Chapter1

2. 架构的共性 (1) 一组基本的构成要素— 之后形成的拓扑结构—— 质量——性能

软件架构的构建和连接件

软件架构的定义

SA={elements, 类: 处理元素。 B. 连接元素把

程序规模和软件架构的关系

· 善软件系统规模越来越大 越来越复杂用尸需来(订旅性) 越来越复杂。变化越来级快度量 (详为版性) 的现实也越来越来 如将将成百上千个功能组合起来。同时需求,空阀越来越周围,此时 每个核软的后即结构现设计显彻域来越重要。 得要体现在旋伸中而不是功能模块内部的实现中,量效对于大规模的复杂软件经全局结构的设计比较对量等运输。这种原始的原则,但可以

本人下不作的以至是不成人。 8. 是软件开发过程和期的产品。在开发的早期阶段就考虑系统的正确设计与方案选择 5.开发,测试,维护各个阶段提供了保证。(2)与其他后期的设计活动相比。SA设计的成。 医对象分别,但有效的 SA设计会给软件开发带来极大的使剂。(4)在大型软件系统 直侧性更多的是由系统结构和加强。例外来实现的,而不再仅仅依靠的选择的算法。运成就

软件架构在软件生命周期中的作用 项目规划

\$\$ 优集(1)在内存里只有一个实例,减少了内存的开销,尤其是频 毁实例。(2)避免对资源的多重占用(比如写文件操作)。\$\$ 缺点 能继承,与单一职责原则冲突,一个类应该只关心内部逻辑,而 性架构和其他软件设计活动的比较

是架构师的关注点以及软件质量定义

廣量 软件与明确的和隐含的需求相一致的程度 质量和质量的关系

提高一个质量、经常会影响(提高或降低)其它质量 软件质量因素及对策

十四里以系及刈束 软件的"时间**\$**20°间"效率:集群,缓弃,异步(2) **安全性**在双 技规用户的使用企图:防火塘,入侵检测,加密.单一入口。 度.用户容易使用和学习:续一风格界面,定制(4) **宣用性**,消 在缓步升代,物件,框架,软件后总线(6) **恒性,可附性** 能力(各据,恢复)集群,灾备、故障状移(6) **可修改性**,软件 使.利:在河底水份等上来可以外外,从

^{此再定获得} 软件架构技术发展与澝化

统 = 算法 + 数据组 1980's) (4) 系统 =

.)分布式应用借助中间件在不同技术之间。 t之上 管理计算机资源和网络通讯。(3) 中 使它们具有不同的接口。(4) 通过中间件 2 一个通道、保障应用信息的可靠传递。(5) 5) 中间件是一个框架、促成企业应用的完整

哪些软件是中间件

舌硬件 (CPU ネロォロ マ / : (不同的存储和访问 他们是不兼容的。(/ 格式)等。长 **拿跟互操作**

会的计算环境和支撑平台、包括对象请求代 用服务基中间件等。(2. **应用表现线中间件** 务集成和数据集成功能。包括常见的消息中间 的高。(3. **生务契构线中间件** 除了可以特型 性抽象至中间件,形成应用模式,如业务流程 消息、分布式效解、(5. 两个

能(英、刘家、泰山、模式寺)。主要选择包括 Class Clagfarffs。 和 sequence diagrams等。(3) **开发调整** 推送系统中的子系统。 4. 主要图形包括 component diagrams,package diagrams等。(生之间的通信的作关系,主要图形包括 activity diagrams, sequen 描述系统如何被安装,部署与配置在分布式的物理环境下,主i

+1 视图模型的地位

Chapter2-1

原則 釜口隔高原則

\$\$ 相同点 (1) 简单依赖和聚合都有接收 A 类型的方法。(2) 聚合和组合都有 A 类型字段 \$\$ 7 同点 (1) 简单依赖没有 A 类型字段。接收 A 类型仅在方法中使用。(2) 组合没有接收 A 类型

Chapter 2-2 创建型模式的要点
 不允许显示创建

别人无法创建自己可以创建。(2) 别人可以获取. 单例实现的方式

blic class SingletonEager{private static .nstance=new SingletonEager();private SingletonEager getInstance(){return

工厂模式的优缺点

抽象工厂模式的定义

及工厂模式是围绕一个超级工厂 企工厂,提供了一种创建对象的最佳 一个相关对象的工厂,不需要显式 模式提供对象。 抽象工厂模式的特点

35 使用场景

模式意图

_{接□} 适配器模式的实现

的问题。 23. **桥接模式的定义 新装载**(Bridge Pattem)) 者的解釋。桥接模式涉及 出实现类。这两种类型的 4. 桥接模式的特点

(1.模式意图 题 在有多种 (3.使用时机 整 把这种多 代码 抽象类

工厂模式的使用场景及注意事项

配器模式(Adapter Pattem)是作为两个不兼容的接口之 设计模式属于结构型模式,它结合了两个独立接口的功能 一个单一的类,该类负责加入独立的或不兼容的接口功能

