

# AgileEAS.NET SOA 中间件/敏捷软件开发平台/魏琼东

博客园 首页 新随笔 联系 订阅 管理

随笔- 137 文章- 0 评论- 2235

分享一个分布式消息总线，基于.NET Socket Tcp的发布-订阅框架，附代码下载

## 一、分布式消息总线

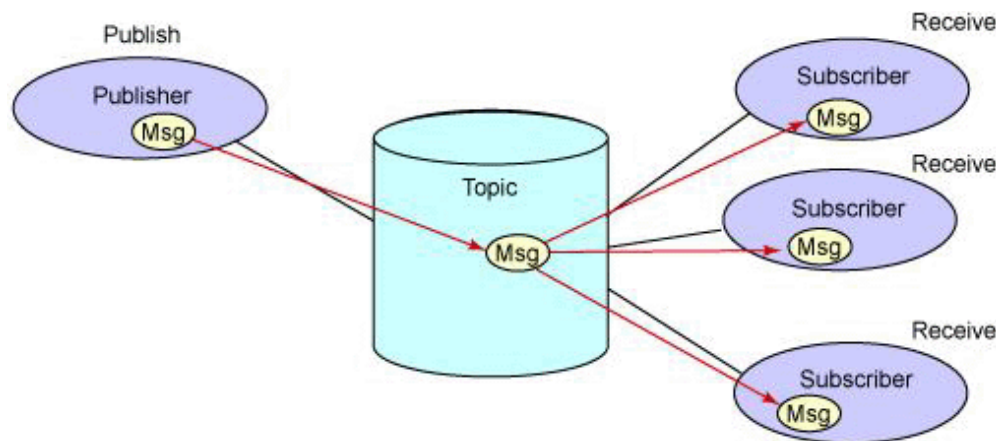
在很多MIS项目之中都有这样的需求，需要一个及时、高效的的通知机制，即比如当使用者A完成了任务X，就需要立即告知使用者B任务X已经完成，在通常的情况下，开发人中都是在使用者B所使用的程序之中写数据库轮循代码，这样就会产品一个很严重的两个问题，第一个问题是延迟，轮循机制要定时执行，必须会引起延迟，第二个问题是数据库压力过大，当进行高频度的轮循会生产大量的数据库查询，并且如果有大量的使用者进行轮循，那数据库的压力就更大了。

那么在这个时间，就需要一套能支持发布-订阅模式的分布式消息总线，那这个问题就可以很好的解决了，比如采用一些成熟的消息总线进行实现，比如MSMQ或者采用比如开源的NServiceBus的发布订阅机制就可以实现处理这种需求，其系统结构就会变成如下所示：

### 公告



魏琼东,系统分析师、系统架构师、高级项目经理、CSAI专业顾问。有13年开发和管理经验，在多家IT企业从事过开发技术与管理工作，为医疗、保险、教育、农业、房地产、电子政务等行业开发软件。有着丰富的理论



本分布式消息总线，目前广泛的被应用于分布式缓存的更新通知，当在N百台客户短在使用缓存的过程之中，某个操作修改了缓存的数据，必须会导致其他终端缓存的失效，那么使用基于Socket的分布式消息总线之后，我们可以做了修改了即可实时通知，做到缓存数据保持最新，再比如医疗应用之中的危急值管理，当发现检验、检查危急值之后，需要及时通知病区启动声光报警系统等，提醒医护工作人员及相关领导做出相应的措施，再比如应用于异构系统整合，当检验系统做出检验报告，通过消息总线进行发布，HIS系统则即时会收到检验报告数据而实现系统的整合。

## 二、基于Socket的实现

目前能够实现发布订阅模式的开源产品非常之多，为什么还要制造轮子呢，其主要原因有以下几点

- 1) 像NServiceBus这种东西基于MSMQ，在大量的发布者-订阅者的情况下性能不佳。
- 2) 此类东西太过于庞大、不易使用和配置。
- 3) 学习成本过高。

那为什么要使用Socket技术进行实现呢，其主要原因是有以下几点:

- 1) 使用高效的Socket通信技术，高效、支持更多的客户端。
- 2) 使用简单，不需要定义太多额外的东西，只需要定义主题和消息即可使用。

知识及资深的行业软件经验，擅长企业软件过程改进、系统分析与架构设计、.NET平台架构技术、SQL Server/ORACLE数据库技术、分布式架构体系及高性能并行计算，尤其对中小软件企业的企业级软件研发管理体系有着深入的研究与应用。



自2004年至今,开发和推广AgileEAS.NET SOA平台，AgileEAS.NET SOA平台是基于敏捷并行开发思想及.Net构件技术而构建的一个快速开发应用平台，用于帮助中小型软件企业建立一个快速适合市场变化的开发团队，以达到节省开发成本、缩短开发时间的目的。

