Framework OO
Análisis & Diseño de Software/Fundamentos de Ingeniería de Software

Pablo Cruz – Gastón Márquez-Hernán Astudillo
Departamento de Informática
Universidad Técnica Federico Santa María

¿Qué es un Framework? [1]

- Es una colección de clases abstractas y sus algoritmos asociados que se constituyen para una aplicación en particular
- *Un Framework consiste en clases abstractas, las operaciones que implementa y las expectativas puestas en las subclases concretas* [Deutsch, 1983]
- *Un framework es un diseño abstracto para un tipo en particular de aplicación, y usualmente consiste en un numero de clases* [Johnson et al., 1988]

¿Qué es un Framework? [2]

- Puntos en común:
  - Un Framework guía un dominio/familia de productos
  - Prescribe cómo se descompone un problema
  - Diseño de una aplicación o subsiste
  - Conjunto de clases y cómo colaboran
  - Se usan Frameworks para construir aplicaciones mediante
    - La creación de nuevas subclases
    - Configuraciones de objetos

¿Qué es un Dominio?

- Un dominio es un área del conocimiento que
  - Limita la máxima satisfacción de los requisitos de los stakeholders
  - Incluye un conjunto de conceptos y terminologías que se entienden en la práctica en un área en particular
  - Incluye el conocimiento de cómo construir sistemas de software en un área

Propiedades básicas de un Framework

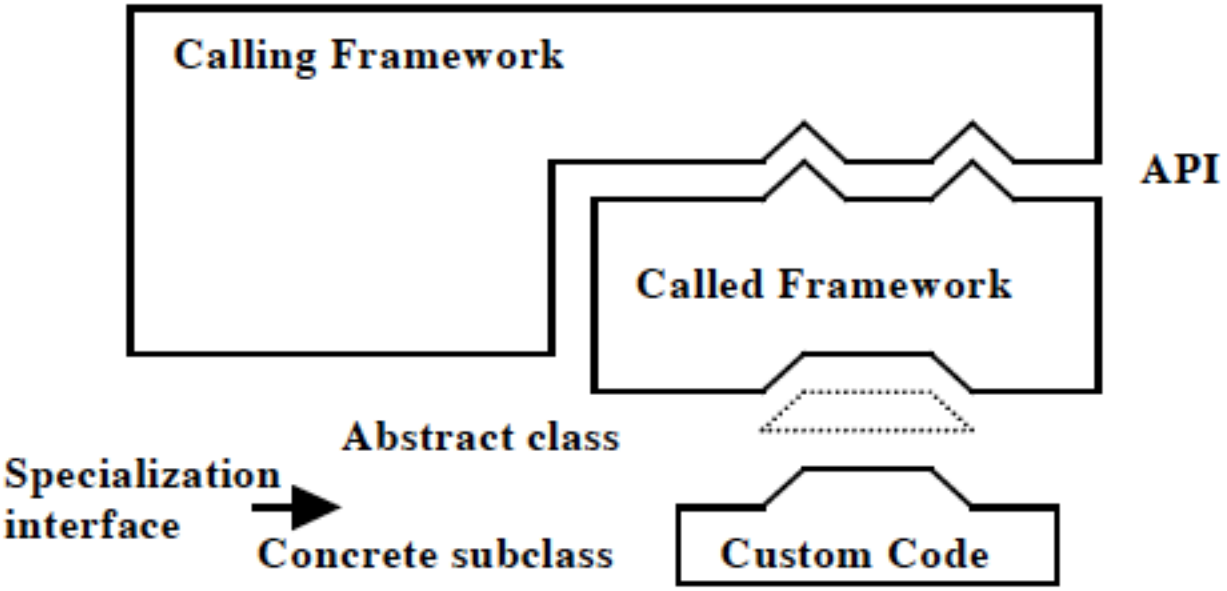
- **Modularidad**
  - Clases abstractas con interfaces estables, encapsulando y localizando cambios en *hotspot*
- **Reusable**
  - En análisis, diseño y código
- **Extensible**
  - Debe tener la capacidad de poder ser escalable

