

一种互操作协议性能评估方法

张弩云¹, 李军国¹, 黄 罡¹, 梅 宏¹, 叶 萌², 陈 滢²

(1. 北京大学信息科学技术学院软件研究所, 北京 100871; 2. IBM 中国研究院, 北京 100085)

摘 要: 在异构网络环境下, 互操作协议对应用系统具有重要意义, 互操作协议的性能对应用系统的性能也起着举足轻重的作用. 为确定应用系统性能瓶颈, 必须评估互操作协议性能. 本文提出了一种互操作协议性能评估方法. 该方法针对实际应用系统中的互操作协议, 提出符合实际的评估指标, 对协议性能进行定性分析和定量分析. 在定性分析中得出安全性、容错能力、可扩展性等性能指标. 在定量分析中, 建立仿真实验模型与排队论数学模型相结合的混合模型, 得出网络带宽占用、响应时间与吞吐量等性能指标. 综合这些指标, 我们给出互操作协议性能评估的结果.

关键词: 性能; 互操作; 评估; 实验模型; 数学模型; 定性分析; 定量分析; 排队论

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2006) 12A-2355-05

An Approach to Performance Assessment of Interoperability Protocols

ZHANG Nu-yun¹, LI Jun-guo¹, HUANG Gang¹, MEI Hong¹, YE Meng², CHEN Ying²

(1. Institute of Software, School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China;

2. IBM China Research Laboratory, Beijing 100085, China)

Abstract: In today's diverse network environments, the interoperability protocols play an important role in performance of application systems. To assist performance assessments of application systems, we have proposed an approach to assess performance of interoperability protocols. The approach takes the real conditions of the system into account and proposes assessment criteria to make reasonable qualitative and quantitative performance analysis of interoperability protocols. In qualitative analysis, we obtain security, fault tolerability, scalability, etc. In quantitative analysis, we build up an experimental model and a math model to obtain performance quantities such as response time, net bandwidth, and throughput. Finally, according to the criteria we have proposed, we produce the result of performance assessment of interoperability protocols.

Key words: performance; interoperability; assessment; experimental model; math model; qualitative analysis; quantitative analysis; queuing theory

1 引言

互操作是对信息的交换和使用. 目前广泛使用的 RPC, SOAP, Java RMI, IIOP 等协议都是互操作协议. 网络环境下的各种软件系统, 要在分布和异构的情况下交换和使用信息, 就需要使用互操作协议. 为了从众多互操作协议中选出最适合应用或性能最优的协议, 需要进行互操作协议性能的评估.

目前国内外对互操作协议性能评估的研究可以分为两类: 仿真实验与数学建模. 仿真实验方法直接测量协议运行指标, 或搭建仿真实验平台, 在平台运行时测试协议性能. 这样得出的数据准确, 但测量结果只能反映已建成或正在运行的系统的情况, 若系统发生变化, 须再次测量或做仿真实验. 这样的测量往往耗费大量时间. 数学建模方法主要以排队网络和随机 Petri 网为基础进行数学建模. 随机 Petri 网系统, 借助

变迁的发生概率和马尔可夫链来评估系统性能. 由于变迁的发生概率无法得到准确值, 以这些值为基础进行评估也就缺乏可信度. 排队网络系统里, 以排队理论为基础, 借助分解系统为网状结构并且综合网中每个节点的性能来评估系统性能. 由于协议分解的不确定性和各部分性能不易获得, 该方法的适用范围有限.

在实际中, 大型企业应用对互操作协议及其应用系统有特定需求: 首先, 企业应用要求在有限的时间和资源条件下预测系统性能, 而不是在系统建成后测量. 其次, 企业关心不同互操作协议性能的对比, 而不是协议性能的绝对值. 因为他们只是从现有互操作协议中选择一种最适合应用的, 而不是自己开发协议. 再次, 企业对应用系统不仅有性能要求, 还有质量需求. 有时, 企业对质量的需求会超越对性能的需求.

