

舰载可靠的消息队列服务 样机系统

江永国, 刘强, 刘乾谦, 苏健, 秦长帅

中国海洋大学计算机与技术系, 青岛, P. R. China

摘要—针对舰载信息系统多源数据融合需求的特点, 对ActiveMQ进行了改进, 并构建了船舶鲁棒通信原型系统。该原型系统利用可靠的数据异步传输优势, 为船用信息系统(SIS)的功能平台提供灵活可靠的系统通信, 克服了异构系统的困难在SIS中的集成。该系统还通过提供多个消息队列来提供单个数据链接, 从而使本机Java消息服务(JMS)更加可靠, 从而改进了传统的消息队列模型。另一方面, 该系统提供了一个可扩展的群集解决方案, 该解决方案利用单独的配置中心来实现消息服务器节点的扩展、控制和监视, 并动态分配消息服务器和消息的负载队列。改进后的系统有效地克服了单线和单点信息源故障的困难, 变得更具干扰和抗损伤能力。

一、导言

随着海洋探测技术的不断进步, 在船舶航行、渔业捕捞[1]、科学调查[2]和海上执法[3]等领域获得大量信息。检测设备。同时, 船舶需要整合各种探测和监测设备, 如传感器网络(SN)、声学设备、红外成像设备、舰载雷达、无人地面飞行器(USV)、无人水下航行器(UUV)、无人飞行器(UAV)等但是, 大多数设备都是由独立供应商制造和开发的, 这些供应商使用不同的操作系统作为操作环境, 并依靠不同的网络架构[8, 9]使船舶信息系统(SIS)成为分布式应用程序系统。在传统的集中式船舶信息系统(SIS)中, 各种应用系统之间的直接通信, 连接关系复杂, 因此传统的集中式船舶信息系统通信效率相对较低[10]。另一方面, 这种通信模式不能满足系统的可扩展性和灵活性。同时, 根据美国船级社(ABS)对SIS的可靠性要求, SIS设计必须使用冗余设计来实现可靠性[9, 11]。因此, 需要一个可靠、灵活、可扩展的异构系统通信解决方案来满足SIS集成的需求。

目前, 企业级分布式应用程序系统集成解决方案通常使用消息中间件(MOM)来实现异构系统的集成。MOM提供了一个跨平台的异步消息传递解决方案, 保护底层复杂的操作系统和网络体系结构, 以确保分布式网络环境中应用程序之间的可靠性、跨平台数据交换[12]。现有的大多数MOM产品可以提供更好的通信服务, 但在最恶劣的通信环境中缺乏服务质量(QoS)支持, 无法满足通信模块。例如, 基于发布/订阅模型的“轻量级”通信协议MQTT[13], MQTT擅长在网络环境不佳的服务场景中提供实时可靠的数据传输, 但往往会导致内存拥塞传输大型文件时。适用于板载视频数据和图像数据等应用系统的集成;ZeroMQ[14]是一种基于消息队列的多线程网络库, 由于其无代理通信模式, 可以提供高吞吐量和低延迟的通信服务。它不太可靠, 不适合SIS应用要求。

为了满足SIS对实时性、可靠性和可扩展性的要求, 本文设计并开发了一个基于ApacheActiveMQ(AMQ)的健机消息队列服务原型系统[15]。AMQ是Java消息服务(JMS)标准[12]的开源实现。其中, JMS包括对等通信和发布/订阅两种通信模型, 并提供了可靠的信息传递机制。另一方面, AMQ利用消息代理来管理消息队列, 并提供了完整备份的冗余设计, 实现了消息代理的聚类, 保证了信息源的可靠性。但是, 鉴于硬件设备在海上环境中的性能有限, 这种数据冗余比例可能是不可接受的。因此, 在保证消息冗余备份的前提下, 我们采用统一的配置管理中心来管理冗余消息队列和消息代理, 实现消息传输服务的可靠性和可扩展性。

我们的主要工作包括以下两个方面：

- B. 另一方面, 系统提供了一种可扩展的聚类方法, 利用配置中心以统一的方式管理主题队列实体和消息服务器, 实现消息中消息队列实体的动态调整服务器和队列服务组, 并保证消息。可靠性。

二、系统设计

A. 系统集成架构

面向消息的中间件是一种使用消息队列模式的系统间通信技术。在分布式系统中, 生成消息的应用程序(生成器)将指定的消息发送到消息队列。另一方面, 与此消息相关的应用程序(使用者)接受来自消息队列的消息。消息的传递由消息队列处理。生产者和消费者不知道彼此的存在, 可以有效地实现松散耦合的系统集成。

本文以AMQ为基础, 结合船舶信息系统集成和应用的要求, 以消息队列组的形式, 为SIS的所有关键任务功能提供了一个完全集成且高度可生存性的通用网络, 包括控制、导航、监控和检测信息。