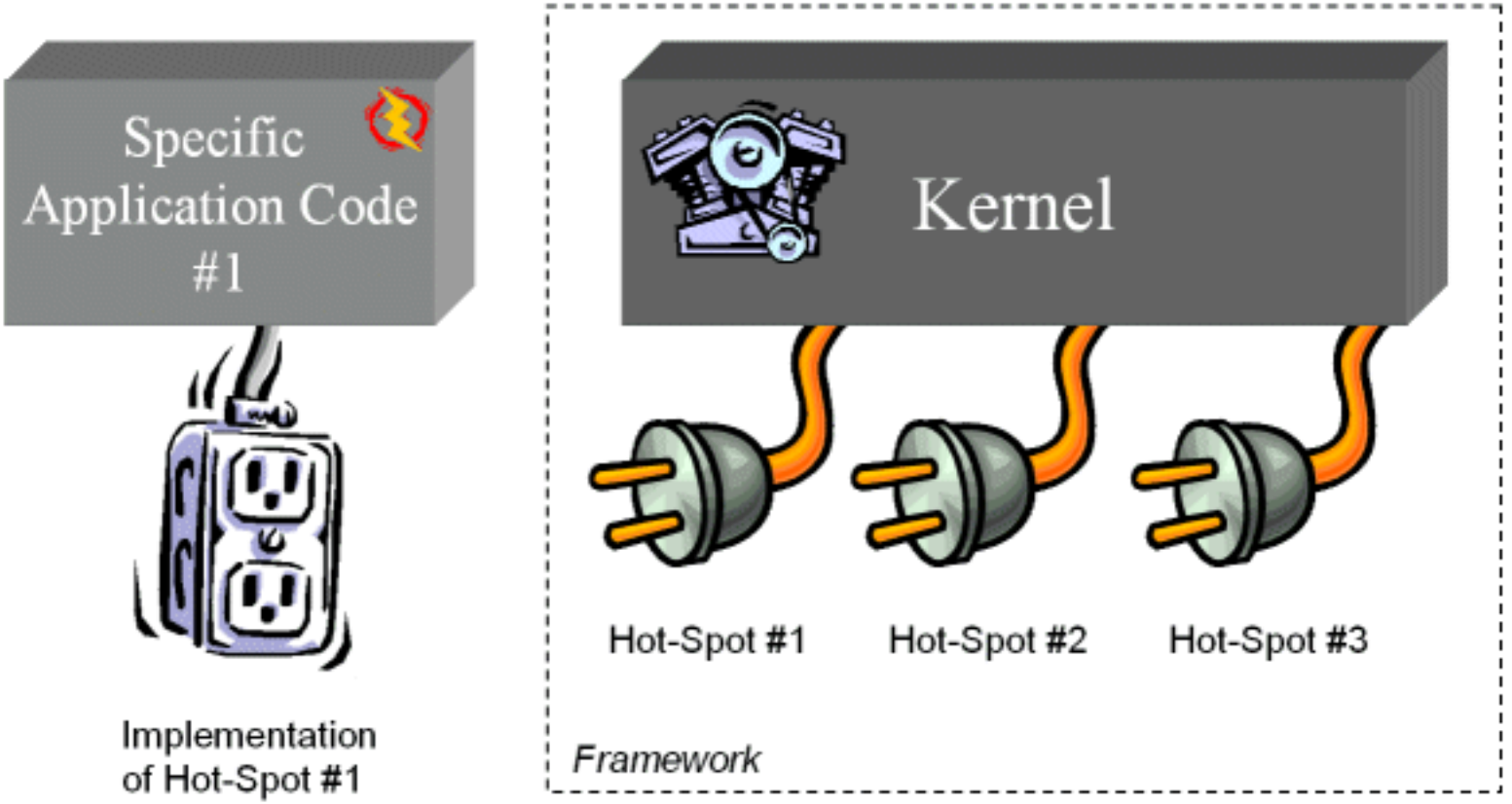
Frameworks versus Librerías

- Librerías
  - El desarrollador de la aplicación escribe un programa principal, determina el flujo de control y la descomposición del problema. El código llama código de librerías
- Framework
  - El desarrollador de la aplicación escribe una subclase. El Framework ya tiene determinado el flujo de control y la descomposición del problema. El Framework llama al código particular.