现有互操作协议性能评估方法不能很好满足上述实际企

收稿日期: 2006-09-22; 修回日期: 2006-11-28

基金项目: 国家自然科学基金重点项目 (No. 60233010 及 No. 60403030); 国家 973 重点基础研究发展规划 (No. 2002CB312000); 国家 973 重点基础研究发展规划 (No. 2002CB312003); 国家 863 高技术研究发展计划 (No. 2005AA112030); IBM 大学合作项目

业需求:仿真实验方法不能满足企业预测系统性能的需要;数学建模方法需要的参数实际中往往得不到;对特定协议性能的精确评估不一定是企业应用所需要的.为此,本文提出了一种互操作协议的性能评估方法.该方法结合专门设计的仿真实验模型及以经典排队论为基础的数学模型,使得协议的性能分析既基于真实实验数据,又具有预测能力.我们还针对互操作协议的其它质量因素进行定性分析,综合评估互操作协议,满足了企业多种实际需求.

2 互操作协议性能评估方法

学术界对软件性能的一种定义是:软件系统或组成部分满足它的目标或者时间限制的程度.互操作协议是一种特殊的软件,其性能定义也应该符合软件性能的定义,即互操作协议或组成部分满足它的目标或者时间限制的程度.性能也是一种统计性质,不是指某次运行会怎样,而是计算在重复运行中呈现出来的规律.在实际中,企业应用对互操作协议的性能要求也是围绕响应时间、延迟时间、吞吐率等时间相关指标.为满足企业实际需求,吸取学术界已有研究经验,我们确定互操作协议的评估目标为:获得在相同负载下,不同互操作协议的对客户请求的平均响应时间、时延、吞吐量、对系统资源的占用和由协议造成的丢包率,从而得出不同协议性能的对比.

由于本文第一节所述理由,我们不能采用已有的方法,如实际测量或者仿真.因为测试不能预测系统性能;仿真需要花费极长时间来编程和运行仿真模型,而且大型企业系统的软件运行环境很难模拟.因此,我们采用建立数学模型的方法,利用数学表达式来描述系统,通过解方程组得到需要了解的参数(响应时间、吞吐率等).我们的数学建模方法以排队论为理论基础.排队论的缺点是,为了得到关于我们关注的参数的方程,不得不做出简化的假设.但实践证明,排队论得到的结果与一个更详尽细致的仿真分析得到的结果相当接近.它也是目前研究最多最成熟的网络性能评估方法.排队论对简化的假设的要求,正是我们在不清楚软件运行过程的情况下评估互操作协议性能所需要的.建立排队论数学模型,能够准确预测系统性能,满足企业对互操作协议性能对比的要求,而且省时省力.我们的数学建模方法,不同于以前的方法,它不像随机 Petri 网那样使用变迁发生的概率,也不像排队网络那样分解互操作协议,而是建立单层排队模型,即把互操作协议当作一个整体来研究.现有基于排队论评估系统性能的方法,通过系统各部分性能推导出系统整体性能.但在企业应用中,互操作协议和业务逻辑交织在一起,系统划分很困难,各部分的性能很难得到.我们采用单层排队模型方法,既避免了实际中互操作协议每部分执行时间和概率难以获得的问题,又满足了对比互操作协议性能的目标.

建立排队模型的关键,是获得客户请求到达分布、服务器处理时间分布、服务器数目、缓存、服务规程等参数.服务器处理时间参数即互操作协议执行时间,由协议决定;其它参数则决定于协议运行的软硬件系统配置.为了获得互操作协议执行时间的分布,我们又建立了实验模型,搭建仿真实验平台测量互操作协议对不同类型和大小的单次客户请求的响应时

间.因此,我们的方法里又引入了仿真实验环节.

针对企业提出的不可量化的质量需求,比如协议字段的可扩展性,我们采用定性分析的方法给出结果.

综上所述,我们提出如图 1 所示的互操作协议性能评估方法,它分为两个层次:

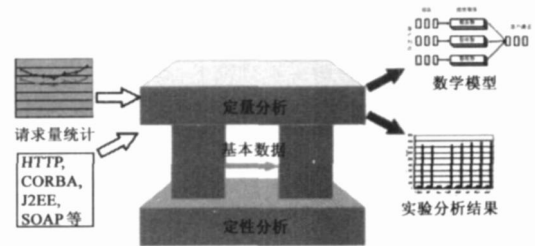


图 1 互操作协议评估方法体系结构图

第一个层次,互操作协议性能的定性评估:根据互操作协议的定义、体系结构及其关键实现技术,分析其安全性、可靠性、容错能力、可扩展性等.例如,根据 IIOP 协议的定义,我们就可以知道它受限于防火墙,而 HTTP 协议可以穿越防火墙.这个层次的分析解决了企业应用对互操作协议的全面质量需求.

