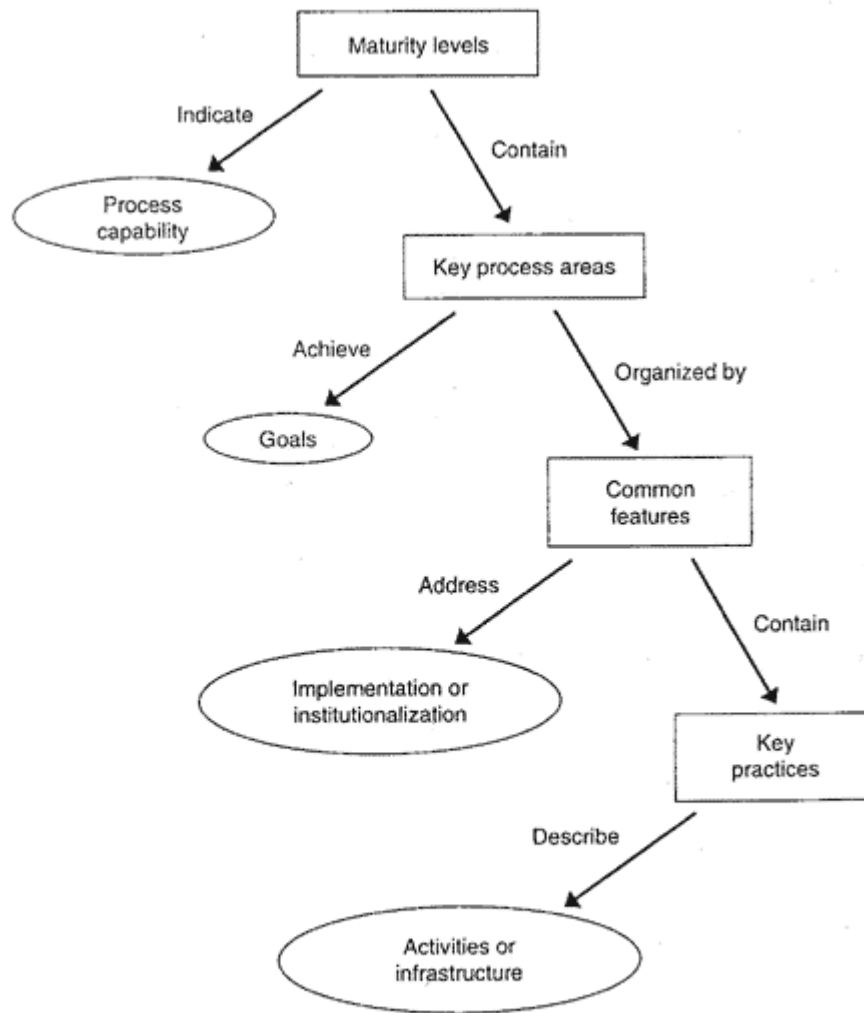
Esta elaboración operacional del CMM se designa para soportar las formas en que será usado incluyendo las siguientes 5:

- 1- Determinar equipos que utilizarán el CMM para identificar fortalezas y debilidades en la organización;
- 2- Equipos de evaluación usarán el CMM para identificar los riesgos de seleccionar entre diferentes contratistas ganar licitaciones y para monitorear contratos;
- 3- La alta administración utilizará el CMM para entender las actividades necesarias para poner en marcha la mejora del proceso de software en su organización;

El staff técnico y los grupos de mejora de procesos como el SEPG (Software Engineering Process Group), utilizarán el CMM como una guía para ayudarlos a definir y mejorar los procesos de software en su organización.

### **6.1- La estructura interna de los Niveles de Madurez**

Cada nivel de madurez ha sido descompuesto en partes constituyentes. Con excepción del nivel 1, la descomposición de cada nivel de madurez varía de resúmenes abstractos de cada nivel a sus definiciones operacionales en prácticas claves. Cada nivel de madurez se compone de varias áreas de proceso clave.



## La Estructura del CMM

### 6.2- Niveles de Madurez

Un nivel de madurez es un esfuerzo evolutivo bien definido para alcanzar la madurez del proceso de software. Los cinco niveles de madurez proveen un alto nivel de estructura del CMM.

Cada nivel de madurez indica un nivel de capacidad. Dado que la capacidad del proceso de software describe el rango de los resultados esperados que se pueden alcanzar siguiendo un proceso de software, la capacidad del proceso de software de una organización provee un medio para predecir la salida más probable para el siguiente proyecto de software que la organización encare.

## 7.- Areas claves de proceso

Con excepción del nivel 1, cada nivel de madurez se descompone en varias áreas clave de proceso que indican cuando una organización se debe focalizar para mejorar sus procesos de software. Las áreas clave de proceso identifican los problemas que se deben tratar para alcanzar un nivel de madurez. Una organización que está en el nivel 3 ha alcanzado todas las áreas clave de proceso de los niveles 2 y 3.

Cada área clave de proceso identifica un grupo de actividades relacionadas que, cuando se realizan colectivamente, se consiguen un conjunto de metas consideradas importantes para mejorar la capacidad del proceso. Las áreas clave de proceso han sido definidas de manera que se refieran a un único nivel de madurez. El camino para alcanzar las metas de un área clave de proceso puede diferir en proyectos con diferencias en el dominio de aplicación o el ambiente. Sin embargo, todas las metas de un área clave de proceso deben ser alcanzadas por la organización para satisfacer esa área clave de proceso.

El adjetivo "clave" implica que algunas áreas de proceso y procesos no son claves para alcanzar un nivel de madurez. El CMM no describe todas las áreas de proceso que involucra el desarrollo y mantenimiento de software. Sólo aquellas áreas de proceso identificadas como claves son determinantes de la capacidad del proceso.

Las áreas clave de proceso pueden ser consideradas los requerimientos para alcanzar un nivel de madurez. Para alcanzar un nivel de madurez, las áreas claves para ese nivel (y para niveles más bajos) deben satisfacerse y los procesos deben institucionalizarse.

