

# 基于 OpenDDS 和反射内存卡的信息分发技术

刘爽, 侯正平, 宋雷军, 申臻

(上海航天电子技术研究所, 上海 201109)

**摘要:** 随着军事领域中信息化系统的应用, 信息网络中对信息分发的实时性要求越来越高; 信息分发的实时性取决于通信模型、信息分发的策略和网络本身的物理特性; 开源数据分发服务 (OpenDDS) 采用以数据为中心的发布订阅通信模型, 为很多信息分发网络的实时数据通信提供了较好的解决方案; 反射内存卡作为通信介质, 其存储共享和快速读写模式可以减少网络损耗, 提高传输效率; 将反射内存卡与 OpenDDS 结合设计信息分发模型, 介绍了 OpenDDS 架构、OpenDDS 网络通信原理、和基于 UDP 的对象请求代理间协议 (DIOP); 通过研究 DIOP 下的远程调用通信流程、反射内存卡读写数据和自适应通信环境反应器 (ACE Reactor) 监控硬件以及多路分离通知机制, 使用反射内存卡作为底层通信介质, 通过反射内存卡组网的硬件设计和 OpenDDS 的软件设计, 实现了在 DIOP 协议上, 将通信数据通过反射内存卡传输, 提出了一种基于 OpenDDS 通信模型和反射内存卡的信息分发技术; 最后通过实验验证传输数据的正确性。

**关键词:** 信息分发; OpenDDS; 反射内存卡; DIOP

## Information Distribution Technology Based on OpenDDS and Reflective Memory

Liu Shuang, Hou Zhengping, Song Leijun, Shen Zhen

(Shanghai Aerospace Electronic Technology Institute, Shanghai 201109, China)

**Abstract:** With the application of information system in military field, information network requires better and better real-time information distribution. The real-time performance of information distribution depends on the communication model, the strategy of information distribution and the physical characteristics of the network itself. Open Data Distribution Service (OpenDDS) adopts a data-centric publish-subscribe communication model, which provides a better solution for real-time data communication in many information distribution networks. As a communication media, reflective memory's storage and fast read-write mode can reduce network loss and improve transmission efficiency. Information distribution model is designed by combining reflective memory with OpenDDS and OpenDDS architecture, OpenDDS network communication and Datagram Inter-ORB Protocol (DIOP) are introduced. By studying the remote call communication process under DIOP, reflective memory reading-writing data method and the monitoring hardware and multi-channel separation notification mechanism of Adaptive Communication Environment Reactor (ACE), using reflective memory as the underlying communication media, the communication data is transmitted through reflective memory and DIOP by reflective memory network hardware design and OpenDDS software design. An information distribution technology based on reflective memory and OpenDDS communication model is proposed. Finally, the correctness of data transmission is verified by experiment.

**Keywords:** information distribution; OpenDDS; reflective memory; DIOP

## 0 引言

军事信息系统中的信息分发网络作为数字化战场的中枢, 将各种信息按照通信协议规定的格式, 实时、快速、自动、保密地进行数据交换, 从而实现信息资源共享, 最大限度地提高武器平台的作战技能<sup>[1]</sup>。因此, 信息分发网络面临数据的分布性、准确性、实时性、正确性、协调性和安全性等挑战<sup>[2]</sup>。OpenDDS 是用于分发实时应用系统数据的网络中间件, 简化了分布式应用程序的开发、部署和维护, 应用程序简单地发布和订阅数据, 发布方和订阅方无需知道对方地址即可通信。其以数据为中心的发布订阅模式极大地减少了在网络上发送数据所需的消耗, 为大容

量传输网络提供了针对关键数据快速且可灵活配置的传输方法<sup>[2-3]</sup>。此外, OpenDDS 规范还定义了大量的服务质量 (Quality of Service, QoS) 策略, 使得 OpenDDS 可以很好地配置和利用系统资源, 协调通信质量和执行效率之间的平衡, 增加了通信的可靠性<sup>[4-5]</sup>。目前, 许多军事领域已广泛采用 OpenDDS 技术<sup>[6]</sup>。