\$使用场景有动机地修改一个正常运行的系统的接口。这时应该考虑使用适配 模式 \$\$ 注意事项 适配器不是在详细设计时添加的。而是解决正在服役的项

ut.printin(unreteringlementor implement bstraction(protected Implementor implement (Implementor implementor){this.implementor abstract void operation();}class RefinedAb traction(public RefinedAbstraction(Impleme er(implementor);}public void operation(){

桥接模式的使用场景和注意事项

桥接模式和话配器结合使用 桥接模式的实现化角色的接口与现

单例模式的注意点

他汉式,线程不安全(最基础的实现方式):(1)Lazy 初始化:(2)无多线程安全 多线程不能正常工作(3)没有加锁synchronized,严格意义上它并不算单例模

使多效性不能此而上[トメインルン ア スアルルル。 アササ、(4) public class SingletonLazyUnsafe(private static SingletonLazyU safe instance;private SingletonLazyUnsafe(){}public static Sin letonLazvUnsafe getInstance(){if(instance==null){instance=new

e;}} 董记文 静态内容集(1)Lazy 初始化、利用了 classicader 机制床保证初始化 tance Byl 另一个线程。而 Singleton 美被装载 了, instance 不一定被刺蛇化 Singleton Holder 美没有被主动使用,只有通过显式调用 gethrstance 方法时, 会显式衰弱 Singleton Holder 美,从而实例化 instance。 实现 Singleton 英远

一个调用者拒包建一个对象,只要知道其名称或引动。1。16万。 思增加一个产品,只要扩展一个工厂共同可;(3)屏蔽产品的具体 多次产品的接口等被赢到10页次增加一个产品,需要要加加一 实现工厂,使得系统中英的个数成倍增加;(2)在一定程度上增加 。同时也增加了系统具体类的依赖

日于某些原因需要给某对象提供一个代 1800年的

^{甲介。} 代理模式的特点

代理模式/委托模式的定义

模式意图

法報多族領、我们可以在访问此次 机、型在河内一次和 可能行理的内容 和在中春在 Subject(wold reque-wbject(public void reque-wbject(public void reque-wbject(public void request) subject realsubject; public realsubject-ness Realsubject; printin(*Proxy: Performing ass Proxy)和trennemo(public ject proxy-mer Proxy(); proxy 22. 静态代理的优数点 36 他们可以被解析不确定用。 /{System.out.println("Re Proxy implements Subject void request(){System.o *processing.");if(realSu ();}realSubject.request(post**\$\$**processing."\ static

曼: (2)增加了系統的复杂度. 7. 代理模式按职责划分的使用场景

远程代理. (2)虚拟代理. (3)Copy \$5 on \$5 Write t 防火場代理. (7)同步化代理. (8)智能引用代理

9. 中介者模式的特点

模式意图

将阿尔语·阿尔 美中单独处理 中介者模式的实现

eMeus.ague colleagu-ague (colleague) meus. ague (colleague) meus. guel sendwessage ('Hello from A. from 81');} 1. 中介者模式和单例模式结合 中介者模式和单例模式结合

定义中介者接口,把具体中介者对象实现成为单例。同事对象不持 不是太無辜的时候再换程即由介含对象并调用

者,而是在需要的时候直接获取中介者对象并调用。 42。中介者模式的优缺点 \$\$ **&点**(1)降低了类的复杂度、将一对多转化成了一对一。(2) (3)附合迪米特原则**\$\$ 集**(中介者会庞大、变得复杂难以维多