Objetivos de los Framework

- Hacer fácil el desarrollo de aplicaciones
- Escribir lo menos posible de código nuevo
- Permitir a programadores novatos a escribir buenos programas
- Potenciar la experiencia del dominio de programadores expertos


Calling Framework
API
Called Framework
Abstract class
Specialization interface
Concrete subclass
Custom Code


Specific Application Code #1
Kernel
Hot-Spot #1
Hot-Spot #2
Hot-Spot #3
Implementation of Hot-Spot #1
Framework

Arquitectura de Software versus Framework

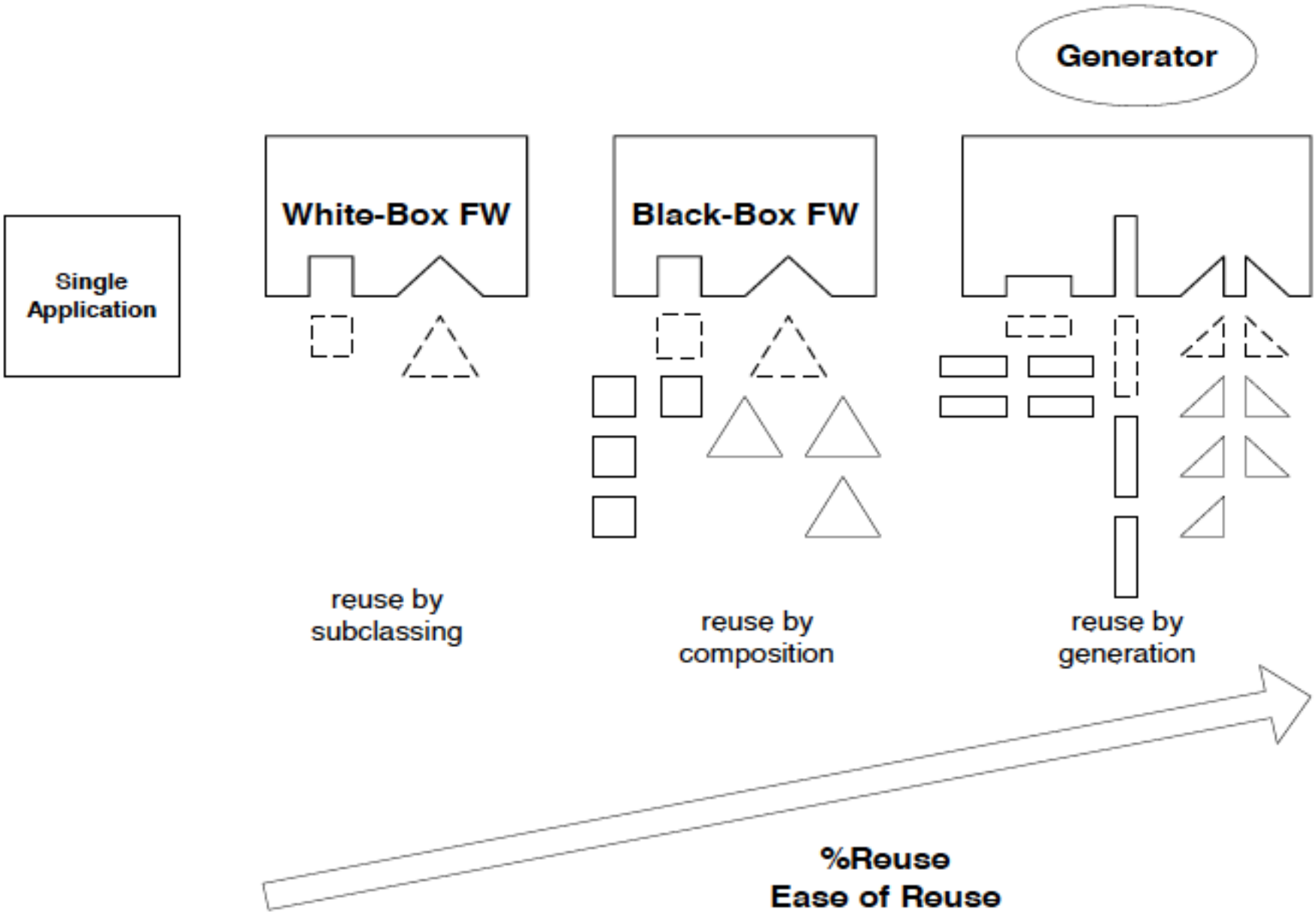
- Arquitecturas
  - Son decisiones
  - Puede ser para una aplicación en particular
  - Son creadas con muchas cualidades en mente
- Framework
  - Son implementaciones concretas de arquitecturas semi-completas
  - Son diseñados para ser
    - Reusables
    - Especializados