第二个层次,互操作协议性能的定量评估:首先,针对互操作协议及其实现方式,在实验室搭建测试平台,获得单次互操作协议请求的响应时间,即服务器端收到请求的时刻和返回结果的时刻之差.这个响应时间不是传统性能评估数学模型中某个处理步骤的时间,而是协议整体处理时间,该数据是我们的方法中数学模型的输入参数来源之一.其次,分析互操作协议可能的应用场景,并在此基础上建立相应的数学模型.我们建立多个多服务器队列的排队模型,而不是排队网络模型.建立模型的思想是:根据客户请求达到的分布、互操作协议响应时间的分布、运行互操作协议的服务器或者线程的数目、队列限制、请求源特征等条件,建立排队论模型,得到请求处理时间和时延的均值、最大处理事务数、丢包率.比如,客户请求到达符合泊松分布,响应时间服从负指数分布,我们建立 $M/M/N/K$ 模型,使用经典的排队论求解公式求解;如果客户请求到达符合自相似分布,响应时间服从一般分布,我们将根据分布的具体参数建立相应多个多服务器队列的排队模型,使用自相似数据量下的特殊解法求解排队论模型.

最后,我们还可以结合互操作协议的应用场景,获取实际客户请求分布、服务器处理时间分布下的排队时间和丢包率等性能数据,对模型进行修正,使模型的精确度进一步提高.

3 实例研究

本节中,我们将以某电信公司为例,说明我们提表 1 互操作协议性能的定性评估示例列表出的方法的实际应用.该电信公司分别采用 CORBA 协议与 SOAP 协议作为短信服务与彩信服务的互操作协议.短信业务量巨大而服务质量良好.彩信业务量相对较小,但日平均丢包率高达 20%.该公司要求得到几种互操作性能的对比,并且要求得到互操作协议的多种质量属性.结合这一背景,我们研究 Socket、GIOP/IIOP、RMI、Http 和 SOAP 这五种互操作协议的性能.

3.1 互操作协议性能的定性评估

质量属性的分析结果, 共有 27 项指标, 这里给出部分作为示

我们依据各种互操作协议的定义, 通过理论分析给出其 例, 如表 1 所示.

表 1 互操作协议性能的定性评估示例表

	TCP/UDP SOCKET	GIOP/IOP	RMI	SOAP
编程语言	多种编程语言如 C、C++、Java 等	多种编程语言如 C、C++、Java 等	Java 语言	多种编程语言如 C、C++、Java 等
数据编码	字节	CDR	对象序列化	XML
异构性	独立于硬件、操作系统、编程语言	独立于硬件、操作系统、编程语言	独立于硬件、操作系统	独立于硬件、操作系统、编程语言、中间件
传输可靠性	TCP SOCKET 保证可靠传输; UDP SOCKET 的可靠性由应用层负责	GIOP 保证 at most once 语义	传输协议保证 at most once 语义	WS Reliability(OASIS)
转发支持	不支持	IOP Fragment 支持	IOP Fragment 支持	WS-Routing 规范和 WS-Referral 规范
穿越防火墙能力	端口不固定, 非明文传送	端口不固定, 非明文传送	端口不固定, 非明文传送	端口固定(HTTP 80), 明文传送
消息的字段可扩展性	无	无	无	可自定义 tag 和属性名
传输安全性	Secure Socket Layer (SSL)	IOP SSL, CORBA Security Service CSI level 0	HTTPS, IOP SSL	SSL/TLS, WS Security (OASIS)

3.2 互操作协议性能的定量评估

3.2.1 实验模型的建立及测量结果分析

建立实验模型, 我们测量各种互操作协议对客户请求的响应时间和网络带宽的占用. 响应时间直接通过实验进行测量, 网络带宽的占用则通过消息的大小来进行衡量. 测量场景为单客户端单服务器环境, 首先由客户端向服务器发送一个请求, 请求消息中仅含用于指示响应消息类型和大小的参数, 服务器收到请求消息后, 根据消息的内容, 向客户端返回包含特定类型或大小的数据响应消息.