Las metas de cada área clave de proceso deben resumirse en prácticas clave y pueden usarse para determinar cuándo una organización o proyecto ha implementado eficientemente el área clave de proceso. Las metas significan la visibilidad, los límites y la intención de cada área clave de proceso. Al adaptar las prácticas clave de un área clave de proceso a un proyecto o situación de una organización específica, las metas se pueden usar para determinar si la adaptación es razonable o no. De forma similar, cuando se evalúan alternativas de implementación de un área clave de proceso, las metas se pueden utilizar para determinar si la alternativa satisface la intención del área clave de proceso.

### 7.1- Areas claves de proceso en el Nivel 2

Las áreas clave de proceso en el nivel se focalizan en el proyecto de software que se relaciona con establecer la administración y control básico del proyecto.

- El propósito de la administración de requerimientos es establecer

un entendimiento común entre el cliente y los requerimientos del proyecto de software del cliente que deben ser satisfechos por el proyecto de software. Este acuerdo con el cliente es la base para planificar y administrar el proyecto de software.

Un entendimiento de los requerimientos es necesario para construir software que satisfaga al cliente. Dado que los requerimientos del cliente con frecuencia evolucionan y cambian, documentar y controlarlos es un pre requisito para usarlos como base para estimar, planear, realizar, y controlar las actividades del proyecto de software a través de todo el ciclo de vida del software.

- El propósito del **planeamiento del proyecto de software** es establecer planes razonables para realizar la ingeniería de software y para administrar el proyecto. Los planes razonables se basan en desarrollar estimaciones realistas para realizar el trabajo y establecer los compromisos necesarios. Estos comienzan con una declaración del trabajo y las restricciones y las metas que definen y limitan el proyecto de software. El proceso de planeación del software incluye pasos para estimar el tamaño del software y los recursos necesarios, para producir un cronograma, para identificar y estimar los riesgos, y para negociar los compromisos. El plan se documenta y se mantiene como una herramienta necesaria para administrar el proyecto de software.
- El propósito del **control de proyectos de software** es establecer una adecuada visibilidad del progreso real para que la administración pueda tomar acciones correctivas cuando la performance del proyecto se desvía significativamente de lo planeado. La administración del proyecto de software se debe basar en el plan de desarrollo de software. La administración involucra controlar y revisar los resultados contra el plan y tomar acciones correctivas cuando sean necesarias, basándose en los resultados reales. Estas acciones pueden incluir: revisar el plan de desarrollo de software para que refleje los resultados actuales, replanificar el trabajo restante, y/o tomar acciones para mejorar la performance.
- El propósito de la **administración de subcontratos de software** es seleccionar subcontratantes de software calificados y administrarlos efectivamente. La selección de subcontratantes se basa en la habilidad para realizar el trabajo, pero muchos factores contribuyen a la decisión de subcontratar una parte del mismo. Los subcontratantes pueden seleccionarse basándose en alianzas estratégicas de negocio, así como capacidad de proceso y consideraciones técnicas. El trabajo a realizar por el subcontratante y los planes para el trabajo son documentados, y el principal contratista monitorea la performance contra esos planes.
- El propósito del **aseguramiento de la calidad del software** es proveer una administración con apropiada visibilidad dentro del



proceso siendo utilizado por el proyecto de software y los productos que están siendo contruidos. Esta visibilidad se alcanza revisando y auditando los productos de software y las actividades para verificar que cumplen con los estándares y procedimientos aplicables.

- El propósito de la **administración de configuración de software** es establecer y mantener la integridad de los productos del proyecto de software a través del ciclo de vida del proyecto. La integridad de los productos de trabajo se alcanza identificando la configuración del software en ciertos puntos dados, controlando sistemáticamente los cambios a la configuración, y manteniendo la integridad de la configuración a través del ciclo de vida del software. Las líneas guía del software son mantenidas en una librería de líneas guía a medida que se van desarrollando. Los cambios a las líneas guía y a la versión del producto de software construido, son sistemáticamente controladas y la configuración es auditada

## 7.2- Areas claves de proceso en el Nivel 3

Las áreas clave de proceso en el nivel 3 tratan problemas del proyecto y la organización, mientras ésta establece una infraestructura que institucionaliza una ingeniería de software y administración de procesos efectiva a través de todos los proyectos.

- El propósito del **foco del proceso organizacional** es establecer la responsabilidad organizacional por las actividades del proceso de software que mejoran la capacidad del proceso de software de toda la organización. El crecimiento constante del proceso involucra desarrollar y mantener un entendimiento de la organización y de los proyectos de procesos de software y coordinar las actividades para de terminar, desarrollar, mantener, y mejorar esos procesos. La palabra "coordinar" se eligió deliberadamente para hacer énfasis en que las personas que implementan los procesos deben estar íntimamente involucradas en su definición y mejora aunque la organización provea compromisos a largo plazo y recursos para coordinar el desarrollo y mantenimiento de los procesos de software a través de actuales y futuros proyectos de software.
- El propósito de la **definición de procesos organizacionales** es desarrollar y mantener un conjunto de datos de los procesos de software utilizables que mejoran la performance del proceso a través del proyecto y proveen una base para definir datos significativos para la administración cuantitativa de procesos. Estos datos pueden ser recolectados de muchas formas; por ejemplo, las descripciones de los ciclos de vida de software pueden ser una parte integral de los procesos de software estándar de la organización. La taxonomía provista en esta área clave delimita los aspectos de la definición de procesos que necesitan ser tratados.