目前主要致力于中小软件企业.NET快速开发技术服务与中小软件企业开发过程管理咨询服务，主要为医疗卫生、铁路、电信、物流、制造等行业提供技术&业务咨询服务，热烈欢迎志同道合的朋友合作，也欢迎大家批评和指正。

在工作和技术之外爱好极为广泛，喜乐音乐、武术、阅读，颇

目前本发布订阅框架是基于AgileEAS.NET SOA中间件平台Socket框架实现的，有关于此Socket框架的技术细节请参考 [AgileEAS.NET SOA 中间件平台.Net Socket通信框架-介绍](#)、[AgileEAS.NET SOA 中间件平台.Net Socket通信框架-简单例子-实现简单的服务端客户端消息应答](#)、[AgileEAS.NET SOA 中间件平台.Net Socket通信框架-完整应用例子-在线聊天室系统-下载配置](#)、[AgileEAS.NET SOA 中间件平台.Net Socket通信框架-完整应用例子-在线聊天室系统-代码解析](#) 文章进行了解和学习。

目前本发布订阅框架并直接集成于AgileEAS.NET SOA Socket通信框架之中并且随其一并发布，下面简单介绍一下其API：

在本框架之中定义了一个消息总线接口IMessageBus：

```
1: using System;
2: using System.Collections.Generic;
3: using System.Linq;
4: using System.Text;
5: using System.Collections;
6:
7: namespace EAS.Messages
8: {
9:     /// <summary>
10:     /// 消息总线接口定义。
11:     /// </summary>
12:     public interface IMessageBus : IDisposable
13:     {
14:         /// <summary>
15:         /// 注册发布者。
16:         /// </summary>
17:         /// <param name="publisher">发布者。</param>
18:         void AddPublisher(string publisher);
19:
20:         /// <summary>
21:         /// 注册发布者。
22:         /// </summary>
23:         /// <param name="publisher">发布者。</param>
24:         /// <param name="topic">主题。</param>
25:         void AddPublisher(string publisher, string topic);
26:
```

为喜欢传统的人文、史哲、中医、管理，平时喜欢听着轻悠的音乐翻翻相关的书籍，也习惯于思考，在技术和管理中也希望能把自己脑袋中杂乱的东西融合起来，个人目标期望能在中化传统文化的框架之下能中小软件企业的开发管理体系能有一点做为，在生活方面极其随意，日子过的心安理得就行了。

本博客文章版权归作者和博客园共有，欢迎转载，但未经作者同意必须保留此段声明，且在文章页面明显位置给出原文连接，否则保留追究法律责任的权利。

快速订阅



昵称：魏琼东

园龄：8年7个月

荣誉：推荐博客

粉丝：640

关注：33

```
27:    /// <summary>
28:    /// 发布一条消息到总线。
29:    /// </summary>
30:    /// <param name="topic">主题。</param>
31:    /// <param name="message">发布的消息。</param>
32:    void Publish(string topic, object message);
33:
34:    /// <summary>
35:    /// 订阅消息。
36:    /// </summary>
37:    /// <param name="subscriber">订阅者。</param>
38:    /// <param name="topic">主题。</param>
39:    /// <param name="notifyHandler">订阅通知。</param>
40:    void Subscribe(object subscriber, string topic, MessageNotifyHandler notifyHandler);
41:
42:    /// <summary>
43:    /// 订阅消息。
44:    /// </summary>
45:    /// <param name="subscriber">订阅者。</param>
46:    /// <param name="friendName">订阅者名称，用于处理离线订阅。</param>
47:    /// <param name="topic">主题。</param>
48:    /// <param name="notifyHandler">订阅通知。</param>
49:    void Subscribe(object subscriber, string friendName, string topic, MessageNotifyHandler notifyHandler);
50:
51:    /// <summary>
52:    /// 订阅消息。
53:    /// </summary>
54:    /// <param name="subscriber">订阅者。</param>
55:    /// <param name="friendName">订阅者名称，用于处理离线订阅。</param>
56:    /// <param name="topic">主题。</param>
57:    /// <param name="notifyHandler">订阅通知。</param>
58:    /// <param name="changedHandler">发布者状态变化委托。</param>
59:    void Subscribe(object subscriber, string friendName, string topic, MessageNotifyHandler
notifyHandler, PublisherSstatusChangedHandler changedHandler);
60:
61:    /// <summary>
62:    /// 退订消息。
63:    /// </summary>
64:    /// <param name="subscriber">订阅者。</param>
```

+加关注

2015年6月						
<	日	一	二	三	四	五
	31	1	2	3	4	5
	7	8	9	10	11	12
	14	15	16	17	18	19
	21	22	23	24	25	26
	28	29	30	1	2	3
	5	6	7	8	9	10