Patrones de diseño versus Framework

- Patrones de Diseño
  - Describen micro-arquitecturas
  - Son abstractos
- Framework
  - Tienen una arquitectura concreta
  - El diseño del Framework puede incorporar patrones de diseño a un nivel de micro-arquitectura
  - La flexibilidad de la especialización de un Framework a menudo se proporciona un patrón de diseño

Los diferentes usos de un Framework

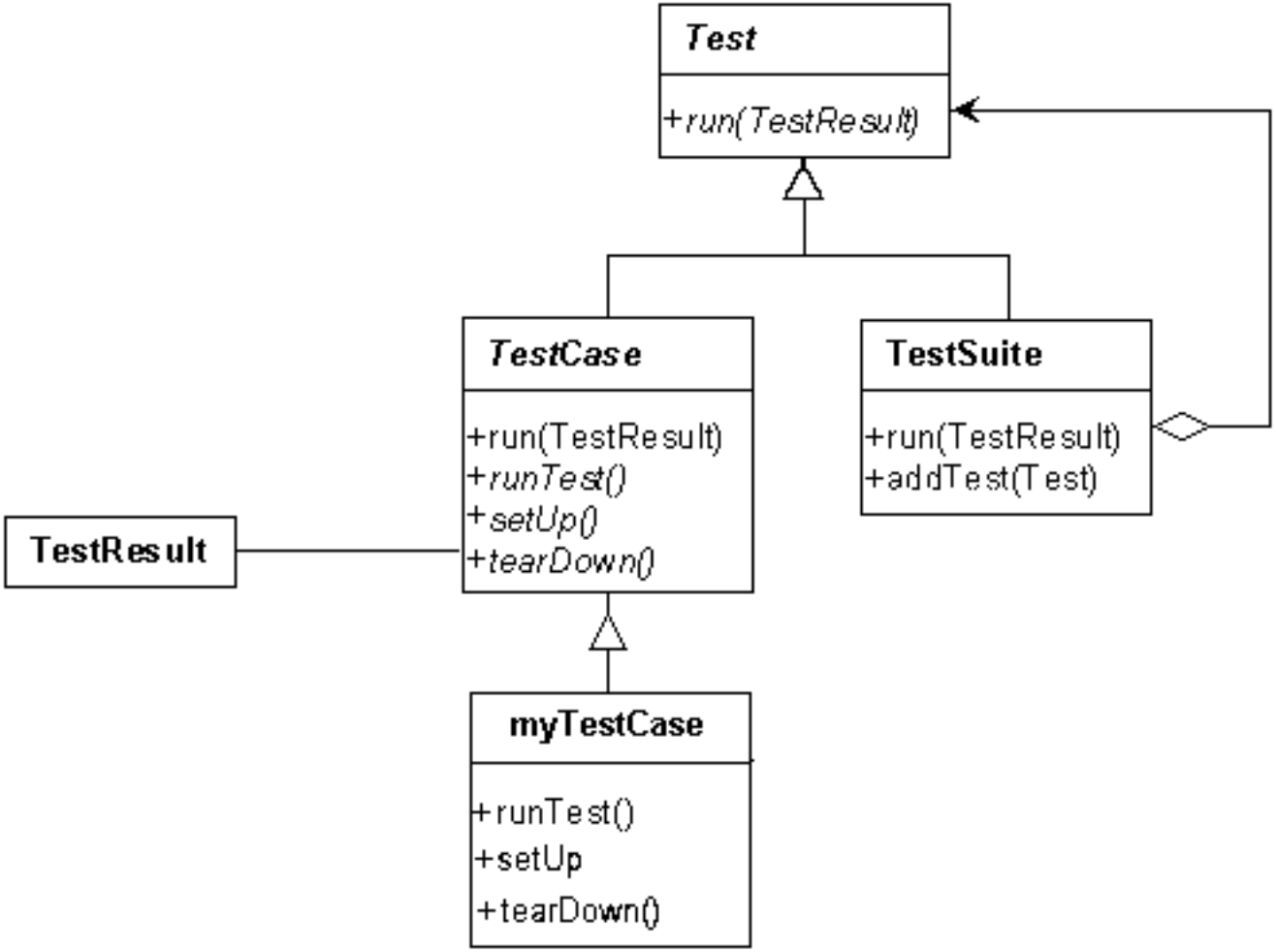
- Desarrollo de un Framework
  - Creación de una infraestructura para un dominio en particular
  - Mediante el dominio de expertos
- Ajustar un Framework para desarrollar una aplicación
  - Realizado por programadores novatos
- Evolución
  - Relacionada con la infraestructura