B. 消息队列原型系统框架

图2显示了船上嵌入式可靠消息队列服务原型系统的总体框架, 包括配置中心、AMQ群集和用户API。配置中心是原型系统的核心部件。它主要负责管理和监视消息队列、管理AMQ群集、监视AMQ节点状态、为生产者和使用者提供消息队列连接信息以及动态调整AMQ节点和消息的负载队列。AMQ群集由多个物理服务器组成, 每个服务器都有一个不同的AMQ代理节点。群集为SIS提供信息备份和消息传递服务。用户API为可快速集成到SIS中的应用程序提供了一个简单的编程接口。如图1所示, 消息服务原型系统的使用过程大致分为七个部分。

首先, 消息生成器询问配置中心它所需的消息服务是否存在。如果服务可用, 则该配置中心将返回消息服务使用的消息队列信息。消息生成器收到配置中心返回的消息服务信息后, 将根据该信息连接到AMQ群集, 并将消息发送到消息服务信息中的指定节点和消息队列, 如步骤3和4所示。使用者还需要执行步骤1、2和3, 从配置中心请求消息服务信息, 并根据信息连接到AMQ群集。然后, AMQ节点将信息从指定的消息队列推送到使用者。

如果生成器请求的服务不存在, 配置中心将注册新服务并执行步骤6以控制AMQ群集, 并打开群集中新服务使用的消息队列组。同时, 配置中心实时监视群集, 并根据步骤7中返回的消息队列的节点和信息确定消息服务是否处于正常状态。如果消息服务处于不正常状态, 则执行步骤6以动态调整消息服务使用的群集节点和消息队列。

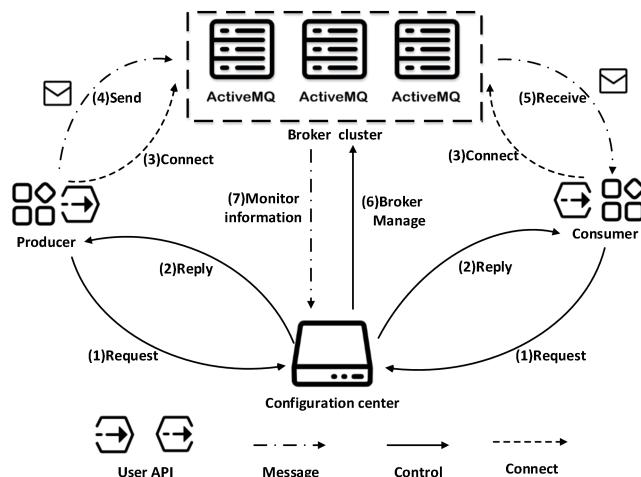


图1。消息队列原型系统框架

三、服务设计

A. 消息队列组模型

对于某个消息生成器, JMS使用消息队列实体提供对等消息和发布/订阅消息服务。消息队列实体通常存储在固定的消息服务器上, 生成器和使用者只能通过消息服务器发送和接收消息。即使使用多个消息服务器提供消息服务, 也不会关联每个消息服务器的消息队列。一旦消息服务器出现故障, 它所承载的所有消息队列都将不可用。为了提高消息服务的可靠性, 满足SIS设计中的冗余要求, 我们设计了一个新的消息队列组模型。消息队列组是多个消息队列实体的抽象, 包含不同消息服务器中的多个消息队列实体。当消息生成器生成消息时, 它会向消息队列组发送一条消息。消息被发送到不同消息服务器的不同消息队列实体。这意味着存储消息的消息服务器是未定义的。因此, 当单个消息服务器出现故障时, 生成器和使用者可以通过消息队列组中的其他消息队列和服务器执行消息服务。它能有效地解决单线故障问题。消息队列组模型如图2所示。

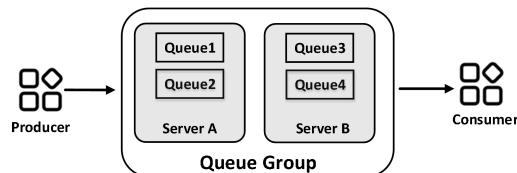


图2。消息队列组模型

B. 集群战略

大多数消息传递中间件产品都支持消息服务器的群集, 以确保消息源的可靠性。群集中的每个消息传递服务器都有一个单独的消息队列, 其中具有详细的消息分发功能。现有的聚类方案分为高性能和高可用性。卓越的性能方案旨在统一的服务时间内提供更高的吞吐量。高可用性方案旨在减少整个消息服务的停机时间。在SIS集成应用程序环境中, 消息服务设计的主要目标是确保服务的正常运行, 并提供高度可靠的灾难恢复性能。现有的消息中间件产品[15-17]提供了多种高可用性集群策略, 主要分为服务处理节点备份和数据备份两部分。服务处理节点备份用于备份消息服务器中的消息转发功能。数据备份是消息队列中消息的备份。可以在不同的应用程序方案中使用不同的群集策略。在SIS中, 服务处理节点备份和数据备份同样重要。但是, 备份策略与群集中硬件的大小密切相关。考虑到硬件设备在海上环境中的性能有限, 将完整备份用于服务处理节点和数据是不可接受的。为了保证服务节点的容灾能力, 我们希望服务处理节点采用完整的备份策略。另一方面, 消息队列数据使用不完整的备份策略来减少数据的冗余。消息队列组中的多个消息实体具有相同的消息数据, 从而实现数据的冗余备份。同时, 为了减少数据的冗余, 消息队列组Nq≤