OpenDDS 是建立在公共对象请求代理体系结构 (common object request broker architecture, CORBA) 上的应用框架, 远程调用通信是其信息分发的主要技术, 在订阅发布通信中起到重要作用, 默认情况下传输介质为网卡<sup>[7-8]</sup>。为了进一步提高信息分发的效率, 降低网络间通讯的时延, 屏蔽子网内部频繁通讯对整个网络中其他设备的干扰, 本文使用光纤反射内存卡作为底层传输介质替换网卡, 提出一种基于反射内存卡和 OpenDDS 通信模型的信息分发技术<sup>[9-10]</sup>。依据 CORBA 通信的原理, ACE 中 Reactor 监控硬件数据到和多路

收稿日期: 2019-06-26; 修回日期: 2019-08-10。

作者简介: 刘爽 (1990-), 女, 山西太原人, 硕士, 工程师, 主要从事软件研发工作方向的研究。

分离方法, 以及反射内存卡应用函数实现相应功能, 将通过 DIOP 完成的通信使用反射内存实现, 设计了 OpenDDS 下基于反射内存卡的信息分发网络。

## 1 系统结构及原理

### 1.1 OpenDDS

OpenDDS 采用以数据为中心的发布订阅机制, 其通信网络由一个或者多个数据域组成。每个域包含一套域参与者 (发布者和订阅者), 参与者为网络中分布式通信节点, 节点间使用 OpenDDS 通信。完整的 OpenDDS 网络域包括域参与者和主题, 域参与者又可分为数据写入者、发布者、订阅者、数据读取者。OpenDDS 中各分布式节点身份可以是发布者、订阅者或两者兼有, 节点间关系平等, 数据域内使用主题标识数据, 每个主题有唯一的名称、数据类型和一套 QoS 策略。发布者发布主题和数据, 订阅者根据自己需要的主题订阅数据。OpenDDS 网络发现模型为集中式发现模型, 网络通信节点中除了域参与者, 还需要一个 DCPSInfpRepo (信息仓库) 节点提供发现服务, 允许发布者和订阅者通过 DCPSInfpRepo 根据主题发现彼此。OpenDDS 模型提供匿名、对上透明、多对多的通信, 通信时随时可有新的节点注册入全局数据空间发布/订阅信息, 或随时退出该网络而不影响其余各节点, 解决了因网络结构动态变化需要重构整个通信系统的问题, 增强了通信组网的灵活性<sup>[2-3,11]</sup>。

OpenDDS 节点框架如图 1 所示, 最上层为应用程序, 底层为操作系统和通信介质, OpenDDS 作为网络中间件构建在 ACE 环境和 CORBA 模型上。其中 CORBA 使用远程调用完成了分布式通信的功能, ACE 完成了操作系统的适配、包装, 为 CORBA 通信提供了框架和模式。本文使用反射内存卡替换远程分布式通信中使用的底层硬件, 是在 CORBA 中间件的基础上实现的。

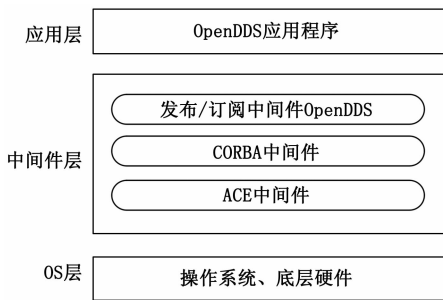


图 1 OpenDDS 节点框架图

### 1.2 OpenDDS 网络通信

集中式发现的 OpenDDS 网络通信分为分布式远程调用通信和本地通信。信息仓库 (DCPSInfpRepo) 节点提供发现服务, 允许发布者和订阅者通过 DCPSInfpRepo 根据主题发现彼此。DCPSInfpRepo 与发布者和订阅者之间采用远程调用通信。发布者和订阅者相互发现后, 发布者直接向订阅者发送数据, 使用本地通信。如图 2 所示。