图 2、图 3 分别表示发送不同类型/ 大小的数据时, 服务器的响应时间; 图 4、图 5 分别表示发送不同类型/ 大小的数据时的消息长度.

(2) 不同互操作协议在发送不同大小数据时的响应时间不同, 其结果见图 3.

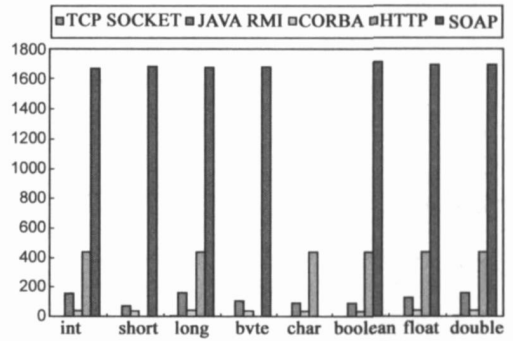


图 4 发送不同类型数据时的消息长度图

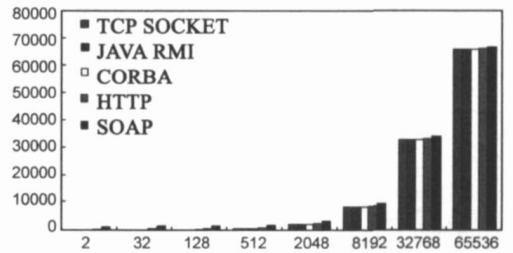


图 5 发送不同大小数据时的消息长度图

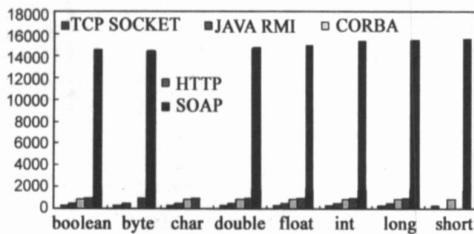


图 2 发送不同类型数据时的服务器响应时间图

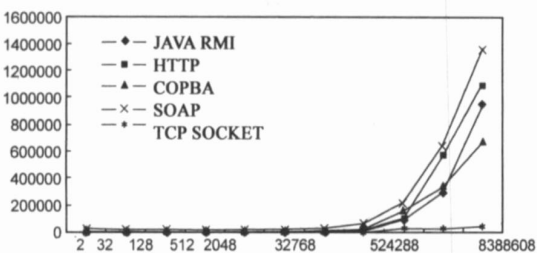


图 3 发送不同大小数据时的服务器响应时间图

通过对上述 4 图的分析, 可得到以下结果:

(1) 不同互操作协议在发送不同类型数据时的响应时间不同, 其结果见图 2.

从响应时间来看 CORBA/IOP 和 JAVA RMI 的性能要优于 HTTP 和 SOAP. 从网络带宽占用来看, 当发送的数据较小时, CORBA/IOP 和 JAVA RMI 占用的带宽要小于 HTTP 和 SOAP 占用的带宽; 当发送的数据较大时, 四种互操作协议的差别不大. 这些实验结果将是下一步数学建模的基础.

3.2.2 数学模型的建立及模型求解结果

根据研究, 传统电信行业的客户到达服从泊松分布, 因此我们采用 M/M/N/K 模型建立排队论模型, 即假设: 客户达到服从泊松分布; 服务时间服从负指数分布; 所有 N 个服务器负荷相等; 响应请求使用先进先出的调度规则; 客户队列长度

有限或者无限, 有限长度的情况下, 将有顾客从队列中被丢弃. 假设中所需参数由我们的实验确定.

平均服务时间, 即互操作协议的平均响应时间的算法是:

$$T_s = \sum_{i=1}^n p_i t_i P(t) = \frac{e^{-wt}}{w} T_{\text{pack}}(b) = f(e^{2\ln 2 \times b})$$

$P(t)$ 表示服务时间等于 t 的概率; w 是负指数分布的参数, 是常数; $T_{\text{pack}}(b)$ 是处理位数为 b 的数据包消耗的时间; b 是数据包长度.