观察者

5. 观察者模式的特点 (模式意图

(1) 都将得到通知。 (2) 都将得到通知。 (5) **观察者模式**

个用于保存观察者对象的聚集类科 可观察者的抽象方法:(2)具体目标 And State Sta

t@. erver observer);voi ifvObservers();}cla

观察者模式的注意事项

: tee{**public** void spe ecificRequest from

明用;(4)**具4**

者模式的实现

時使用拓景(1)对象结构中对象对应的类很少改变,但经常对象进行很多不同的并且不相关的操作。而需要避免让这类转往需專項:访问者可以对功能进行统一,可以做报表。

计算复杂性的表

統垂直架构改造的核心就是要对应用进行服务化,服务化改造使用到的核心技术就是分 从集中到分布式依旧存在的问题

业务进行梳理,根据业务的? **业务横向拆分方法**

开销院槽等较重条块财。需要在线调大服务调用 全:身份验证与权限验证 功能度量指标/性能指标包括什么

)定义适应不断增长的处理 1. 高可用性的定义

聚集酐来提升系統整体的处理 . **高性能集群的复杂性** 性能集群的复杂性主要体现在 . 常见的负载均衡器

硬件负载均衡的定义

进行配置,但无法进行扩展和 **软件负载均衡的定义**

等數位住先类算法則是於在客户編的角度來進行分配的。 中時应客戶編的目的,负载最低优先类算法是站在服务器的角度來進行分配的 性能最优算法的特点及好处坏处

能最优优先类算法本质上也是感知了服务器的状 主要体现在:负载均衡系统需要收集和分析每~ |本身也会消耗较多的性能:(3)工业上使用采样

成功的分布式系统依赖于什么

① 画向消息的中间件服务的重要类别。② 持久异步通信。③ 为消息提息传输期间保持后跃状态。⑤ 支持允许消息传输的时间开销。

(2)速度快,(3)可以补偿发送方和接收方之间的速度偏差(但不补偿速度差异)。(4)发送方接收; 步更复杂。(2)发送方不知道是否已经送达。(3)存在缓冲区溢出风险。

作为发送和接收之间的命名消息目的地。②允许进程独立执行和失败。

Chapter3-1

传统垂直结构改造的核心

部署效率低,应用数量膨胀,数据库连接数共 应敏捷持续交付的挑战。

业务纵向拆分方法

. <mark>为什么要设计高性能集群/负载均衡</mark> 服务牆有性能天花板,无论如何优化,无论采用多好的硬件,总会有一个上限 能集群来提升系统整体的处理性能.

负载均衡算法分类

据算法期望达到的目的,大体上可以分 **任务股平分类** 负数均衡系统将收到的 比例或者权量,的平均。(3) **负数均衡** 等来衡量系统的压力。(3) **性能最优美** 快的服务簿。(4) **Hash 委** 负载均局系统 4, **轮间的定义**

tash 类算法的定义

进程通信的挑战

》异步交互 (2)可拿服务交付 (3)通过中间消息服务器处理消息 (4)支持数据库集成。

Docker 目前还在版本的快速更新中 市干高版本内核 存在版本養容问题 を 复割定义及做法 (1)—N= (Message Passing)
Chapter3-5 上读取不到。 **主主复制的优缺点** 摆缓存的两个术语:命中列和最大元素/最大空间 _{数据冲突} 数据集中集群的定义 据集中集群称为一主多备/从。数 主从的架构进行灵活变化。 1. 数据集中集群的特点 (1)多备即多通道,增加了主机的第 "践中,需要考虑降低该压力);(2)多 "要在备机之间进行数据一致性检查? 安层出现不同的到新和决管(4)每 位力、無 一致、需要在各別・近不同的ラティー 実不一致、各易出現不同的ラティー 注、需要算法 ? 数据分散集群的定义 "****を个服务器组成一 服务治理中心 新的服务器后,算法能够自动将数据分区服务器的均衡性。 数据分散集群中的特别角色 文授 F性写操作。 \$\$ Failb 回操作。 \$\$ Failfast 有。 3. 双流涂下下水时、2. X 加口/16.水平时 项数据角色、数据中中都原理的中一条户端只能将依据三别主机。数据分散身 实构中、各户端可以向任置服务等中等与数据(2 应用场景、数据集中表标注 数据最大水、集新创意费量不多的分景。 20.046cm 全集下一个位数机 最级的成果所。由于其设好的中特征。运合业务依据重压大、集新机器参 发现企业企业。2 1166cm 全排一个成本。 操作。 황**周沙賀用**,消费者发起服务调用之后,同步阻塞等待服务端响应。 \$**3.异** 发起服务调用之后,不阻塞立即返回,由服务端返回应答后异少通知消费者。 消费者同时对多个服务提供者批量发起服务调用请求,批量发起请求后,集中 『新典者同时対多个服务提供者批量发生服务適用请求、批量发起请求后、集 対表者的対象 支持・主義特殊者持致、支持私有协议定制和が展。 **33 公有的** であるのである。 は10 であった。 10 であった。 11 であった。 12 であった。 13 であった。 13 であった。 14 であった。 15 であった。 16 であった。 17 であった。 17 であった。 18 である。 18 であった。 18 である。 18 である。 18 である。 18 である。 1 Chapter4-2 med 相反。它的目的是尽 Mesos 组成 数据库和数据库实例区别与联系