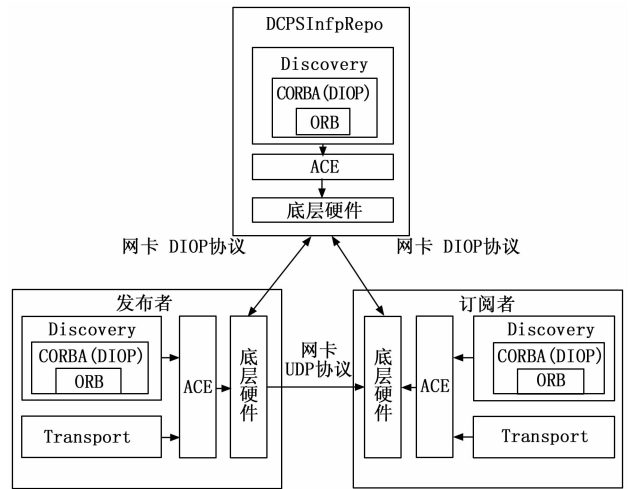


图 2 OpenDDS 通信模型

远程调用通信节点使用的 Discovery 模块包含 CORBA 中间件, 采用基于 UDP 的 DIOP 协议, CORBA 的核心是 ORB (对象请求代理), 通过 ORB 实现了客户端和服务对象的完全分开, 客户端不需要了解服务对象的实现过程和具体位置, 就可以使用服务对象。本文主要研究 OpenDDS 基于 CORBA 的远程通信。

### 1.3 CORBA 通信

CORBA 为可移植的、面向对象的分布式计算应用程序解决了远程对象之间的互操作问题。CORBA 基于客户端/服务器模型, 在 OpenDDS 框架中, DCPSInfpRepo 作为 CORBA 服务器, 发布者和订阅者为客户端。CORBA 核心是对象请求代理 (Object Request Broker, ORB), 通过 ORB 实现了客户端和服务对象的完全分开, 客户端不需要了解服务对象的实现过程和具体位置, 就可以使用服务对象。CORBA 通信原理如图 3 所示。

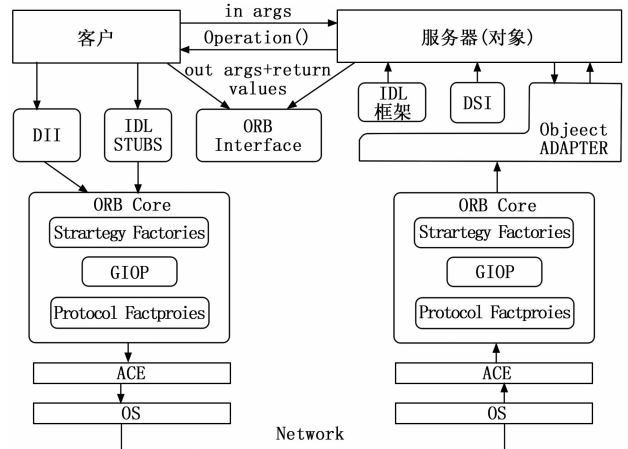


图 3 CORBA 通信原理图

CORBA 的基本工作方式如下:

- 1) 客户端使用静态存根 (IDL STUBS) 或动态调用接口 (DII) 提出请求, 将请求传递给 ORB 内核。
- 2) 客户端 ORB 内核通过网络传送到与服务器应用程序

序相连的服务器 ORB 内核。

3) 服务器 ORB 内核将请求分配给对象适配器 (Object ADAPTER), 产生目标内核。

4) 服务器 ORB 内核将请求分配给目标对象的伺服程序。

5) 伺服程序执行客户机请求后, 将结果返回。

客户端和服务端端的 ORB 必须协同起来完成工作。ORB 间通信协议为通用对象请求代理间协议 (General Inter-ORB Protocol, GIOP), 存在于 ORB 内核中, 结合不同的通信硬件, GIOP 可以具体化为很多协议, 基于 UDP 数据报的传输协议是 DIOP。DIOP 协议框架主要由 Acceptor, Connector, Connection Handler, Endpoint, Profile, Protocol Factory, Transport 类的子类组成。其中 Profile 是基于 CORBA 的 IOR 定义的, 封装了所有创建和解析一个特定协议的 IOR 所需的全部方法和成员, 在 ORB 启动服务端创建一个 Acceptor, 监听客户端发来的请求, 同时为每个客户端启动一个 Connector。当客户端涉及对象时, ORB 必须获得相应 IOR 的 profile 列表。该 profile 被选定后, Connector 注册会将一个具体的 connector 实例与相应的 profile 类型进行匹配并向 ORB 发出连接请求。而 Acceptor 一旦监听到客户的连接请求, 就将他送给 ORB 处理。只要连接关系确立, 客户端与服务端就可以通过可插拔协议进行数据通信<sup>[7-9]</sup>。DIOP 协议的主要部分继承了 7 个基类, 并加入了自己的特性。

