



Available Online through

[www.ijptonline.com](http://www.ijptonline.com)

## INTER COMMUNICATION PROTOCOL FOR TWO DIFFERENT TOWERS IN MOBILE COMMUNICATION

**K.Dileep Kumar, Rajagopal.C**

UG Scholar, Department of Computer Science & Engineering, Saveetha School of Engineering, Saveetha University  
Chennai, India.

Assistant Professor (SG), Information Technology, Saveetha School of Engineering, Saveetha University.  
Chennai, India.

Email: [dileepkumar.kudum@gmail.com](mailto:dileepkumar.kudum@gmail.com)

Received on: 10.08.2016

Accepted on: 06.09.2016

### Abstract:

To provide a protocol model using inter communication between two different towers using mobile communication. Different locations under two or more different towers we can transfer data from server. We can also provide this inter communication protocol through custom protocol or protocol bean in mobile communication. As technology is developing so fast, so we need to develop inter communication in mobile communications. A protocol that supports the sharing of resources that exist in different packet switching networks is presented. The protocol provides for variation in individual network packet sizes, transmission failures, sequencing, flow control, end-to-end error checking, and the creation and destruction of logical process-to-process connections.

### 1. Introduction

The spreading of data through the Inter communication between towers has grown tremendously. This trend is also likely to be intensified by the increasing popularization of applications that use inter communication protocol in areas such as entertainment, education, culture and so on.

A data delivery network is one of the main technologies enabling the consumption of towers constantly while shifting from one place to another. Area is analysed as the hexagonal sectors to implement towers for transferring signals. So that it can properly analyse the required sector's tower signal which helps all to communicate without any disturbances.

The disturbance or cross talks is the main concept to be discussed in mobile communication. So that to avoid them to provide tower in particular area that area is analysed as a hexagonal structure. That particular hexagonal structure is considered as a cell. That cell is splitted into different sectors using cell splitting concept. Each sector is provided

with different antennas to avoid cross talks. Communication between people at different locations under different towers is called as inter communication protocol. To have better communication we need to provide a separate base station for each cluster. This cluster contains 7 sectors and that sectors must be provided with different frequencies to avoid cross talks.

## **2. Related Work**

### **2.1 Object-oriented communication:**

It presents an activity model about MMOG's improvement. In this model, an activity is an essential system correspondence unit[3] presents a system middleware for MMOG's advancement, which utilizes objects as fundamental units of conveyance and replication of state. Clearly, question arranged correspondence can better coordinate these innovations and improve comprehensibility and reusability altogether. In MMOG' advancement, transmission control convention (TCP) is connected usually as transport layer convention. In any case, TCP ordinarily bargains straightforwardly with twofold stream. Instructions to get object example from parallel stream and how to lessen the. many-sided quality of improvement have turned into a vital work. For article situated correspondence, a key innovation is serialization innovation. This innovation can encode an article case as twofold stream, and can likewise interpret as an item example from double stream. In straightforward terms, serialization innovation is an encoding arrangement, which implies that commonly every dialect has its own particular serialization design. For instance, Java dialect bolsters object serialization [4] and ActionScript dialect bolster AMF group [5]. In any case, there are a few issues in these innovations.

Analysis from the execution point of view. On one hand, since every dialect has own serialization position, communicating over various dialects requires particular change, which would bring about more multifaceted nature. Then again, these serialized information has more repetitive data, bringing about creating more excess information and possessing bigger transfer speed. Analysis from the epitome viewpoint. Normally in item arranged correspondence, transmitted article is an information model, as well as a plan of action. For a customer, a complete correspondence more often than not contains two procedures: sending and accepting. It is extremely conceivable that sent business information is not the same as got business information. On the off chance that we consider transmitted article as an information model, there are two exemplification ways. Way one, embody two model in a solitary item. Way two, exemplify a model in an article individually. The previous results in transmitting excess information and the last can prompt the expanding many-sided quality of programming structure. Agreeing the way two, could

characterize two classes as code.2 b. In spite of the fact that don't bring about transmitting repetitive business information, it will prompt firmly related business information isolated. For a minimized business rationale, it might bring about the model turns out to be more intricate, which is not helpful for the epitome of programming, likewise prompt more class definitions that make venture administration and upkeep more perplexing.

## **2.2 Google protocol buffer:**

Convention Buffer [6] is an outsider serialization innovation. In no time it bolsters Java, C++ and Python dialects. Contrasted and local Java serialization execution, Protocol Buffers can secure littler serialization size and speedier handling speed. For instance, on the off chance that we serialize an item case in code.1, the length of information is just 26 bytes. Contrast and Java serialization, evidently it has fundamentally diminished transmission capacity. In any case, when we utilize Protocol Buffers, it needs to order a transitional convention record to create a class definition. For instance, subsequent to incorporating convention document, it will get a class definition like code 3. This strategy limits programming epitome; particularly it can confine the legacy pecking order with the goal that designers can't extend their business objects through legacy.

## **2.3 Custom protocol:**

At the point when build up a continuous framework, we can utilize some modern conventions, for example, constant informing convention (RTMP). These advanced conventions are intended to determine unique information transmission. For instance, RTMP is particularly for the transmission of sight and sound information. Clearly, usage of these conventions need to make additional parcel trade, which will add additional data transfer capacity overhead. Be that as it may, huge multiplayer web recreations, whose components are of high correspondence recurrence and little measure of information in single correspondence, have evident extra transfer speed overhead. Along these lines, the ideal methodology is to utilize custom convention as per the genuine circumstance of task.