## 搜索

## 常用链接

- 我的随笔
- 我的评论
- 我的参与
- 最新评论
- 我的标签
- 更多链接

## 最新随笔

```

65:     void Unsubscribe(object subscriber);
66:
67:     /// <summary>
68:     /// 退订消息。
69:     /// </summary>
70:     /// <param name="subscriber">订阅者。</param>
71:     /// <param name="topic">主题。</param>
72:     void Unsubscribe(object subscriber, string topic);
73:
74:     /// <summary>
75:     /// 退订消息。
76:     /// </summary>
77:     /// <param name="subscriber">订阅者。</param>

```

IMessageBus就基于Socket发布订阅消息总线的灵魂接口，也是基唯一的发布者调用者功能入口，也就是说不管你是发布者还是订阅者都需要调用这个接口，如果你是发布者请调用IMessageBus接口的Publish方法向消息总线发布消息，如果是你订阅者请通过IMessageBus的订阅方法进行订阅，当你订阅了某个主题之后，有发布者发布该主题的消息，你即可以收到消息并调用订阅回调函数进行处理。

## 三、实现一个简单的例子

现在我们开始一个简单的应用消息总线的例子，本例子代码解决方案由下图4个项目组成：

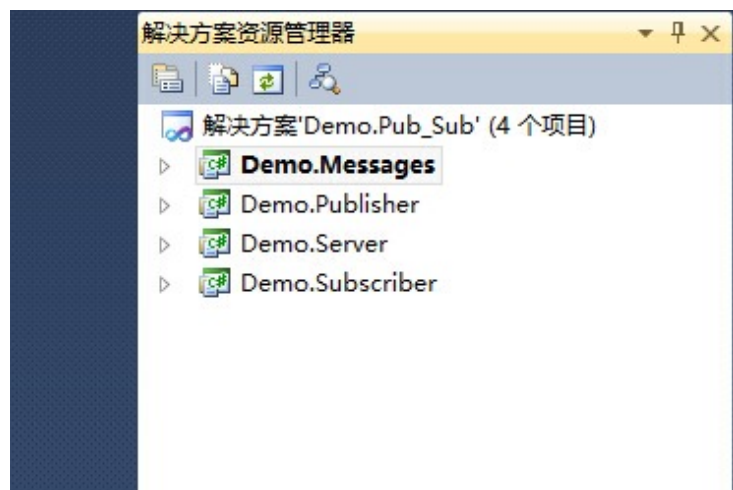
1. AgileEAS.NET SOA中间件...
2. 分布式消息总线，基于.NET...
3. 分享一个分布式消息总线，...
4. 3小时搞定一个简单的MIS...
5. WinForm/MIS项目开发之中...
6. MIS系统开发利器，实施、...
7. 使用Lua脚本语言开发出高...
8. 高并发应用场景下的负载均...
9. MIS系统开发利器，快速的...
10. 犀利的报表系统，发票据...

## 我的标签

中间件 (89) 平台 (81)  
.NET (75) EAS (61) 插件 (52)  
模块 (46) 架构 (44) ORM (41)  
分布式 (24) SOA (24) 更多

## 随笔分类

AgileEAS.NET(108)  
C#语言(26)  
dotNet相关(33)  
WF工作流(3)  
编程技术(58)  
程序人生(5)  
创业交流(9)  
架构设计(107)  
脚本引擎(1)  
企业管理(7)



其中：Demo.Messages项目定义发布者、订阅者所使用的消息对象和消息主题。

Demo.Publisher项目为发布者代码。

Demo.Subscriber项目为订阅者代码。

Demo.Server项目为服务端代码。

在Demo.Messages项目之中，我们定义了一个消息Message：

```
1: using System;
2: using System.Collections.Generic;
3: using System.Linq;
4: using System.Text;
5: using System.Xml.Serialization;
6:
7: namespace Demo.Messages
8: {
9:     [Serializable]
10:     public class Message
11:     {
12:         [XmlElement]
```

软件工程(103)

设计模式(52)

心情感悟(11)

## 随笔档案

2015年4月 (1)

2014年8月 (2)

2014年4月 (1)

2014年3月 (1)

2014年1月 (3)

2013年12月 (4)

2013年11月 (5)

2013年10月 (7)

2013年4月 (2)

2012年2月 (2)

2011年11月 (2)

2011年8月 (1)

2011年7月 (5)

2011年6月 (3)

2011年5月 (10)

2011年1月 (1)

2010年11月 (5)

2010年10月 (17)

2010年9月 (24)

2010年8月 (6)

2010年7月 (8)

2010年6月 (7)

2010年4月 (12)

2008年6月 (2)

```
13:     public Guid ID
14:     {
15:         get;
16:         set;
17:     }
18: }
19: }
```