- El propósito del **programa de capacitación** es desarrollar las habilidades y el conocimiento de los individuos para que puedan cumplir con sus roles efectiva y eficientemente. Capacitación o entrenamiento es una responsabilidad organizacional, pero los proyectos de software son responsables de identificar sus necesidades en cuanto a habilidades y de proveer el entrenamiento necesario cuando las necesidades del proyecto son únicas. Un programa de entrenamiento comienza por identificar necesidades de entrenamiento de la organización, del proyecto, y del individuo, para así desarrollar un entrenamiento adecuado a las necesidades identificadas. Estas necesidades pueden ser específicas para el proyecto o el individuo en un momento particular, pero el entrenamiento requerido puede identificarse basándose en los roles y responsabilidades especificadas en el proceso de software estándar de la organización. Algunas habilidades son efectiva y eficientemente impartidas a través de vehículos informales, por ejemplo, la tutoría. Otras habilidades necesitan vehículos de entrenamiento más formales, como por ejemplo, entrenamiento en aulas.
- El propósito de la **administración integral de software** es integrar la ingeniería de software y la administración de actividades en un coherente, y definido proceso de software que es adaptado al proceso de software estándar de la organización. Esta adaptación se basa en el entorno de negocios y las necesidades técnicas del proyecto.  
Esta área clave de proceso es la evolución de la planeación del proyecto de software y del control del proyecto de software en el nivel 2 para tomar ventaja de la infraestructura organizacional establecida en el nivel 3.
- El propósito de la **ingeniería de productos** de software es llevar a cabo consistentemente un proceso ingenieril bien definido que integra todas las actividades de ingeniería necesarias para producir software correcto y consistente de manera efectiva y eficiente. La ingeniería de productos de software describe las actividades técnicas del proyecto, por ejemplo, análisis de requerimientos, diseño, codificación y prueba.  
Estos procesos de ingeniería involucran documentar los productos de software y mantener la consistencia y facilidad de seguimiento entre ellos. Esto es necesario para asegurar una transición controlada entre las etapas del ciclo de vida del software y para proveer productos de software de alta calidad al cliente.
- El propósito de la **coordinación entre grupos**, es establecer un medio para que el grupo de ingeniería de software participe activamente junto con otros grupos de ingeniería de manera que el proyecto satisfaga las necesidades del cliente efectiva y eficientemente.  
El grupo de ingeniería de software debe trabajar pro activamente con los otros grupos de ingeniería para determinar los

requerimientos y objetivos a nivel sistema. Idealmente esto conformaría un equipo de producto integrado o alguna forma de ingeniería concurrente. En cualquier caso las interfaces e interacciones entre los grupos deben ser planeadas y administradas para asegurar la calidad e integridad del sistema en su totalidad. Todos los grupos de ingeniería deben conocer el estado y los planes de los demás grupos, y los asuntos relativos al sistema y a los grupos deben recibir atención apropiada.

- El propósito de las **revisiones puntuales** es eliminar defectos de los productos de software eficientemente y en etapas tempranas. Un efecto importante es desarrollar un mejor entendimiento de los productos de software y de los defectos que deben ser prevenidos. La revisión es un método de ingeniería importante y efectivo, que se puede implementar mediante inspecciones, revisiones estructuradas y otros métodos.

### **7.3- Áreas claves de proceso en el Nivel 4**

Las áreas clave de proceso en el nivel 4 se focalizan en establecer un entendimiento cuantitativo, tanto del proceso de software como de los productos que se construyen.

- El propósito de la **administración cuantitativa de procesos** es controlar la performance de procesos del proyecto de software de manera cuantitativa. La performance de procesos de software representa los resultados reales alcanzados al seguir un proceso de software. Existirán variaciones aleatorias (ruidos) en cualquier proceso. Con un proceso estable, la performance se encuentra generalmente dentro de intervalos conocidos. Cuando la performance cae afuera de esos intervalos, se necesita identificar esa "causa especial" de la variación y, de ser apropiado, corregir las circunstancias que llevaron a que ocurra esa variación. El resultado de satisfacer esta área clave es un proceso que permanece estable y cuantitativamente predecible.
- El propósito de la **administración de calidad de software** es desarrollar un entendimiento cuantitativo de la calidad de los productos de software del proyecto y alcanzar objetivos de calidad específicos. Los objetivos cuantitativos se establecen para los productos de software, basándose en las necesidades de la organización, el cliente, o el usuario final. Para que estos objetivos se alcancen, la organización establece estrategias y planes, y el proyecto ajusta específicamente sus procesos de software definidos para alcanzar los objetivos de calidad. La administración de calidad de software está focalizada en los productos, mientras que la administración cuantitativa de procesos está focalizada en los procesos.

## 7.4- Areas claves de proceso en el Nivel 5

Las áreas de procesos claves en el nivel 5 cubren los temas que las organizaciones y proyectos deben localizar para implementar mejoras continuas y medibles de procesos de software.

- El propósito de la **Prevención de Defecto** es identificar las causas de los defectos y prevenirlos de su reincidencia. El proceso de software analiza los procesos, identifica sus causas y toma acciones para prevenirlos de su reincidencia. Muchas veces esto involucra el cambio del proceso de software definido en el proyecto. El análisis causal resultaría también en cambiar elementos del proceso estándar de la organización para controlar las causas comunes de la variación. La Prevención de Defectos es un mecanismo de mejora incremental del proceso de software en una forma evolutiva.
- El propósito de la **Administración de Cambio de Tecnología** es identificar nuevas tecnologías beneficiosas (herramientas, métodos y procesos) y transferirlas a la organización de una manera ordenada. Un foco en la transición de la tecnología implica identificar, seleccionar, y evaluar nuevas tecnologías e incorporar tecnologías efectivas a la organización. El objetivo es mejorar la calidad de software, incrementar la productividad y reducir el tiempo de desarrollo del producto. El resultado de este foco es la innovación eficiente en un mundo cambiante. La Administración del Cambio de Tecnología se refiere a la mejora del proceso de software de una manera revolucionaria.
- El propósito de la **Administración del Cambio del Proceso** es mejorar continuamente el proceso de software usado en la organización con la intención de mejorar la calidad del software, aumentando la productividad y reduciendo el tiempo para el desarrollo del producto. La mejora continua del proceso incluye la definición de los objetivos de la mejora de procesos y, con la ayuda del administrador senior, identificar proactiva y sistemáticamente, evaluando, e implementando mejoras al proceso de software estándar de la organización y al proceso de software definido del proyecto. La Administración del Cambio del Proceso se focaliza en la entrega apropiada de cambios incrementales evolutivos y cambios innovadores y revolucionarios.