#### 1.4 反射内存卡

反射内存卡类似于共享内存, 硬件安装在不同的计算机主机上, 所有节点共享同一片存储区域, 同一时刻允许系统中所有节点同时读写共享内存。由于不需要通信协议, 在任意一块反射内存卡被写入数据后, 可以在极短的时间内更新数据, 网络上所有计算机都可以访问这个新数据<sup>[11]</sup>, 传输延迟小, 可以实时进行信息传输, 更能满足大型通信网络的数据分发要求。

反射内存卡对于任意节点的数据读写没有硬件限制, 在实际使用时需要制定具体的协议监控数据收发, 数据的收发监控可以使用查询或中断模式: 查询模式需要有监控线程监控数据的读写, 中断模式需要使用反射内存卡提供的中断函数。

#### 1.5 ACE Reactor

ACE Reactor 处于 OpenDDS 中的底层模块 ACE 中间件。ACE Reactor 类似一种监控程序, 提供多路分离和事件分派框架, 简化了事件驱动应用的开发。通过在操作系统事件多路分离接口上进行侦听事件发生, ACE Reactor 触发对用户预先登记事件处理器 (event handler) 对象中方法的回调 (callback)。该回调方法由应用开发者实现, 其中含有应用程序处理此事件的特定代码<sup>[9-10]</sup>。

OpenDDS 中使用 ACE Reactor 进行 UDP 数据的接收和处理, 当 ACE Reactor 监控到某个 Socket 上 UDP 数据到达这类事件时, 触发此 Socket 对应的 DIOP 模块中的事件

处理器函数。这样的模式即查询模式, 由于 ACE Reactor 作为 OpenDDS 底层数据通信重要的环节, 当传输介质由网卡替换成反射内存卡时, 也使用 ACE Reactor 对反射内存卡上的数据进行监控。

## 2 基于反射内存卡通信的硬件设计

### 2.1 反射内存卡网络地址设计

反射内存卡允许系统中所有节点同时读写共享内存, 在 OpenDDS 分布式网络应用中, 设计最大有 128 个通信节点, 需要对每个节点的反射内存卡进行地址分配。反射内存卡的地址以反射内存 ID 标识, 反射内存 ID 由板卡硬件设置。网络中的节点要实现 ID 与 UDP 协议中的 IP 地址的对应, 这里使用算法:

$$IP = 0xc0a800 + ID;$$

DCPSInfpRepo 服务器在网络中心, 设置反射内存 ID 为 129, 其他反射内存节点为 1~127。由此 DCPSInfpRepo 服务器和其他发布者/订阅者对应的 IP 地址设为 192.168.0.129 和 192.168.0.1~192.168.0.127。

### 2.2 反射内存卡节点存储设计

反射内存卡存储区设计如图 4 所示, 每块反射内存卡地址分为 2 个部分, 状态区和数据区, 状态区记录每个节点是否可读写, 数据区记录每个节点收到的帧内容。由于 OpenDDS 网络通信分为分布式远程调用通信和本地通信, 远程调用通信为 DIOP 帧, 本地通信为 DCPS 帧。因此每个反射内存卡节点内部还需要针对不同的传输帧划分为 2 个数据, 将节点内部划分地址可以实现数据在单个节点上的分离。这里实现了 socket 端口到反射内存卡数据区的映射。

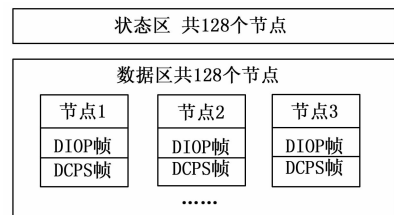


图 4 反射内存卡存储分配图

## 3 基于反射内存卡通信的软件设计

### 3.1 配置 OpenDDS 的 DIOP 协议

OpenDDS 使用 DIOP 前要配置 DIOP 协议, 方法如下:

1) DIOP 协议代码被包含在 TAO\_Strategies 库中, 将 TAO\_Strategies.lib 加入到 OpenDDS 应用链接库中。

2) 通过服务配置器动态加载 DIOP 协议: 在 DCPSInfpRepo 应用路径下创建 svc.conf 文件, 输入动态加载 DIOP 协议指令:

```
dynamic DIOP_Factory Service_Object *
TAO_Strategies:_make_TAO_DIOP_Protocol_Factory () ""
Dynamic Advanced_Resource_Factory Service_Object *
TAO_Strategies:_make_TAO_Advanced_Resource_Factory ()
"-ORBProtocolFactory DIOP_Factory"
```

3) 修改发布者/订阅者应用程序的配置文件 dds\_udp\_

conf. ini, 将其中的 DCPSInfoRepo = corbaloc::192.168.0.129:12345/DCPSInfoRepo

改成 DCPSInfoRepo = corbaloc:diop:192.168.0.129:12345/DCPSInfoRepo

4) 由于服务器和客户端都需要创建 TAO\_DIOP\_Acceptor, 在运行应用程序时, 服务器端和客户端需要声明 DIOP 端点, 方法是在命令行输入 -ORBListenEndpoints diop://主机号:端口号。

至此, 配置 DIOP 协议配置完毕。

### 3.2 DIOP 客户端注册

发布者和订阅者作为客户端与 DCPSInfoRepo 服务器端 DIOP 通信图如图 5 所示。

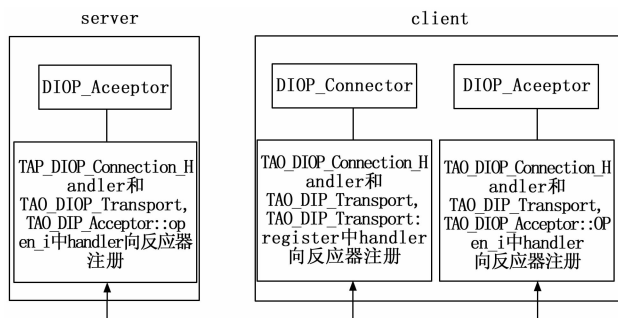


图 5 发布者/订阅者与 DCPSInfoRepo 服务器 DIOP 通信图

DIOP 协议中, 客户端 DIOP\_Connector 和 DIOP\_Acceptor 创建的 DIOP\_Connection\_Handler 和 TAO\_DIOP\_Transport 负责与服务器端实现通信, 为保证可以正确收到服务器发送的数据, 因此需要在 TAO\_DIOP\_Transport 中将 DIOP\_Connection\_Handler 注册到反应器上, 使得反应器可以监控 handler 句柄上是否有数据到来。具体操作为在 TAO\_DIOP\_Transport::register\_handler 函数中加入代码:

```
ACE_Reactor * const r = this->orb_core->reactor();
if(r==this->event_handler_i()->reactor()){
    return 0;
}
this->ws->is_registered(true);
return r->register_handler(this->event_handler_i(),
ACE_Event_Handler::READ_MASK);
```

### 3.3 反射内存卡通信设计

在 OpenDDS 的 DIOP 上实现反射内存卡的数据传输, 需要解决反射内存卡数据监控问题, 这里的数据监控使用程序查询模式, 通过 ACE Reator 触发接收回调函数接收数据。图 5 描述了在 DIOP 中使用反射内存卡作为通信介质的实现方法。

如图 6 所示, 在服务器端 DIOP\_Acceptor 类中创建反射内存卡监控线程函数, 用于监控数据是否已经写入反射内存, 监控到反射内存有数据写完成时通过 socket 向本机发送可读的 1 字节数据, ACE Reator 监视到事先注册在其中的 socket 有数据到来, 进而能够触发到事先注册在 Reac-

tor 中注册的事件处理函数接收数据。反射内存卡监控函数作为一个单独的线程创建位置较为重要, 既要满足用于监控网卡数据的 socket 已经建立并完成监听端口绑定的初始化之后, 又要满足发送端发送第一帧数据之前, 以确保接收端能及时准确地监控到发送端写入反射内存卡的所有数据。这里监控线程函数的创建位置是 DIOP\_Acceptor.cpp 的 open\_i 函数中。