(3) (1) 新闻电影中的电影,是使取得影響模型现象
(3) (1) 新闻电影中的电影,是使取得影響模型现象
(3) 新闻电影电影,是使用现象
(4) 有效数据类别是更多。是使取得数据类别是是一种。
(4) 数据类别是一种。
(4) 数据类别是一种。
(4) 数据类别是一种。
(4) 数据类别是一种。
(5) 数据类别是一种。
(5) 数据类别是一种。
(6) (方式 \$\$二进制类序列化 \$\$文本类序列化 . **统一配置 \$\$ 本地静态配置** 安装部署修 配置文件中。**\$\$ 基于配置中心的动态配置** 選判 Maratho K8S **组成** \$\$\$注册中心),\$\$\$之后统一下发,实时 幂等性的概念,重要性和场景 分库分表的定义 体分音位从高层级质分。是对数据运行分而没之的通用概念、为了分数数 结构证月,采用价度分差稀一个表结构为为多个差。或者稀一入表的数数分 层的效金分壳。这些表可以效在同一个单生,也可以按照不同的库里。甚至 以效本不向数据要求例 更是最少,但是是多的程度,将原本的一个件(图)另外分合个库。 是一个样。(图)对外分多个库(图)。 一个年(图),对分为多个库(图)。每一年,但图的有一种的形象。 分库分类的发展阶段 日生年来。(日本年来)。) 幸的幂等性及解决方案 幂等性的影响往往作用在数据上,而不同数 第一章 \$\$ Predic Chapter4-1 。 平滑迁移的做法 数据的定义 \$ 目标 将旧缓存中的历史数据迁移到新的缓存集群。\$\$ 实现 发迁移工具,并验证数据迁移的成功。\$\$ 优化 在某些情况下 据 仿佛新光洞安地⁴⁴⁴² ^{的通称。} 数据与信息的联系和区别 1)数据是信息的表达、载体。信息是数据的内 设有意义,数据只有对实体行为产生影响时才 3. 数据与语义的关系 \$ 目标: 将所有读操作转移到新的缓存集群。\$ \$ 实现 5. 效益中一個人口多人以 (防衛的表現於近天衛宇含多次其内容,廣東縣过鄉程,數据的古义称 道書等不可分的。可數的數等一部对数据含义的说明,数据的古义称 4. 数据的数件价值 4. 数据的数件价值 数据·语义·强盟·当类(码。业务一数件应用系统 5. 数据带来的变化 直接在应用决策取分片规则,然后解析分片规则,据此实现切分的路由 从应用层直接决定每次操作应该使用哪个数据库实例。库、表等。[2] 从少务,但实现第一。运合性重线,现分逻辑由并发者自行交、容易 维护,但要求开发者既要实现业务逻辑,还需要实现框架需求。[3]该实 点计数据度性协作连性分类。对象体应用服务器论的维护格语所干扰。 让数据库保持的连接比较多,对 定制 JDBC 实现方式 可让开发者集中精力实现业务逻辑 DBC协议来实现。也就是针对业务 IDBC 的内部实现。(3) 开发者需) 数据: 海量离散 (2) 语义 用性到跨界,自定义用户 非发复杂分布式异构系统 数据架构的变化 分类 URL 的章洵字符串(即问号后面的参数)传递参数 分传递参数。 <mark>句的服务集成(和微服务对比)</mark> **^{类则影响。} 致性哈希的缺点和解决办法 通过定制 ORM 实现 19、**停机工移比) 政法及 坪町** 路**备法**① 傳机应用,先將应用傳止服务 ② 迁移历史数据,按照新的規則把历史数据 应用的数据源配置,指向新的媒存集群 ④ 重新启动应用**转 评价** 该方式简单,高效 要由业务为评估影响,一般在晚上访问量较小、或者非核心服务的场景下比较运用。 ※**夜仔原则** 繁被访问的数据可以被放置于缓存当中,以供频繁访问。 垂直分离。 (**缓存原则** 1业务万评估影响。 一般 **缓存导致的问题** 一致【工门或即以床(四个IIII) 化对流 缓停属于持久数据的一个副本,所以不可避免的会出现数 1) 年写歷年,再写数据库 【描述】提布写成功,但写数据库 深。【解决】这个写耀存的方式,本身就是错误的。需要改 "推述】写数据库成功,但写耀存失败,则下次误败(针之 10. Mysq! 主从复制和读写分离做法 1±从复制 各页角份,据与各份,保证按照的注意性(C语写分育、揭示改 。 查力场。读的现在分片从查询的数据等多为高级规则。