## 8.- Prácticas claves

Cada área de proceso clave está descrita en términos de prácticas claves. Las **prácticas claves** describen las actividades y la infraestructura que contribuye más a la implementación e institucionalización efectiva del área de proceso clave. Las prácticas específicas a ser ejecutadas en cada área de proceso clave se incluirán como niveles más altos de alcances organizacionales de madurez del proceso. Por ejemplo, muchas de las capacidades y estimaciones del proyecto descritas en el área de proceso clave Planificación del Proyecto de Software en el nivel 2 debe incluir manipulación de datos adicionales al proyecto disponible en el nivel 3,

como está descrito en la Administración de Software Integrado.

Las prácticas claves describen “que” hay que hacer, pero no deberían ser interpretadas por “como” el proceso debería estar implementado. Prácticas alternativas podrían cumplir los objetivos del área de proceso clave. Las prácticas claves deberían ser interpretadas racionalmente para juzgar si los objetivos del área de proceso clave son alcanzados, aunque tal vez de forma diferente, efectivamente.

Cada práctica clave consiste de un único enunciado, casi siempre seguidos por una descripción más detallada, la que incluiría ejemplos y elaboración. Estas prácticas claves describen la infraestructura y actividades que más contribuyen a la implementación en institucionalización del software de un área de procesos claves. Subprácticas más detalladas son provistas frecuentemente en las prácticas claves como guía en la interpretación e implementación adecuada de la práctica clave. Ejemplos, definiciones, y referencias a otras prácticas son provistas también según sean ampliadas.

## 9.- Características comunes

Por conveniencia las prácticas que describen las áreas de procesos clave está organizadas por características comunes. Las características comunes son atributos que indican si la implementación e institucionalización de un área de proceso clave es efectiva, repetible y perdurable. Hay 5 características comunes:

- Compromiso de realización: el compromiso de realización describe las acciones que las organizaciones deben tomar para asegurar que el proceso está establecido y durará. El compromiso de realización típicamente incluye el establecimiento de políticas organizacionales y liderazgo.
- Habilidad de realización: la habilidad de realización describe las precondiciones que deben existir en el proyecto o la organización para implementar competentemente el proceso de software. La habilidad de realización técnicamente incluye recursos, estructuras organizacionales y entrenamiento.
- Actividades realizadas: las actividades realizadas describen las actividades, roles y procedimientos necesarios para implementar un área de proceso clave. Las actividades realizadas incluyen típicamente el establecimiento de planes y procedimientos, realizar el trabajo, su seguimiento, y tomar medidas correctivas según sea necesario.
- Mediciones y análisis: las mediciones y análisis describen las prácticas básicas de medición que son necesarias para determinar el estado relacionado al proceso. Estas mediciones son usadas para controlar y mejorar el proceso. Las mediciones y el análisis típicamente incluyen ejemplo de las mediciones que pueden ser tomadas.

- Verificando implementación: verificando implementación describe los pasos para asegurar que las actividades son llevadas a cabo conforme a los procesos que han sido establecidos. Verificando implementación típicamente incluye revisiones y auditorías por la administración y el aseguramiento de la calidad de software.

Las prácticas en la característica común Actividades Realizadas describe qué debe ser implementado para establecer la capacidad de un proceso. Las otras prácticas tomadas como un todo, forman la base por la cual una organización puede institucionalizar las prácticas descritas en la característica común anterior.

## **10.- El Modelo GQM (Goal Question Metrics)**

El paradigma de Goal-Question-Metric (GQM) a sido probado de ser particularmente efectivo en aproximarnos para seleccionar e implementar métricas adecuadas para nuestros objetivos.

Para usar el GQM, debemos expresar los objetivos globales de la organización. La organización puede abarcar toda la corporación o empresa, o simplemente un grupo de aseguramiento de la calidad al cual estamos asignados; el alcance de las métricas y los objetivos depende de la persona que los realiza y de sus necesidades. Luego debemos generar preguntas cuyas respuestas nos dirán si hemos alcanzado los objetivos. Finalmente, debemos analizar en términos de qué medidas son necesarias para responder cada pregunta realizada.

Una vez que tenemos la lista de las posibles mediciones, debemos evaluar si la organización es capaz de proveernos información útil para esas mediciones. Por ejemplo, si los desarrolladores no pueden decirnos cuando finalizará el diseño y comenzará la codificación, entonces medir la actividad de diseño será imposible. Para ayudarnos a entender cuando las medidas son apropiadas para nuestra organización, se introduce el concepto de madurez de procesos. Cuanto más maduro sea nuestro proceso, mayor será la visión que tengamos del mismo, así como también mejor serán las mediciones.

Una introducción al GQM provee un entorno que involucra tres pasos:

1. Lista de los objetivos principales para el desarrollo y el mantenimiento del proyecto.
2. Derivar de cada objetivo preguntas que deben ser contestadas para determinar si los objetivos han sido alcanzados.
3. Decidir que medidas deben realizarse de modo de ser capaz de responder las preguntas adecuadamente.

Derivando las medidas de esta forma, se ve más claramente cómo utilizar los datos resultantes.

### 10.1- Ejemplo

## [10.2- Plantillas para la definición de objetivos](#)

## [10.3- Medidas y mejoramiento de procesos. CMM y GQM](#)

### **10.1- Ejemplo**

Supongamos que nuestro objetivo global es evaluar la efectividad de la utilización de los estándares en la codificación. Esto es, queremos saber si el código producido por un determinado estándar es superior de alguna forma al código producido sin dicho estándar. El árbol de GQM para este caso sería el siguiente.

Para decidir si el estándar es efectivo debemos hacernos una serie de preguntas claves. Primero, es importante conocer quien está utilizando estándares, para poder comparar la productividad de los programadores que utilizan estándares con los que no lo hacen. Del mismo modo, probablemente nos gustaría comparar la calidad del código producido con el estándar con la calidad de un código no estándar.