消息Message很简单，只有一个属性ID，同时 还需要定义一个消息主题：

```
1: using System;
2: using System.Collections.Generic;
3: using System.Linq;
4: using System.Text;
5:
6: namespace Demo.Messages
7: {
8:     public class Topics
9:     {
10:         public static readonly string DEMO_TOPIC = "演示消息";
11:     }
12: }
```

我们定义了一个消息主题为“演示消息”。

在Demo.Publisher项目之中，没有太多额外的代码，只有在Program.cs写了以下简单的调用代码：

```
1: using System;
2: using System.Collections.Generic;
3: using System.Linq;
4: using System.Text;
5: using EAS.Messages;
6:
7: namespace Demo.Publisher
8: {
9:     class Program
10:     {
```

2008年1月 (1)  
2007年6月 (1)  
2006年11月 (4)

## 文章分类

AgileEAS.NET

## 相册

AgileEAS.NET(1)  
agilelab.cn(6)  
rss(8)  
贴图相册(14)

## 博客链接

计文忻的博客  
王珩的博客  
我在CSAI的博客  
我在CSDN的博客  
殷长波的博客  
张阳的博客

## 产品站点

AgileEAS.NET平台

## 朋友博客

CallHot 's Blog  
王涛的博客

## 企业站点



```

11:     static void Main(string[] args)
12:     {
13:         var container = EAS.Objects.ContainerBuilder.BuilderDefault();
14:         var bus = container.GetComponentInstance("MessageBus") as IMessageBus;
15:         System.Console.WriteLine("Publisher");
16:
17:         while (System.Console.ReadLine()!="exit")
18:         {
19:             var m = new Messages.Message { ID = Guid.NewGuid() };
20:             bus.Publish(Demo.Messages.Topics.DEMO_TOPIC, m);
21:             System.Console.WriteLine(string.Format("Publish:{0}", m.ID));
22:         }
23:     }

```

从IOC容器获取一个消息总线IMessageBus对象，并调用Publish函数发布消息“bus.Publish(Demo.Messages.Topics.DEMO\_TOPIC, m);”。

当然了，使用了IOC容器，就离来开配置文件了，其App.config文件内容如下：

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <configuration>
3:   <configSections>
4:     <section name="eas" type="EAS.ConfigHandler,EAS.MicroKernel" />
5:   </configSections>
6:   <eas>
7:     <objects>
8:       <!--消息总线-->
9:       <object name="MessageBus" assembly="EAS.MicroKernel" type="EAS.Sockets.Bus.SocketBus" LifestyleType="Singleton">
10:         <property name="Url" type="string" value="socket.tcp://127.0.0.1:6606"/>
11:       </object>
12:     </objects>
13:   </eas>

```

在Demo.Subscriber项目之中，使用与Demo.Publisher一模一样的配置文件，其Program.cs代码如下：

```

1: using System;
2: using System.Collections.Generic;
3: using System.Linq;

```

敏捷软件工程实验室

## 最新评论

1. Re:AgileEAS.NET SOA 中...  
mark

--xuliyuan173

2. Re:AgileEAS.NET SOA中...  
我来顶一个

--贺臣

3. Re:AgileEAS.NET SOA中...

@zengfanlin引用最近在看DDD方面的书籍，一头雾水，不知道有啥学习方法么不要光看，去做，尝试做一点，即使是抄某些人的，也得去做...

--魏琼东

4. Re:AgileEAS.NET SOA中...

最近在看DDD方面的书籍，一头雾水，不知道有啥学习方法么

--zengfanlin

5. Re:AgileEAS.NET SOA中...

@薛小坏引用接触这个平台也有些年头了，突然发现 目前能做的也只有点个赞啊 点个推荐啥的，真是汗颜谢谢，这个确实是有项目，有需要才能接触上。...

--魏琼东

## 阅读排行榜

1. .net执行性能，Winform性...
2. 使用DotNetBar制作漂亮的...
3. 3小时搞定一个简单的MIS...



```
4: using System.Text;
5: using EAS.Messages;
6:
7: namespace Demo.Subscriber
8: {
9:     class Program
10:    {
11:        static void Main(string[] args)
12:        {
13:            var container = EAS.Objects.ContainerBuilder.BuilderDefault();
14:            var bus = container.GetComponentInstance("MessageBus") as IMessageBus;
15:            System.Console.WriteLine("Subscriber");
16:
```

其中代码**bus.Subscribe(new Program(), "Subscriber1", Demo.Messages.Topics.DEMO\_TOPIC, MessageNotify);**：完成把消息订阅到MessageNotify通知函数之中。

在Demo.Server项目之中，启动服务器并且开始接收订阅和发布：