由于缺少实际请求数据分布, 我们采用估计平均值的方法: 假设 1~128k 的数据出现的概率是 80%, 128k~8M 数据出现的概率是 19%, 大于 8M 的数据出现的概率是 1%. 即我们认为数据量极大的数据, 出现的概率极小; 数据量较小的数据出现的概率较小; 而数据量最小的数据出现的概率较大. 这种数据量分布用负指数分布来近似是合理的, 而且在同一假设下做出的服务时间估计, 基本不影响互操作协议性能的比较, 所以我们用负指数分布来表达服务器处理时间的分布. 如果可以获得互操作协议的实际应用场景, 这些分布参数可以得到修正.

根据上述方法得出各种互操作协议的平均响应时间如下 (单位 s):

RMI

$$T_s = \sum_{i=1}^n p_i t_i = (46 \times 80\% + 219 \times 19\% + 1371 \times 1\%) \times 10^{-3} = 0.092$$

其中, 46ms 是我们从实验模型中 (如图 3) 得到 RMI 对小于 128k 数据的响应时间, 219ms 是其对 128k~8M 数据的响应时间, 1371ms 是其对大于 8M 的数据的响应时间. 其他协议的响

表 2 五种互操作协议性能的比较

性能指标 协议	对单个请求的平均 响应时间(s)		资源利用率 (%)		队列长度		时延		最大处理事务数		丢包率 (%)	
	缓存无限	缓存为 20	缓存无限	缓存为 20	缓存无限	缓存为 20	缓存无限	缓存为 20	缓存无限	缓存为 20	缓存无限	缓存为 20
Socket	0.0026	0.0026	2.1781	2.1781	0	0	0.0026	0.0026	5769	5769	0	0
IOP	0.046	0.046	39.3672	39.3672	0.0008	0.0008	0.0470	0.0470	319	319	0	0
RMI	0.0336	0.0336	28.1496	28.1496	0	0	0.0336	0.0336	446	446	0	0
HTTP	0.019	0.019	15.9174	15.9174	0	0	0.0190	0.0190	789	789	0	0
SOAP	0.092	0.092	77.0684	75.7645	0.8629	0.3797	0.0989	0.0951	163	163	0	1.7

队列长度即缓存大小. 我们假设队列长度无限, 因为实际的计算机物理内存比较价廉, 可以完全满足需求. 我们也可以假设队列长度有限, 这样可以考察到几种互操作协议的丢包率; 在比较中, 队列长度取为相同值, 不管队列长度的具体数值是多少, 这不影响对互操作协议性能的比较.

最后, 我们给出数学模型的求解结果如表 2.

从表中可以清晰的看出五种互操作协议的性能对比情况. 例如, 在缓存为 20 的情况下, CORBA 协议的丢包率为 0%, 而 SOAP 协议的为 1.7%. 这两种协议的丢包率优劣与该公司的业务现状相符. 可见, 此数学模型准确的反映了协议性能的对比较关系. 同时, 该公司将模型得出的协议性能对比, 和定性分析得出得结论, 作为其今后进行系统改造和开展新业务时选择互操作协议的根据之一.

应时间计算方法与此类似.

SOAP

$$T_s = \sum_{i=1}^n p_i t_i = (11 \times 80\% + 165 \times 19\% + 684 \times 1\%) \times 10^{-3} = 0.047$$

IOP

$$T_s = \sum_{i=1}^n p_i t_i = (0.5 \times 80\% + 18 \times 19\% + 40 \times 1\%) \times 10^{-3} = 0.0026$$

HTTP

$$T_s = \sum_{i=1}^n p_i t_i = (10 \times 80\% + 84 \times 19\% + 940 \times 1\%) \times 10^{-3} = 0.0336$$

Socket

$$T_s = \sum_{i=1}^n p_i t_i = (10 \times 80\% + 101 \times 19\% + 1094 \times 1\%) \times 10^{-3} = 0.019$$

客户请求平均达到率的算法如下 (次/秒):

$$\frac{\text{请求总数}}{\text{请求发生的时间段长度}} = \frac{\sum \text{日请求总数}}{\text{天数} \times 24 \text{ 小时/天} \times 3600 \text{ 秒/小时}}$$

我们以电信公司实际数据为例, 客户平均请求达到率是

$$\frac{7.6 \times 10^7}{7 \times 24 \times 3600} = 125.66 \text{ 次/秒}$$

假设有 N 台服务器 (实际上, 服务器数量可以任意设置). 在我们的模型中, 对 N 为 1, 5, 10, 15, 20 进行了计算. 我们发现 N 取这些值, 可使 5 种互操作协议都满足应用需求, 又避免利用率太低; 且 N 取同一值, 不影响我们对互操作协议性能的比较. 这里服务器是处理单元的抽象, 不只是指物理服务器, 可以是并发的线程等.