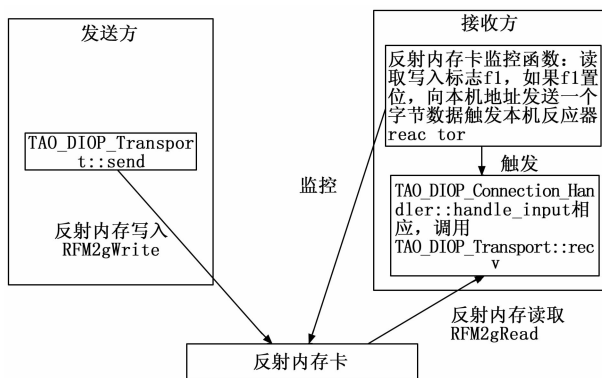


图 6 OpenDDS 反射内存卡替换原理图

在触发接收函数后, 在接收数据函数内部从反射内存卡对应地址读出发布端写入的数据。由此, 基于 DIOP 的反射内存卡数据传输, 客户端向反射内存写数据, 服务器端从反射内存读数据的单词流程得以实现。服务器端向客户端传输数据流程与此类似。

### 3.4 反射内存卡数据同步

由于发送和接收方是 2 个不同的进程, 而操作反射内存卡的同一块地址范围可能会出现接收端还未将前一次数据取走, 发送端已将下一帧数据写入的丢帧问题。为解决数据在反射内存卡中数据共享时的同步问题, 程序中创建读写标志位, 即设置两个标志:

$f_1$ : 写入标志位, 置位时表示发布端已将完整一帧数据写入反射内存卡对应地址。

$f_2$ : 读取标志位, 置位时表示订阅端已将数据从反射内存卡对应地址读走, 此时可写。

标志位配合反射内存读写的流程如图 7 所示。

1) TAO\_DIOP\_Transport::send 依据读标志  $f_2$  是否置位, 决定是否将将要发布的数据写入反射内存卡, 写入后清除读标志  $f_2$ , 置位写标志  $f_1$ 。

2) 线程函数 ThreadProc 依据写标志  $f_1$  是否置位, 决定是否向自己进程的发送 1 个字节通知数据以便触发 ACE reactor, 并清除写标志  $f_1$ 。

3) TAO\_DIOP\_Transport::recv 首先通过 this->connection\_handler->peer().recv() 接收由线程函数发来的一字节通知数据, 从反射内存卡取出订阅的数据, 并置位读标志  $f_2$ 。

## 4 实验结果与分析

实验验证使用 3 台计算机分别作为 DCPSInfoRepo, 发布者和订阅者, 运行 DCPSInfoRepo 服务器程序如图 8, 命

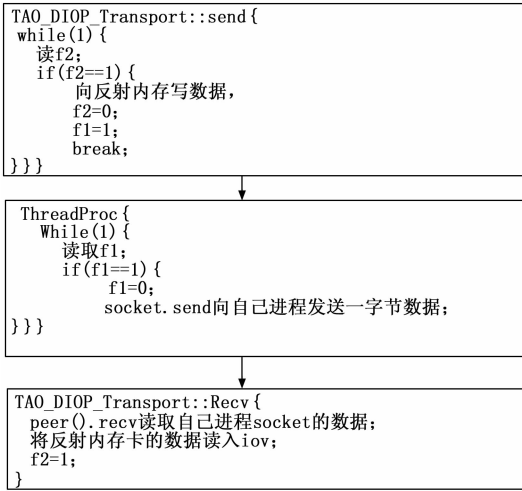


图 7 反射内存卡读写流程图

命令行输入:

DCPSInfoRepo - ORBListenEndpoints diop://192.168.0.129:12345

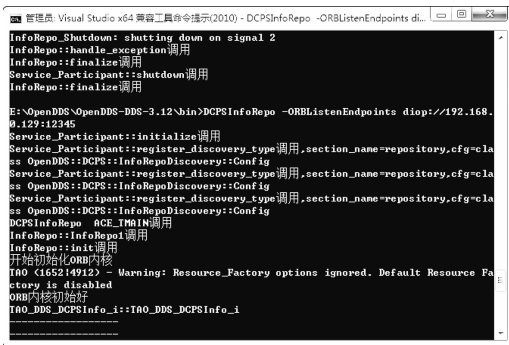


图 8 DCPSInfoRepo 服务器运行示意图

运行发布者程序, 命令行输入:

./subscriber - ORBListenEndpoints diop://192.168.0.1:10000 -DCPSConfigFile dds\_udp\_conf.ini