```
1: using System;
2: using System.Collections.Generic;
3: using System.Linq;
4: using System.Text;
5: using EAS.Sockets;
6:
7: namespace Demo.Server
8: {
9:     class Program
10:    {
11:        static void Main(string[] args)
12:        {
13:            SocketServer socketServer = new SocketServer(128);
14:            socketServer.Port = 6606;
15:            socketServer.Logger = new EAS.Loggers.ConsoleLogger();
16:            socketServer.Initialize();
17:            System.Console.WriteLine("Server Starting...");
18:            socketServer.StartServer();
19:            System.Console.WriteLine("Server Startup!");
```

4. DotNET企业架构应用实践-...
5. 我的学习工作经历，一个园...

## 评论排行榜

1. 流氓话题-从QQ与360之的...
2. 分享一个分布式消息总线，...
3. 分布式消息总线，基于.NET...
4. 我的学习工作经历，一个园...
5. 3小时搞定一个简单的MIS...

## 推荐排行榜

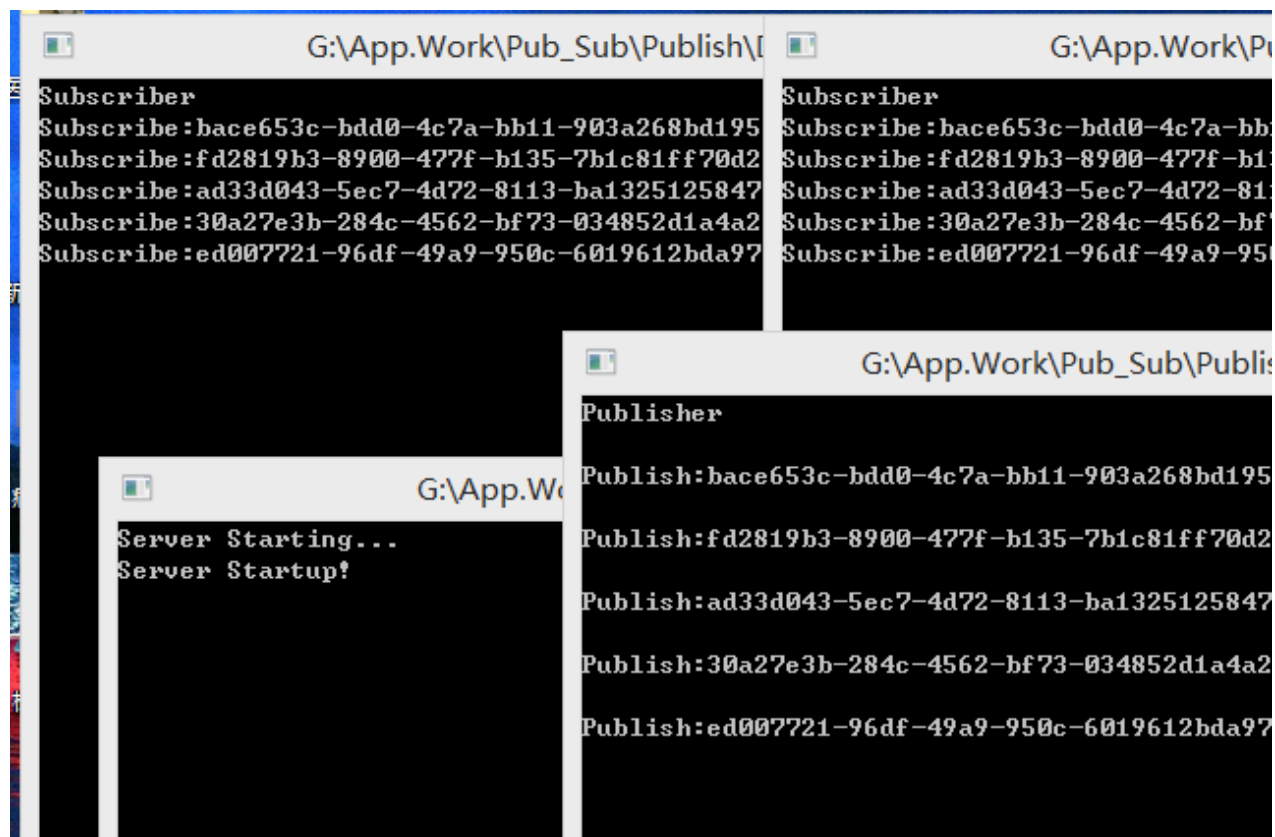
1. 3小时搞定一个简单的MIS...
2. 分布式消息总线，基于.NET...
3. 分享一个分布式消息总线，...
4. 高并发应用场景下的负载均...
5. AgileEAS.NET SOA 中间件...