Una vez que tenemos las preguntas identificadas, debemos analizar cada una de estas para determinar que medidas son necesarias para responder la pregunta. Por ejemplo, para entender quien está utilizando estándares es necesario saber que proporción de los programadores está utilizando estándares. Sin embargo, es también importante tener un perfil de programadores experimentados, teniendo en cuenta cuanto tiempo han trabajado con estándares, los entornos, los lenguajes y otros factores que ayudan a evaluar la efectividad de los estándares.

De esta manera, generamos solamente aquellas métricas que están relacionadas con el objetivo. Notar que en muchos casos, varias métricas son necesarias para responder una sola pregunta. El objetivo nos provee un propósito para la recolección de datos, y las preguntas nos dice como utilizar esos datos.

Lo que no es evidente para el árbol de GQM es el modelo necesario para combinar las medidas de forma tal de responder las preguntas. Por ejemplo, el árbol anterior sugiere la medición de la productividad de los programadores; este atributo puede ser medido en términos del esfuerzo invertido por cada línea de código, pero esta relación no está explícita en el árbol. El árbol tampoco nos dice como medir las líneas de código o los puntos de función. Es necesario un trabajo adicional para definir cada métrica. En el caso de la inexistencia de una medición objetiva, debemos identificar las medidas subjetivas.

Por lo tanto, el modelo de GQM debe ser utilizado con uno o más modelos suplementarios que expresen las relaciones entre las métricas.

Es importante que los objetivos y las preguntas sean comprendidos por quienes las perciben: la productividad como objetivo para el director de proyecto puede ser diferente que para el gerente de departamento y para el director de la corporación. Para auxiliar en la elección de objetivos,

preguntas y métricas, se proveen una serie de puntos a ser analizados o *templates*.

## 10.2- Templates para la definición de objetivos

**Propósito:** para (caracterizar, evaluar, predecir, motivar, etc.) el (proceso, producto, modelo, métrica, etc.) de modo de poder (entenderlo, evaluarlo, dirigirlo, impulsarlo, aprender, mejorarlo, etc.).

**Ejemplo:** Evaluar el proceso de mantenimiento de modo de mejorarlo.

**Perspectiva:** Examinar el (costo, efectividad, exactitud, defectos, cambios, medidas de producto, etc.) desde el punto de vista del (desarrollador, director, cliente, etc.).

**Ejemplo:** Examinar el costo desde el punto de vista del gerente.

**Entorno:** El entorno consiste en lo siguiente: factores del proceso, factores de personas, factores de problemas, métodos, herramientas, contratos, etc.

**Ejemplo:** El personal de mantenimiento son programadores desmotivados quienes tienen un acceso limitado a las herramientas.

El GQM complementa al entorno de medidas de entidades y atributos presentados anteriormente. El resultado del análisis GQM es una colección de medidas relacionadas con el objetivo y el modelo global. Sin embargo GQM no localiza los problemas en las escalas de mediciones, la objetividad o la viabilidad. Por lo tanto el GQM debe utilizarse con mucho cuidado, teniendo en cuenta que los objetivos globales no proveen datos importantes para mejorar nuestros procesos productos y recursos.

## 10.3- Medidas y mejoramiento de procesos. CMM y GQM

Las medidas proporcionan una visión de cómo los procesos, productos, recursos, métodos y tecnologías del desarrollo de software se relacionan unas con otras. Las medidas también pueden ser útiles para:

- Comprender
- Establecer una base de comparación
- Evaluar y predecir

Pero las medidas son útiles solamente en el extenso contexto de la evaluación y mejoramiento. Elegir las métricas, recolectar los datos, analizar los resultados y tomar las acciones apropiadas de un proceso que requiere tiempo y recursos; estas actividades tienen sentido solamente si están dirigidas a mejorar los objetivos específicos.



Algunos procesos de desarrollo están más maduros que otros. Mientras algunas organizaciones tienen claramente definidos los procesos, otras varían significativamente. El Software Engineering Institute (SEI) ha sugerido que existen cinco niveles de madurez de procesos, con un rango entre *inicial*(el menos predecible y controlable) a *repetible, definido, dirigido, y optimizado*(el más predecible y controlable). El SEI distingue un nivel de otro en término de las actividades principales llevadas a cabo en cada nivel.

En el menor nivel de madurez, el proceso no es bien comprendido; a medida que aumenta la madurez del proceso, este es mejor comprendido y está mejor definido. Los cinco niveles de madurez son utilizados para determinar qué medir primero y como planificar un programa de medidas para abarcar los aspectos adicionales del desarrollo y el mantenimiento.

La tabla a continuación presenta los tipos de medidas utilizados en cada nivel de madurez, donde la selección depende de la cantidad de información visible y disponible en cada nivel.

<b>Nivel de madurez</b>	<b>Características</b>	<b>Tipos de métricas utilizadas</b>
5. Optimización	Mejoramiento de la realimentación del proceso	Proceso más la realimentación para cambiar el proceso
4. Gestionado	Proceso de medición	Proceso más la realimentación para controlar
3. Definido	El proceso está definido y establecido	Producto
2. Repetible	El proceso dependiente de sus componentes individuales	Dirección de proyecto
1. Inicial	Desordenado	Base de comparación.

Las medidas en el nivel 1 proveen una base de comparación en la búsqueda de mejoramiento de procesos y productos; las medidas en el nivel 2 se focalizan en la dirección del proyecto, mientras que en el nivel 3 las medidas se concentran los productos intermedios y finales producidos durante el desarrollo. Las medidas en el nivel 4 capturan las características del proceso de desarrollo en sí mismo, permitiendo controlar individualmente las actividades del proceso. Un proceso de nivel 5 es suficientemente maduro y cuidadosamente dirigido, que por medio de las medidas proveen una realimentación para cambiar dinámicamente el proceso durante el desarrollo del proyecto.

El entorno de madurez de procesos se comporta como una guía para expandir y construir un programa de métricas, que no solo tiene ventajas con respecto a la visibilidad y a la madurez, sino que también expande las mejoras en las actividades de los procesos.