运行订阅者程序, 命令行输入:

./publisher - ORBListenEndpoints diop://192.168.0.4:10001 -DCPSConfigFile dds\_udp\_conf.ini

发布者发送数据, 订阅者可以顺利接收到数据, 如图 9 和图 10 所示。

实验结果证明, 基于反射内存卡和 OpenDDS 的通信模型能够顺利实现 OpenDDS 发布订阅的过程, 数据可以正确分发。

### 5 结束语

以数据为中心的 OpenDDS 增加了信息分发系统通信的灵活性。反射内存卡作为网络的底层硬件允许系统中所有节点同时读取共享的内存数据, 其无需通信协议, 减少了网络中不必要的数据, 提高了数据的传输效率, 是一种非常适合实时信息网络的底层硬件。基于 OpenDDS 的分层架构, 采用修改 DIOP 协议的方法, 使远程调用数据通过反射内存卡传输, 实现了基于 OpenDDS 和反射内存卡的信息分

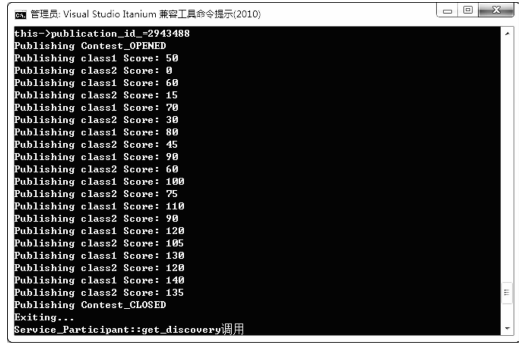


图 9 发布者发布数据

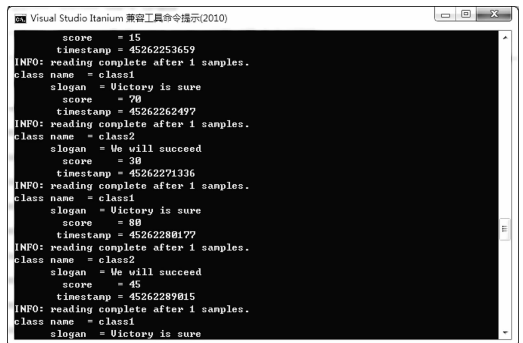


图 10 订阅者接收数据

发系统。将反射内存卡和发布/订阅模式结合设计的信息分发网络, 更好地满足军事领域中信息分发系统实时性、可靠性的要求。

### 参考文献:

- [1] 孙义明, 杨丽萍. 信息化战争中的战术数据链 [M]. 北京: 北京邮电大学出版社, 2005.
- [2] 朱华勇, 张庆杰, 沈林成, 等. 分布式系统实时发布/订阅数据分发技术 [M]. 北京: 国防工业出版社, 2013.
- [3] 任昊利, 等. 数据分发服务—以数据为中心的发布/订阅式通信 [M]. 北京: 清华大学出版社, 2014.
- [4] 冯剑尘, 柳中华. 基于 OpenDDS 的中间件实时通信机制扩展设计 [J]. 舰船电子工程, 2017, 37 (2): 98-101.
- [5] 杜涛涛, 张庆杰, 朱华勇. 面向实时数据分发服务的 DDS 性能研究 [J]. 软件时空, 2010, 26: 155-158.
- [6] 艾晓峰. CEC 中实时信息分发控制技术 [D]. 长沙: 国防科学技术大学, 2007.
- [7] 孙海彬, 吴宇, 魏急波. 基于 CORBA 的可插拔协议研究 [J]. 电脑知识与技术, 2009, 5 (27): 7783-7787.
- [8] 许海涛, 张建伟, 郑灿祥. 基于 CORBA 与 DDS 技术改进 DRS 系统通信 [J]. 计算机工程与设计, 2012, 33 (1): 357-361.
- [9] Micro Focus. CORBA Programmer's Guide: C++ [Z]. UK: Micro Focus, 2017: 475-492.
- [10] Developer's Guide [Z]. Version2. 2a, St. Louis: Object Computing Inc., 1999: 1-588.
- [11] 姚兵, 蔡婷, 李俊林, 等. 基于 DDS 模型的数据分发中间件的设计与实现 [J]. 计算机工程与设计, 2009, 30 (3): 619-623.