该公司采用 SOAP 协议的彩信业务的丢包率高达 20%, 是用户欠费或认证失效造成的丢包. 根据该公司内部的统计结果, 由于协议造成的丢包率仅为 1% 左右, 这与我们得到的 1.7% 接近. 由数学模型得出的结果与真实数据存在一定差别, 是因为缺乏建立模型的依据和输入参数, 即实际客户请求分布和服务器处理能力分布. 如果获得这两个参数, 我们可以对模型作出修正, 使互操作协议性能评估结果具有更高的准确度. 由表 1 我们得到协议的质量属性, 由表 2 我们得到了协议的性能指标, 互操作协议的性能评估通过我们的方法全面得出, 并且该性能评估满足企业实际需求.

4 结论

本文提出的方法将协议性能指标分为可定性分析和可

定量分析的两类, 进行相应的定性和定量分析, 综合应用仿真实验模型与基于排队论的数学模型方法, 使得协议的性能分析既基于真实实验数据又能预测系统性能, 同时, 该方法还结合应用进行模型修正, 以优化互操作协议性能评估. 应用实践表明, 本文的方法对于互操作协议的性能的评估是切实可行的.

参考文献:

- [1] Christopher Kohlhoff, Robert Steele. Evaluating SOAP for high performance business applications: real time trading systems [A]. World Wide Web Conference [C]. Berlin: Springer, 2003. 255– 264.
- [2] Matjaz B Jnrlic, Bostjan Kezmah, Marjan Hericko. Java RMI, RMI tunneling and web services comparison and performance analysis [J]. ACM SIGPLAN Notices, 2004, 39(5) : 58– 65.
- [3] Dan Davis, Manish Parashar. Latency performance of SOAP implementations [A]. Proceedings of the 2nd IEEE/ ACM International Symposium on Cluster Computing and the Grid [C]. US: IEEE computer society, 2002. 407– 413.
- [4] Vibhu Saujanya Shama, Pankaj Jalote, Kishor S Trivedi. Evaluating performance attributes of layered software architecture [A]. Heineman George T. Component Based Software Engineering [C]. Berlin: Springer, 2005. 66– 81.
- [5] M Woodside. Modeling the performance of CORBA using layered queuing networks [A]. Tutorial Introduction to Layered Modeling of Software Performance [C]. New York: ACM Press, 2002. 189– 194.
- [6] M K Molloy. Performance analysis using stochastic Petri nets [J]. IEEE Trans Computers, 1982, C-31(9) : 913– 917.
- [7] Sven Bartelsen, Manfred MittlereOliver Rose. Approximate flow time distribution of a queue with batch service [R]. Germany: Bayerische Julius Maximilians University, Report No. 174,

1997.

- [8] William Stallings. 高速网络与互联网性能与服务质量 [M]. 北京: 电子工业出版社, 2003.
William Stallings. High Speed Networks and Internets Performance and Quality of Service [M]. Beijing: publishing house of electronics industry, 2003. (in Chinese)
- [9] Connie U Smith, Lloyd G Williams. 软件性能工程 [M]. 北京: 机械工业出版社, 2003.
Connie U Smith, Lloyd G Williams. Performance Solutions a Practical Guide to Creating Responsive, Scalable Software [M]. Beijing: China Machine Press, 2003. (in Chinese)
- [10] C Y Yuan. Principals and Application of Petri Nets [M]. Beijing: Publishing House of Electronics Industry, 2005. 1– 29.

作者简介:



张弩云 女, 1982年9月生于陕西西安, 博士研究生, 主要研究方向为分布计算、软件工程, 特别是软件部署相关技术.
E mail: zhangny04@sei.pku.edu.cn



李军国 男, 1978年生于内蒙古, 博士研究生, 主要研究领域为软件工程、中间件可靠性及相关技术. E mail: lij05@sei.pku.edu.cn