El paradigma GQM en relación con la madurez de los procesos, ha sido

utilizado como base para determinadas herramientas, las cuales son utilizadas por los gerentes para diseñar programas de mediciones. GQM nos ayuda a entender porque medimos un atributo, y la madurez de los procesos no sugiere si somos capaces de medir este atributo de una forma significativa. Juntos, proveen un contexto para las mediciones. Sin este contexto, las medidas serían usadas inapropiadamente, dándonos un falso sentido de tranquilidad con nuestros procesos y productos.

## **11.- Organizaciones de Software inmaduras vs maduras**

Establecer objetivos apropiados para el mejoramiento de los procesos requiere un conocimiento previo de la diferencia que existe entre organización de software inmaduras y maduras. En una organización de software inmadura los procesos de software son generalmente improvisados por el personal y sus administradores durante el curso del proyecto. Aún si un proceso de software ha sido especificado, no es rigurosamente seguido. En la organización de software inmadura se espera a que las cosas sucedan para reaccionar, es por esto que los administradores se encuentran focalizados en resolver las crisis inmediatas. Planes y presupuestos son rutinariamente excedidos porque los mismos se encuentran basados en estimaciones realistas. Cuando a estas organizaciones le son impuestas fecha límites para la entrega del producto, las funcionalidad y la calidad son generalmente comprometidas para cumplir con los tiempos.

En una organización inmadura, no existen objetivos bases para analizar la calidad del producto o para resolver problemas con los procesos o productos. Por lo tanto, la calidad del producto es difícil de predecir. Actividades cuyo objetivo es mejorar la calidad como las revisiones o proceso de testing son generalmente eliminados cuando los proyectos tienen tiempos establecidos.

Por el contrario, una organización de software madura posee una amplia habilidad para la administración del desarrollo de software y el mantenimiento de procesos. El proceso de software es correctamente comunicado al personal existente y a los nuevos empleados, y las actividades de trabajo son realizadas de acuerdo a los procesos planeados. Los procesos son utilizables y consistentes con la forma en que se realizan los trabajos. Los procesos definidos son actualizados cuando es necesario, y las mejoras en el desarrollo son controladas a través de pruebas pilotos y/o análisis de costo-beneficio.

Los roles y responsabilidades en los procesos definidos son claros a través del proyecto y de la organización.

En una organización madura, los administradores monitorean la calidad de los productos de software y los procesos que los producen. Existe un objetivo que establece las bases cualitativas con las cuales se analizarán la calidad del producto y los problemas con los productos y procesos. Planes y presupuestos están basados en una performance histórica y son

realistas; los resultados esperados para costos, tiempos, funcionalidad y calidad del producto son generalmente alcanzados. En general, una disciplina de proceso es continuamente seguida por todo el personal que entiende el valor de hacerlo, y la infraestructura necesaria para soportar los procesos existe. **12.- Prácticas comunes de las organizaciones de madurez alta**

Este documento corresponde a los resultados de una [encuesta](#) realizada por el Instituto de Ingeniería de Software de la Universidad Carnegie Mellon en marzo de 1999.

Algunas organizaciones con algo [grado](#) de madurez (como Onboard Shuttle and Boeing Space Transportation Systems) estuvieron haciendo mejoramiento de sus procesos mucho tiempo antes de que el CMM fuera publicado. Otros, como Motorola India, empezaron con un alto nivel de madurez como objetivo de negocio.

Si hay una iniciativa de TQM dentro de la organización, el programa de mejoramiento de los procesos de software de la organización con alto nivel de madurez está alineado y coordinado explícitamente con la iniciativa TQM. Alinear con la iniciativa TQM puede ser un desafío para las organizaciones con alto nivel de madurez.

Once de trece empresas encuestadas tenían certificación ISO 9001. Algunas comentaron que comenzaron sus esfuerzos en el mejoramiento de procesos usando ISO 9001, y cambiaron a CMM luego de obtener la certificación. Otros obtuvieron la certificación como un requerimiento de negocio en el mercado.


Los programas de mejoramiento de las organizaciones con alto nivel de madurez incluyen un énfasis en la automatización de procesos de software y en atender problemas del personal. Procesos, recolección de datos y análisis estadístico son automatizados si esto resulta práctico. Las organizaciones con alto nivel de madurez enfatizan una comunicación abierta, y un compromiso con la calidad y con el cliente en todos los niveles. Motivan un proceso de orientación en su staff, y tienen como costumbre diaria dar participación en el proceso de definición y mejora a sus empleados; el mejoramiento de los procesos es parte del trabajo de todos. La aproximación estratégica a la administración de calidad involucra la calidad en los objetivos de negocios y se focaliza en la satisfacción al cliente.

Hay una cultura de calidad en estas organizaciones. Incentivos y recompensas se establecen para los esfuerzos de mejoramiento de procesos y la participación del trabajador es más que un eslogan. La gente cree en el proceso, y cuando los errores ocurren, el foco es en mejorar el proceso, no en disciplinar a la gente.

El conocimiento y las habilidades son sistemáticamente cultivadas en la ingeniería de software, administración, habilidades interpersonales, y en el dominio de aplicación.

## 12.1- Interacción con el cliente

Es muy difícil ser un proveedor con un alto nivel de madurez si tiene clientes con bajo nivel de madurez (partners en el caso de alianzas estratégicas). Los datos forman una base muy fuerte para tomar decisiones. A niveles de madurez altos, hay una mejor base y entendimiento para explicar a los clientes porqué se hacen las cosas de una forma determinada. El cliente usualmente es razonable cuando se presentan datos, hechos y patrones frecuentes de éxito, entregados conforme los tiempos y presupuestos. La filosofía general de estas organizaciones es trabajar proactivamente con el cliente para formar un entendimiento mutuo de qué es lo que se hará, lo que significa educar al cliente sobre cómo hace las cosas la organización, con datos y resultados para apoyar sus métodos.

Los resultados de la [encuesta](#)  indican un énfasis general en administrar proactivamente la evolución de los requerimientos del cliente a través de ciclos de vida evolutivos e incrementales. Once de trece organizaciones usan ciclos de vida incrementales, y nueve de trece usan ciclos de vida evolutivos.