```
20:     System.Console.ReadLine();  
21: }  
22: }  
23: }
```

到此为止，所有代码均已完成，是不是很简单，接下来，我们跑起来验证一下效果。

## 四、验证效果

我们在编译输入目录Publish下先启动Demo.Server.exe，再启动两个Demo.Subscriber.exe,再启动一个Demo.Publisher.exe，在Demo.Publisher.exe控制台按回车键：



OK，搞定。

## 五、源代码下载

本程序的源代码已上传到服务器，请通过 [http://42.121.30.77/downloads/eas/Demo.Pub\\_Sub.rar](http://42.121.30.77/downloads/eas/Demo.Pub_Sub.rar) 进行下载，如果在开发过程之中想要了解更多有关Socket通信框架以及更多AgileEAS.NET SOA中间件平台的技术资源，请通过AgileEAS.NET SOA 网站：<http://www.smart eas.net> 的 [最新下载栏目](#) 进行下载。

## 六、问题反馈

麻烦大家在通过视频进行学习的时候能及时把问题反馈给楼主，或者有什么需要改进的一些建议都请向楼主直接反馈,以下是联系方式：

AgileEAS.NET SOA 网站：<http://www.smart eas.net>

官方博客: <http://eastjade.cnblogs.com>

楼主QQ:47920381,AgileEAS.NET

QQ群：113723486（AgileEAS SOA 平台）/上限1000人

199463175（AgileEAS SOA 交流）/上限1000人

120661978（AgileEAS.NET 平台交流）/上限1000人

邮件:james@agilelab.cn,mail.james@qq.com,

电话：18629261335。

作者：魏琼东

出处：<http://www.cnblogs.com/eastjade>

关于作者：有13年的软件从业经历，专注于中小软件企业软件开发过程研究，通过在技术与管理帮助中小软件企业实现技术层面开源节流的目的。熟悉需求分析、企业架构、项目管理。现主要从事基于 AgileEAS.NET 平台的技术咨询工作，主要服务于医疗卫生、铁路、电信、物流、物联网、制造、零售等行业。如有问题或建议，请多多赐教！

本文版权归作者和博客园共有，欢迎转载，但未经作者同意必须保留此段声明，且在文章页面明显位置给出原文连接，如有问题，可以通过 [mail.james@qq.com](mailto:mail.james@qq.com) 联系我，也可以加入QQ群:113723486、199463175、116773358、116773358、212867943、147168308、

59827496、193486983、15118502和大家共同讨论，非常感谢。

分类: [架构设计](#), [C#语言](#), [dotNet相关](#)

标签: [中间件](#), [平台](#), [分布式](#), [ESB](#), [Socket](#), [消息](#), [tcp](#)

绿色通道：[好文要顶](#) [关注我](#) [收藏该文](#) [与我联系](#) 



魏琼东  
关注 - 33  
粉丝 - 640

450

荣誉：[推荐博客](#)  
[+加关注](#)

(请您对文章做出评价)

« 上一篇：[3小时搞定一个简单的MIS系统案例Northwind，有视频、有源代码下载、有真相](#)  
» 下一篇：[分布式消息总线，基于.NET Socket Tcp的发布-订阅框架之离线支持，附代码下载](#)

posted @ 2014-08-14 08:55 魏琼东 阅读(5079) 评论(96) 编辑 收藏

[< Prev](#) [1](#) [2](#)

评论列表

#51楼 2014-08-14 20:57 MrLeung  
mark，正好要用到  
支持(0) 反对(0)

#52楼 [\[楼主\]](#) 2014-08-14 20:57 魏琼东  
@MrLeung

[引用](#)

mark，正好要用到

欢迎使用

支持(0) 反对(0)

#53楼 2014-08-14 21:51 aXinNo1

不錯，正要用。

支持(0) 反对(0)

#54楼 [\[楼主\]](#) 2014-08-14 21:52 魏琼东

@aXinNo1

[引用](#)

不錯，正要用。

欢迎使用，能给大家带来好处，是我乐意的

支持(0) 反对(0)

#55楼 2014-08-14 23:00 秋叶残念

mark,学习学习

支持(0) 反对(0)

#56楼 2014-08-15 08:13 遗忘海岸

每个程序员都搞自己的框架

支持(0) 反对(0)

#57楼 2014-08-15 08:46 love begins coding  
用UDP是不是快点

支持(0) 反对(0)

#58楼 2014-08-15 08:51 PBER  
收藏

支持(0) 反对(0)

#59楼 [楼主] 2014-08-15 09:05 魏琼东  
@秋叶残念  
引用  
mark,学习学习

多谢支持

支持(0) 反对(0)

#60楼 [楼主] 2014-08-15 09:05 魏琼东  