与可称能从于 11. 为什么使用外压分离。水平平折分,Mysql 集群 12. 为什么使用外压分离。水平平折分,Mysql 集群 12. 放出每十四数据和一一是当时的。在未进行为带力条的排发下,随着时间 12. 数据单中的数据和一一是当时的。在未进行为带力条的排发下,随着时间 12. 数据集中的数据和一一是一个数据制度,就是基本。和思想是 12. 数据集中的数据和一个是一个数据制度,就是基本的。如果是 12. 数据集中的数据量处理的,数据集中的 /构数描述行外方。/存数插: 文化更新频率证,宣词次数 化更新频率高,活跃的数据。(3)也可以人为将一个表 例如将查询较多,变化不多的字段拆分成一张表放在查 将頻繁更新的字段拆分并部署到更新性能高的服务器。 3 小 ★ (1) 单序单次和 (1) 单原单级点 (1) 数据的 (1) 数据的 (1) 数据的 (1) 数据的 (1) 数据的 (1) 数据的 (2) 数据的 (2) 数据的 (3) 数据的 (2) 数据的 (4) 数据的 (2) 数据的 (4) 数据的 (2) 数据的 (3) 数据的 (3) 数据的 (3) 数据的 (4) 数据的 (. 缓存雪崩的原因和解决办法 架构的数据去中心化的表现 •1.1 系统体系结构概述 ○ 软件架构 1-5 ●1.1 系统体系结构概述 ●1.1 系统体系结构概述 ●1.2 系统体系结构概述 ●1.2 系统体系结构与中间性 ←1.2 系统体系结构与中间性 ←1.2 系统体系结构与中间性 ←1.2 系统体系结构与中间性 E. 不足..... 自治。 **容与迁移** Side 2-2 - 22 常用软件设计模式 单例模式 1-8 1/模式 9-16 适配器模式 17-22 桥接模式 者模式 4-50 访问者模式 51-55 Chapter 3 - 计算层的软架构技术 Side 3-1 歌夢教報分高问题。
対于一些动前 据分离。
选取时间点对静历史数据进 查询问题定义和解决方案 lide 3-3 •3.3 负载均衡 二・MUTIMEAC、小中所で八万余
会ぐ弁以后、知度海的体施星や片的土壁、則可以適益分片提削再次路 歯詢,但是对于其他主键的套詢、范围查詢、美联查詢、查詢結果排除。 升下不長提別方字表理度来書面前。 美間裏面接欠力震 (1 左多个分片 海后台片發展集 (改革保証) (2) 任告海衛末定义多分片煙底 形成多松 (2 回接時间) (3) 通过搜索引擎源火、如果有卫级束、火器要实际 21. 互注九条构定义 各和直接建立状态传递的渠道(1)可以是网络连接(如、多开一个端口以及非网络连接(周串口数连接)(2)可以是主机效道状态络备机。也相拟故主机的状态(3)可以和致增级制造道共用。也可以独立一条通道的传递通道可以是一条。也可以是多条。还可以是不同类型的通道报合。 8+#口) 如何充分利用主备自动倒换方案 Chapter4-3 ()主备机共享一个对于客户端来说唯一的地址(如,虚拟 IP) (2)客, 6机各自的 IP,备机具有拒绝服务的能力。 o 进程通信模型 1-6 ○MOM 面向消息中间件&消息队列 7-18 後則各自的P,备机具有距离場合的能力 23、**互连式架构缺点** 以收态传递通道本身故障了,则备机会主动升级为主机 (2)虽然可以 採除低通道故障的机率,但是通道越多,后续的状态决策越复杂; 约岁多种矛盾的信息。 24、**中介式架构定义 文架构技术** 4.1 数据驱动的软件架构造化 1.4 数据驱动的软件架构造化 1.4 数据读写与主从公装调带来的架构变化 5-13 -- 4.2 数据读写与主从公装调带来的架构变化 5-13 -- 5 数据缓存基本理论 1-12 多种类型的状态传递连接通道,只 连连接重杂度(2)**次态决策**更简单,无 何决策的问题 **像**点:结构复杂了。 5. 中介式架构决策步骤)无论主机还是备机,初始状态都是 降级为备机,因此可能出现双备机 (卡如各机。各机能被并被分主机(2) 分布式缓存 14-19

模拟式架构定义

・ 分産分表概念 1-5 ・ 分库分表解次方案 ・ 分库分表解次方案 ・ 分库分表架构设计 ・ 単直切分 12-14 ・ 大学分 15-19 ・ 分库分表中间件