Generator
Single Application
White-Box FW
Black-Box FW
reuse by subclassing
reuse by composition
reuse by generation
%Reuse
Ease of Reuse

Ejemplo: The JUnit Testing Framework [1]

- JUnit es un Framework para Java con el objetivo de realizar testing.
- La arquitectura de JUnit se describirá a continuación bajo la notación UML

Ejemplo: The JUnit Testing Framework [2]



Ejemplo: The JUnit Testing Framework [3]

- Al tener la estructura del Framework, el desarrollador lo debe usar como un *libro de cocina*
  - Por ejemplo, este Framework puede ser instanciado mediante la implementación de clases TestCase
  - El método run() heredado de la clase Test ejecutará casos de pruebas a través de setUp(), runTest() y tearDown() en secuencia

Ejemplo: The JUnit Testing Framework [4]

- El método runTest() lleva a cabo las llamadas a las funciones de la API.
- El Framework JUnit también provee una infraestructura para ejecutar dos o más pruebas en similares o idénticos conjuntos de objetos
- Esto se conoce como *test fixture* y es posible mediante los métodos setUp() y tearDown()
- La aplicación que necesite usar JUnit debe instanciar el Framework mediante la clase myTestCase la cual consecuentemente instancia los métodos mencionados anteriormente

Desarrollo de Framework [1]

- Se pueden identificar tres etapas
  - Análisis del dominio
  - Diseño del Framework
  - Instancia del Framework

Desarrollo de Framework [2]

- Análisis del dominio
  - Este análisis intenta descubrir el dominio de los requisitos y posibles requisitos futuros
  - Para lo anterior, es útil saber si en la organización existe documentación antigua, software similar, entre otros

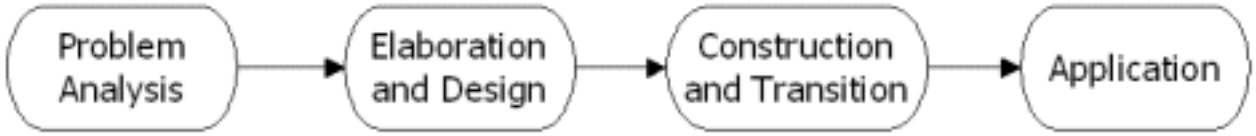
Desarrollo de Framework [3]

- Diseño del Framework
  - Se definen las abstracciones del Framework
  - Los elementos más importantes se comienzan a modelar (se recomienda usar notación UML)
  - La extensiblidad y flexibilidad definida en el análisis comienza a ser considerada
  - Los patrones de diseño son utilizados en esta etapa