@遗忘海岸  
引用  
每个程序员都搞自己的框架



造太多轮子也没有太多必要，但是优秀的东西值的制造

支持(0) 反对(0)

#61楼 [楼主] 2014-08-15 09:06 魏琼东

@love begins coding

引用

用UDP是不是快点

UDP更快，但要处理丢包的问题

支持(0) 反对(0)

#62楼 [楼主] 2014-08-15 09:06 魏琼东

@PBER

引用

收藏

谢谢支持

支持(0) 反对(0)

#63楼 2014-08-15 09:06 Tiny Mang

mark,学习学习

支持(0) 反对(0)

#64楼 [楼主] 2014-08-15 14:28 魏琼东  
-----  
@Tiny Mang  
引用  
mark,学习学习

谢谢支持

支持(0) 反对(0)

---

#65楼 2014-08-15 18:09 aXinNo1  
-----  
@魏琼东  
运行出来报错了，缺少EAS的引用，请问，那个dll在哪里？

支持(0) 反对(0)

---

#66楼 [楼主] 2014-08-15 19:29 魏琼东  
-----  
@aXinNo1  
引用  
@魏琼东  
运行出来报错了，缺少EAS的引用，请问，那个dll在哪里？

应该不会报错，如果缺少到[http://www.smart eas.net/page/EAS NET Downloads.aspx](http://www.smart eas.net/page/EAS_NET_Downloads.aspx)下载

支持(0) 反对(0)

---

#67楼 2014-08-16 22:28 zengfanlin  
-----  
@魏琼东

请问一下我用Entity Framework 来代替您这个ORM可不可以呢

支持(0) 反对(0)

#68楼 [楼主] 2014-08-17 12:41 魏琼东

@zengfanlin

引用

@魏琼东

请问一下我用Entity Framework 来代替您这个ORM可不可以呢

可以的，没有问题

支持(0) 反对(0)

#69楼 2014-08-17 12:44 zengfanlin

@魏琼东

那我觉得您以后也可以转到Entity Framework上来，不用单独开发维护一个ORM是不是更方便，而且还通用

支持(0) 反对(0)

#70楼 [楼主] 2014-08-17 12:47 魏琼东

@zengfanlin

引用

@魏琼东

那我觉得您以后也可以转到Entity Framework上来，不用单独开发维护一个ORM是不是更方便，而且还通用

这个不会，因为我们的ORM有其独特的一方面，比如式分布式Linq的支持。再一个，我们及客户都使用在非常广泛，也是和Linq集成的，学习成本代，并且陪套的领域模型设计器要比EF陪套的强太多，所以我们不会转到EF，但是你们可以自由选择

支持(0) 反对(0)

#71楼 2014-08-17 12:56 zengfanlin

@魏琼东

好的谢谢，主要是我觉得EF的设计器更方便，而且我用您的设计器改数据库设计然后覆盖代码的时候，有时候总是改不过来，也不知道是不是使用方法不对

支持(0) 反对(0)

#72楼 [楼主] 2014-08-17 13:00 魏琼东

@zengfanlin

引用

@魏琼东

好的谢谢，主要是我觉得EF的设计器更方便，而且我用您的设计器改数据库设计然后覆盖代码的时候，有时候总是改不过来，也不知道是不是使用方法不对

我们倡导的思路是，设计和模型优先，并不是数据库优先，当然，还有一些小Bug和未完善的地方，正在努力完善之中，覆盖代码是采用分部类的实现，也就是说，你自己的写的代码写对了位置，设计器生成 是不会覆盖的

支持(0) 反对(0)

#73楼 2014-08-17 13:02 zengfanlin

@魏琼东  
不是不是，这个我还是清楚的，我要生成覆盖的是Generat文件夹里面的代码。现在我的解决方法是生成到其他文件夹然后手动拷贝覆盖，这样就没有问题了

支持(0) 反对(0)

#74楼 [楼主] 2014-08-17 13:03 魏琼东

@zengfanlin

引用

@魏琼东  
不是不是，这个我还是清楚的，我要生成覆盖的是Generat文件夹里面的代码。现在我的解决方法是生成到其他文件夹然后手动拷贝覆盖，这样就没有问题了

要把输出路径设置正确就没有问题，因为其生成的文件夹有限定

支持(0) 反对(0)

#75楼 2014-08-17 13:05 zengfanlin

@魏琼东

好的，谢谢您解答，我再去检查一下生成参数

支持(0) 反对(0)

#76楼 2014-08-18 09:56 麦克刘.Mr

订阅者离线消息存储机制 什么时候写啊？

支持(0) 反对(0)

#77楼 [楼主] 2014-08-18 09:56 魏琼东

@麦克刘.Mr

引用

订阅者离线消息存储机制 什么时候写啊？

本周之内吧，星期二或者星期三应该就能上来

支持(0) 反对(0)

#78楼 2014-08-18 10:36 cpypzl

mark，研究研究。

支持(0) 反对(0)

#79楼 2014-08-19 10:03 Indream Luo

在我的系统中，一开始也写了个类似的中间件，主要是基于WCF，就是为了实现强类型和模拟直接调用的效果，降低客户端成本。不过本身我们的系统是业务驱动型的，没有太大的性能瓶颈。

