Atitit 开发效率补充哦哦那个、、

目录

[1. 架构方法上选择快速开发的架构 1](#_Toc31038)

[2. 编程方法上选择快速的编程范式和编程方法 1](#_Toc8100)

[3. 开发方法论上需要快速的方法 2](#_Toc193)

# 架构方法上选择快速开发的架构

Atitit arch design context软件架构设计的内容

目录

1. 考虑到架构设计原则 1

1.1. 开发效率 稳定性 可靠些等 性能 1

1.2. 简单原则则 1

1.3. 配置化 vs 开发 2

2. 云平台vs自建 2

2.1. Rds mysql 2

3. 开发语言选项 4gl优先 2

4. 免编译免部署 配置化 2

4.1. Db oritd模式优先 2

4.2. Table oritd vs java oritd 2

5. 架构模式 单体 vs 分布式 2

5.1. 模块化 井字模式 2

5.2. 通用化模块 vs 业务相关模块 2

6. 分层层次数 双层 》 三层 》多层 3

7. 类库选择 简单优先 3

8. 数据库选择sql erver vs mysql 3

9. Ati的架构演化路线 3

9.1. 淘宝的架构演化路劲 3

# 编程方法上选择快速的编程范式和编程方法

Atitit 编程范式之道 attilax著 v2 u66.docx

Atitit 编程范式之道 attilax著 著

1. 编程范式与编程语言的关系是什么？ 2

2. 高效率的编程范式 2

2.1. DP(Declarative Programming)描述性范式 2

2.1.1. 俩种实现模式 LP逻辑编程 FP 函数式编程 2

2.2. LOP 面向语言编程（LOP, Language Oriented Programming） 2

2.3. AOP 3

2.4. 泛型式、元编程、切面式和事件驱动式。 3

2.5. 1.2.5. MP(Meta Programming) 6 2. Table-oriented Programming 7 3

3. OOP 3

4. OOP 3

5. Fp 函数式编程 3

6. 命令式 Imperative  3

7. 其他 3

7.1. 泛型式、 3

7.2. 产生式编程（Generative Programming） 自动生成源代码的编程也属于另一种编程范式 3

7.3. 过程化（命令式）编程  4

7.4. 事件驱动编程  ]响应式编程范式-- 4

7.5. 4个常用的编程范式：泛型式、元编程、切面式和事件驱动式。 4

8. 元编程 原来元编程就是编写能自动生成源代码的程序。" 4

8.1. 通过向导、拖放控件等方式自动生成源码 4

8.2. 产生式编程（Generative Programming 区别 4

8.3. 许多脚本语言都提供eval函数，可以在运行时将字符串作为表达式来运算[4]。 5

9. Other 5

9.1. Atitit 编程范式总结 5

10. 参考资料 6

10.1. 编程范式，程序员的编程世界观 -- 简明现代魔法.html 6

10.2. 3.2 超级范式--提升语言的级别（2） - 51CTO.COM.mhtml 6

10.3. 编程范式思考问题 - huangshanchun的专栏 - CSDN博客.mhtml 6

10.4. Atitit 编程范式总结 v2 taf.docx 6

# 开发方法论上需要快速的方法

Atitit各种驱动的xdd tdd bdd设计 ATDD ddd v3 u66.docx

Atitit各种驱动的xdd tdd bdd设计 ATDD ddd v2 s66

开发方法论与效率提升

Atitit各种驱动的xdd tdd bdd设计 ATDD ddd

Tdd bdd 行为驱动

目录

1. 2. 软件开发过程中最常见的两个问题 1

1.1. 需求和开发脱节： 1

1.2. 开发和测试脱节： 2

2. 高效率的开发范式 开发方法论 2

2.1. Xp 2

2.2. 完美的组合是TDD，DDD和BDD 2

2.3.  FDD (Feature-driven Development)  2

2.4. BDD 指的是Behavior Drive Development 2

3. Xdd "X" Driven-Development Methodologies 2

3.1.  DDD (Defect-Driven Development) – 3

3.2. RDD (Responsibility-Driven Design) – 3

3.3. UGDD (User Guide-Driven Development) 3

3.4.  MDD (Model-driven Development)  3

3.5. DDD (Documentation-Driven Development) 3

3.6. Edd 事件驱动 3

3.7. TFD -- Test First Development) 3

3.8. 2. BDD 指的是Behavior Drive Development，也就是行为驱动开发 3

3.9.  DDD DDD指的是Domain Drive Design，也就是领域驱动开发 3

3.10. CBD（核心Core+行为Behavior+驱动Driver）架构模式 4

3.11. TDD（测试驱动开发(Test-Driven Development)） 4

3.12. ATDD：验收测试驱动开发（Acceptance Test Driven Development） 4

4. 区别 5

4.1. 简而言之，完美的组合是TDD，DDD和BDD 5

5. ref 7