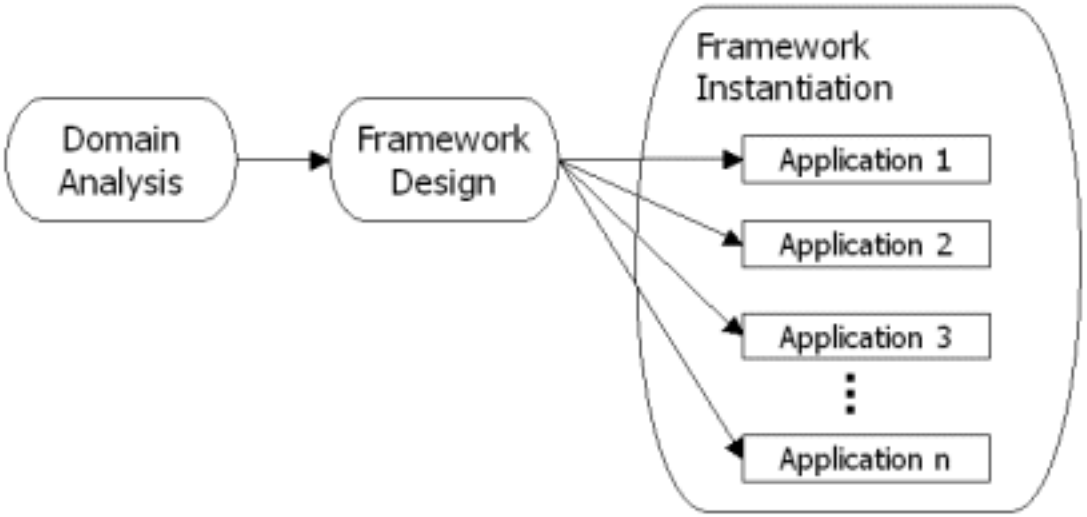
Desarrollo de Framework [4]

- Instancia del Framework
  - Aquí el Framework es capaz de generar sistemas de software
  - Cada instancia de sistema de software tendrá características en común que son parte del Framework

Desarrollo de Framework [5]



Traditional Object-Oriented Design



Framework Development Process

Algunos ejemplos de Framework

Grails [1]

- Framework de aplicaciones open source
- Utiliza como lenguaje de programación Apache Groovy (basado en Java)
- Antes era conocido como Groovy on Rails

Grails [2]

- Provee un Framework web para la plataforma Java
- Reusa tecnologías Java existentes como por ejemplo Hibernate y Spring bajo una interfaz simple
- Ofrece un Framework consistente para programación
- Es posible documentar, como por ejemplo, templates GSP (Groovy Server Pages)
- Posee aplicaciones de ejemplo en donde se especifica el uso de Grails
- Fomenta la alta productividad
- Es un Framework MVC

CodeIgniter [1]

- Web Framework
- Open source
- Creación dinámica de páginas web en PHP

CodeIgniter [2]

- Está basado en el patrón MVC
  - Las clases que están en el Controlador son obligatorias, pero las del Modelo y Vista son opcionales
- También se puede expandir a Hierarchical MVC (HMVC)
  - Permite que los desarrolladores puedan mantener una agrupación modular del Controlador, el Modelo y la Vista pueden mostrarse en formato como sub-directorio
- La industria lo considera un framework rápido en desarrollo

Yii [1]

- Framework web
- Basado en orientación a objetos
- Compuesto por MVC
- PHP

Yii [2]

- Puede generar servicios WSDL (Web Service Description Language) complejos
- Gestiona de manera eficiente Error Handling y logging.
- Se consideran prevenciones de seguridad, como por ejemplo Cross-site Scripting (XSS), Cross-site request forgery (CSRF) y cookie tampering
- Probes Unit incluidas (PHPUnit)
- Aplicaciones CRUD incluidas

Django

- Framework web
- Open source
- Escrito en Python
- Promueve el model MVT (Model-View-Template)
- Permite la extensibilidad de las aplicaciones

ASP.NET

- Framework web de Microsoft
- Se pueden crear páginas web dinámicas y servicios web
- Conglomerado de tecnologías Microsoft para proveer soluciones
- Provee escalabilidad estable
- Se pueden crear páginas en HTML, CSS y JavaScript

FIN