支持(0) 反对(0)

#80楼 [楼主] 2014-08-19 14:19 魏琼东

@Indream Luo

引用

在我的系统中，一开始也写了个类似的中间件，主要是基于WCF，就是为了实现强类型和模拟直接调用的效果，降低客户端成本。不过本身我们的系统是业务驱动型的，没有太大的性能瓶颈。

WCF更容易上手阿

支持(0) 反对(0)

#81楼 2014-08-19 14:32 Indream Luo

@魏琼东

引用

@Indream Luo

引用

引用在我的系统中，一开始也写了个类似的中间件，主要是基于WCF，就是为了实现强类型和模拟直接调用的效果，降低客户端成本。不过本身我们的系统是业务驱动型的，没有太大的性能瓶颈。

WCF更容易上手阿

是的，另外用的一个原因是因为封装很重要。我希望使用调用中间件的用户只要var client = new Client(); client.Sent(message);就解决问题，开发效率可以提高。

另外我把Client挂在了Nuget上，如果要降级或者优化，只要统一Update就可以了。

支持(0) 反对(0)

#82楼 [楼主] 2014-08-19 16:01 魏琼东

@Indream Luo

这个的确要方便

支持(0) 反对(0)

#83楼 2014-08-21 17:02 ljs0109



这个东西很有必要

支持(0) 反对(0)

#84楼 [楼主] 2014-08-21 17:03 魏琼东

@ljs0109

引用

这个东西很有必要

希望能给你们带来用处

支持(0) 反对(0)

#85楼 2014-08-21 22:41 麻将我会

不错不错

支持(0) 反对(0)

#86楼 [楼主] 2014-08-22 09:31 魏琼东

@麻将我会

引用

不错不错

感谢支持

支持(0) 反对(0)

#87楼 2014-08-28 14:56 kimi\_gyj  
-----  
@魏琼东  
@aXinNo1  
编译时报错是因为类库使用默认使用了  
.NET Framework 4 client profile  
改成  
.NET Framework 4 就能运行通过  
  
支持(0) 反对(0)

---

#88楼 [楼主] 2014-08-28 14:57 魏琼东  
-----  
@kimi\_gyj  
引用  
@魏琼东  
@aXinNo1  
编译时报错是因为类库使用默认使用了  
.NET Framework 4 client profile  
改成  
.NET Framework 4 就能运行通过  
  
是这样的  
  
支持(0) 反对(0)

---

#89楼 2014-09-01 18:47 zengfanlin  
-----  
这个可不可以把server端和publish端做到一个程序里面，如果可以，需要多线程处理么？  
支持(0) 反对(0)

#90楼 2014-09-01 19:01 zengfanlin  
-----  
我试了一下可以  
支持(0) 反对(0)

#91楼 [楼主] 2014-09-01 19:09 魏琼东  
-----  
@zengfanlin  
引用  
这个可不可以把server端和publish端做到一个程序里面，如果可以，需要多线程处理么？  
  
是可以的。  
支持(0) 反对(0)

#92楼 [楼主] 2014-09-01 19:09 魏琼东  
-----  
@zengfanlin  
引用  
我试了一下可以  
  
不错  
支持(0) 反对(0)

#93楼

2014-12-04 09:23 jhkmnm

-----

同一个主题有多个订阅者，能实现向一个特定的订单者发送消息吗

支持(0) 反对(0)

#94楼

[楼主] 2014-12-04 09:32 魏琼东

-----

@jhkmnm

引用

同一个主题有多个订阅者，能实现向一个特定的订单者发送消息吗

多个订阅者一定是向所有订阅者发送了，向一个特定订阅者发送，他一个订阅就可以

支持(0) 反对(0)

#95楼

2014-12-09 17:48 jhkmnm

-----

同一个客户端，可以多次注册同一个主题，然后会收到多份消息

支持(0) 反对(0)

#96楼

2014-12-12 10:34 ljs0109

-----

很不错，赞一个

支持(0) 反对(0)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库

【推荐】融云即时通讯云 - 专注为 App 开发者提供IM云服务

【推荐】如何让你的程序拥有象Excel一样强大的数据编辑功能

【活动】RDS邀您6.5折体验PostgreSQL



最新IT新闻:

- Airbnb发布开源的机器学习软件包Aerosolve
  - 微软的许多流行应用不再支持Facebook
  - 全栈工程师？给把瑞士军刀你去砍鬼子好不好！？
  - 今天的科技将如何塑造明天的面试过程
  - 为什么苹果当初没让应用直接跑在Apple Watch上？
- » 更多新闻...



最新知识库文章:

- 领域驱动设计阅读思考
- 程序员不是砌砖工人，他们是作家
- 技术债务偿还计划
- 那些令人喷饭的代码注释：仅以此代码献给...
- 给代码多留一些空间
- » 更多知识库文章...

---

Copyright ©2015 魏琼东