## 12.2- Administración de proyectos

Una práctica en las áreas clave de proceso relativas a la administración de proyectos en CMM que requiere una significativa interpretación es la estimación de tamaño. Las organizaciones de mantenimiento se basan generalmente en el nivel de esfuerzo, y una pequeña porción del valor del negocio se deriva de la estimación de tamaño. Sin embargo, el tamaño puede ser útil para medir mejoras, para el mantenimiento correctivo el tamaño suele ser "número de reportes de problemas" y la variabilidad suele ser alta. Las trece organizaciones de la encuesta usan líneas de código como medida de tamaño (dos usan puntos de función), pero los comentarios indican que en ciertas circunstancias la estimación de tamaño no es un valor para el mantenimiento, y el parámetro variable es la funcionalidad del paquete de actualización.

Las organizaciones con alto nivel de madurez sistemáticamente administran el riesgo. Se realizó un estudio detallado de las dos organizaciones que no lo hacían en forma "sistemática", y se encontró que se pueden caracterizar porque a) operan en un ambiente de alto riesgo; b) tienen un alto conocimiento de los riesgos asociados con su dominio de aplicación; y c) tienen tan profundamente internalizada la administración de riesgos que ya no reconocen cuán maduros están sus procesos de administración de riesgos. Como conclusión se puede decir que la administración sistemática de riesgos es una característica intrínseca de altos niveles de madurez, que está integrada en el proceso e internalizada por el staff.

## 12.3- Medición

Las organizaciones con alto nivel de madurez tienden a tener personas

dedicadas a la medición en los proyectos en los niveles de la organización, aunque pueden no estar asignados a tiempo completo. También motivan a los clientes a involucrarse activamente en especificar medidas y en establecer metas de calidad.

Las medidas y métricas en las organizaciones de alto nivel se puede caracterizar por:

- Conducidas por metas de negocio en el sentido del paradigma Goal-Question-Metric (GQM).
- Estandarización de las medidas comunes a través de toda la organización.
- Adaptadas a las necesidades específicas del usuario.
- Basadas en definiciones operacionales que definen cómo recolectar datos consistentes.
- Recolectadas lo más cerca posible del punto de origen.
- Involucran activamente a las partes afectadas, incluyendo al cliente.


Una de las barreras culturales que las organizaciones maduras deben sobrellevar es la expectativa de los administradores senior, que consideran que deberían estar observando y reaccionar a los datos en los cuadros de control. Los administradores e ingenieros tienen diferentes necesidades en cuanto a medición y análisis estadístico. Es cuestionable si los administradores obtienen valores significativos directamente del proceso de control estadístico. El valor administrativo es saber si los procesos de software son estables y están bajo control, que la organización puede alcanzar las metas dentro de su capacidad de proceso. Los ingenieros, por otro lado, son capaces de usar los datos de medición y las herramientas estadísticas para controlar los procesos, identificar problemas, y escalar problemas a la administración cuando sea apropiado.

La utilización de datos para evaluar la performance de los empleados es una preocupación actual en las organizaciones de alto nivel de madurez. Demming tenía una fuerte creencia en las técnicas estadísticas y estaba en contra de las evaluaciones de performance. Algunas organizaciones han adoptado la técnica de evaluación de performance "360", donde los subordinados evalúan a sus superiores como parte del sistema de evaluación de performance.

## **12.4- Aseguramiento del producto y del proceso**

El aseguramiento de la calidad del software (SQA – Software Quality Assurance) es tal vez el área de proceso clave más controversial en el CMM. Hay opiniones opuestas sobre si debe haber una organización SQA independiente o si la función de SQA debe ser integrada en el proceso

como parte de la cultura de calidad que se espera de una organización con alto nivel de madurez.

La [encuesta](#)  reveló que las trece organizaciones usaban un grupo independiente de SQA y incorporan la función de SQA al proceso. Por otro lado, algunas de ellas separan el aseguramiento del proceso y del producto. El grupo de SQA se puede focalizar en el monitoreo de procesos, mientras que el aseguramiento del producto se incorpora a las revisiones formales y al sistema de administración de la configuración.

## **12.5- Definición y deployment de procesos**

Uno de los cambios culturales más grandes que muchas empresas deben superar para alcanzar un alto nivel de madurez es transferir la propiedad de los procesos a quienes los utilizan y alcanzar un balance apropiado entre control y poder. El mejoramiento de los procesos es controlado por equipos que son "dueños" del mismo y que están formados por quienes los utilizan. Las actividades de mejoramiento de procesos son coordinadas por un grupo de ingeniería de procesos de software, pero no directamente realizadas por éste. Los errores se estudian por lo que se puede aprender de ellos, no para culpar a alguien.

Las organizaciones con alto nivel de madurez proveen descripciones de procesos mínimas pero útiles, que pueden ser usadas por profesionales experimentados o principiantes, y los procesos son altamente documentados.

Estándares, procedimientos y listas de comprobación pueden ser detallados, pero su uso es específico para determinadas circunstancias. Los clásicos principios de diseño, de ocultamiento de información y de abstracción se respetan.

## **12.6- Capacitación y tutoría**

El entrenamiento en las organizaciones con alto nivel de madurez puede llegar al extremo, algunas organizaciones tienen entrenamientos obligatorios que pueden durar hasta ocho semanas, más requerimientos de educación continua. Este entrenamiento incluye el desarrollo de materiales de entrenamiento internos y externos, programas de interés y talleres. Otras organizaciones confían firmemente en programas de tutoría formal para impartir conocimientos y habilidades al personal. Estas organizaciones por lo general tienen ambos programas: entrenamiento y tutoría.

Las organizaciones de alto nivel de madurez reconocen la importancia crítica de la experticia en el dominio del problema. Los grupos usualmente tienen requerimientos de experticia en el dominio de la aplicación, así como técnicas de ingeniería de software y tecnologías.

Las características comunes de los programas de tutoría formal son:

- Los tutores poseen amplios conocimientos y son respetados.

- Los tutores son entrenados en cómo desenvolverse efectivamente como tutores.
- Las expectativas para el tutor y el alumno son especificadas explícitamente.
- La relación de tutoría dura por extensos períodos de tiempo, típicamente un año.
- El tutor y el alumno están físicamente cerca uno del otro, tal vez compartiendo una oficina.
- La tutoría es controlada por la administración.
- Las habilidades de tutoría son parte de los criterios de evaluación de performance para el tutor.

## **12.7- Administración cuantitativa de procesos vs pensamiento estadístico**

Conceptualmente los niveles de madurez 4 y 5 se basan en el control estadístico de procesos. El nivel 4 focaliza en control identificando y eliminando las causas de la variación en el proceso, los eventos extraordinarios que no permiten al proceso realizar la tarea requerida. El nivel 5 focaliza en mejoras, abordando las causas comunes de la variación que son intrínsecas al proceso. Más generalmente las organizaciones con alto nivel de madurez aprecian los fundamentos del pensamiento estadístico: todo el trabajo es una serie de procesos interconectados, todos los procesos son variables, las decisiones se deben basar en hechos, y una reducción en la variación provee oportunidades de mejora.

Las áreas clave de proceso de nivel 4, sin embargo, hablan acerca de "Administración cuantitativa" más que de "Control estadístico". El CMM distingue entre desvíos (performance deseada o esperada) en el nivel 3 y límites de control (lo que el proceso puede hacer) en el nivel 4, pero la terminología utilizada para las prácticas de nivel 4 es "Límites aceptables".

La mayoría de la organizaciones de nivel 4 y 5 alcanzaron estos niveles usando una interpretación poco precisa de lo que significa "Administración cuantitativa". Solamente unas pocas de ellas realizan consistentemente análisis estadísticos rigurosos, aun hoy. En la mayoría de los casos el análisis parece estar centrado en la detección y predicción de defectos.

**12.8- Calidad de software** Los datos recolectados en el nivel 4 pueden proveer una visión profunda de la calidad del software, y la implementación de la "Administración cuantitativa de procesos" y la "Administración de la calidad del software" se encuentran fuertemente unidos.

## **12.9- Captura del conocimiento del dominio de aplicación y del producto**

Las organizaciones con alto nivel de madurez sistemáticamente capturan el conocimiento del producto y del proceso, lo cual les permite

alcanzar mayores niveles de madurez. Enfatizan en líneas de producto, incluyendo una fuerte reutilización de arquitecturas y componente de software.

Siete de las trece organizaciones encuestadas aplican reutilización "sistemática" y ocho tienen líneas de producto. Dos organizaciones indicaron que utilizaban líneas de producto a nivel piloto, pero no reutilización sistemática, y una organización usa pilotos de reutilización sistemática pero no líneas de productos.

**13.- CMM e ISO 9000** A continuación se enumeran los puntos sobresalientes del Modelo CMM, versión 1.1 e ISO 9001, e ISO 9000-3, los estándares ISO 9000 que se aplican al desarrollo y mantenimiento de software.

### **13.1- CMM**

El Modelo de Capacidad Madurativa describe los principios y prácticas subyacentes a la madurez de los procesos de software e intenta ayudar a las organizaciones a mejorar la madurez de sus procesos de software a través de un camino evolutivo desde ad hoc, caótico hasta maduro, disciplinado. También podría ser usado por los clientes de una organización para identificar las fortalezas, debilidades y riesgos asociados a sus proveedores de software.

Por conveniencia las prácticas clave en cada área clave de proceso están organizadas según características comunes:

- Compromiso a la realización: ¿Qué acciones debe tomar la organización para asegurar que el proceso está establecido y es robusto? Esto incluye prácticas concernientes a las políticas y el liderazgo.
- Habilidad para realizar: ¿Qué precondiciones deben existir en el proyecto o la organización para implementar el proceso de software en forma competente? Incluye prácticas que conciernen a los recursos, la capacitación, la orientación, la estructura organizacional y las herramientas.
- Actividades realizadas: ¿Qué roles y procedimientos son necesarios para implementar un área clave de proceso? Incluye prácticas sobre planes, procedimientos, el trabajo realizado, seguimiento y acciones correctivas.
- Medición y análisis: ¿Qué procedimientos son necesarios para medir el proceso y analizar las medidas? Incluye prácticas de medición de procesos y análisis.
- Verificación de la implementación: ¿Qué pasos son necesarios para asegurar que las actividades son realizadas cumpliendo con el proceso establecido? Incluye prácticas administrativas sobre las revisiones y auditorías.




Satisfacer un área clave de proceso depende tanto de la implementación como de la institucionalización del proceso. La implementación se describe en la característica "Actividades realizadas"; la institucionalización en las demás características.

### **13.2- ISO 9001, 9000-3**

Los estándares ISO 9000 especifican requerimientos de sistemas de calidad para ser usados cuando un contrato entre dos partes requiere la demostración de que uno de los proveedores es capaz de diseñar o proveer un producto. Las dos partes pueden ser un cliente externo y un proveedor o ambas pueden ser internas como los grupos de marketing e ingeniería de una misma compañía.

De la serie ISO 9000, ISO 9001 es el estándar más pertinente al desarrollo y mantenimiento de software. Las organizaciones lo usan cuando deben asegurarse de que un proveedor cumple requerimientos específicos de diversas etapas del desarrollo, incluyendo el diseño, desarrollo, producción, instalación y servicio. ISO 9000-3 provee los lineamientos para aplicar ISO 9001 al desarrollo, soporte y mantenimiento del software.

Las organizaciones típicamente usan los estándares ISO 9000 para regular sus sistemas de calidad internos y asegurar los sistemas de calidad de sus proveedores. De hecho, los estándares se usan frecuentemente para registrar sistemas de calidad de terceros. La certificación del [registro](#)  tiene visibilidad dentro de la organización y es llevada a cabo por entidades registrantes. Los auditores son entrenados en los estándares ISO 9000, pero no son entrenados en cuestiones específicas del software. Si el alcance de una auditoría específica software, deben incluirse en el equipo con conocimiento de software.