It actualizes a code-era motor for encode/disentangle custom convention. In this paper, we plan another method for custom convention which is more adaptable, has more meaningfulness and littler data transfer capacity. Also, by module advancement it can resolve convention administration

## **3. Protocol-Bean definition**

In programming improvement, object-situated is an extremely phenomenal thought of secluded configuration. The most essential part in this system is custom convention plan. Keeping in mind the endgoal to comprehend how to tweak object situated convention, this structure proposes the idea of convention bean. A convention bean is a class

definition as indicated by the determination of JavaBean. Through metadata mapping setup, a convention bean is mapped as a paired stream. In this manner utilizing reflection component as a part of dialect, a parallel stream is amassed into an item occasion. For a complete system correspondence business, it more often than not contains two correspondence conventions: sending and getting. In this structure, a correspondence procedure is spoken to by a convention bean. Every convention bean is an information bearer, whose some portion of fields is utilized to store sending information, and another piece of fields is utilized to store accepting information.

```
public class Person{  
    [Segment(Codec.Side_both)]  
    public int idcard{get;set;}  
    [Segment(Codec.Side_both)]  
    public String name{get;set} }  
}
```

As code demonstrating, each arrangement of convention bean requires "ProtocolMap" quality that contains three parameters.

- 1) Protocol ID: one of a kind identifier used to recognize the convention.
- 2) Plugin sort: this structure depends on module advancement. So every convention mapping will be connected with a module sort which is utilized to locate the coveted administration module.
- 3) Notice sort: when the module has completed business preparing and needs to return information, this structure will have an occasion framework; subsequently it will circulate an occasion to a processor which is connected with indicated notice sort.

Seen from the above code scrap, there is an essential property 'Section'. Every "Portion" speaks to a section of the information parcel; it has a parameter "Side" which is utilized to recognize the information can be utilized for customer or server. There are three alternatives:

- 1) Codec.Side\_both: demonstrate this field is utilized to get information in customer and send information to server in the meantime.
- 2) Codec.Side\_client: demonstrate this field is just used to get information in customer.
- 3) Codec.Side\_server:indicate: this documented is just used to send information to server.

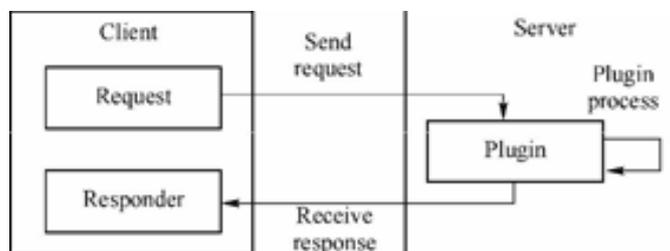
For the server we additionally need to characterize such a convention. At the point when application is initially begun, the system will discover these definitions. By bean definition, it will consequently create convention codec

which is connected with convention ID. The codec is in charge of disentangling and encoding of twofold stream.

Through this usage, an item case can be mapped into a double stream by evading transmit excess business data, (for example, field name) and lessening the information transmission capacity. By parameter 'Side', which permits the designer just uses the piece of business fields for information transmission, which can either typify firmly related business information in a same module, or guarantee the information transmission execution. In the mean time, the utilization of metadata can dodge nosy code plan.

### 3.2 Plugin development

Module improvement is not another idea, and a considerable measure of innovations depend on this thought. In this engineering plan, the greatest point of preference is the adaptable versatility by adding and uprooting module modules to accomplish business capacities changes. In the mean time, the module highlight permits dynamic arrangement abilities. Comprehensively talking, the module is a measured idea. In the solid usage, it might be a class definition, and it additionally might be made out of a gathering of class definitions. The Bundle in OSGi [10] and the Bean in Spring [11] system are only indication of this measured considering. This structure depends on Spring holder which implies through enrolling business module in Spring compartment module usefulness can be accomplished. The accompanying definite depiction demonstrates to create business by module implies.



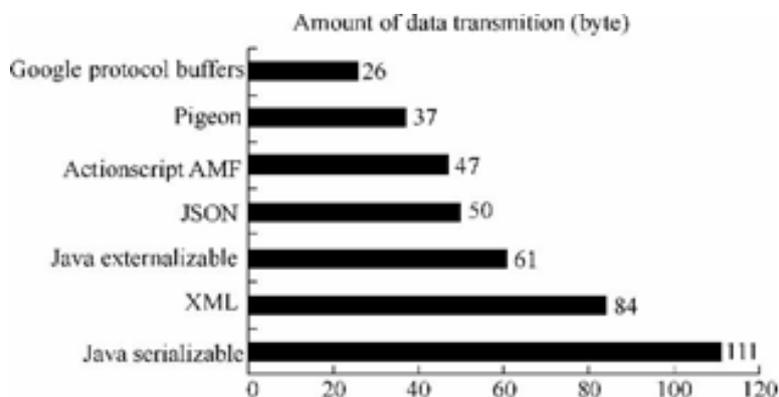
**Fig 1 Processing.**

- 1) Request: in charge of submitting information to server. The information bearer is exemplified by the convention bean, in this way every solicitation must be connected with a convention bean.
- 2) Plugin: a server-side business preparing module which is connected with convention bean's "PluginType" parameter.
- 3) Responder: a customer business preparing module which is connected with a convention bean's "NoticeType" parameter.

In these three ideas, the information bearer is a convention bean. Accordingly, in this structure, these three ideas are firmly connected together by a convention bean.

### 3.3 Test amount of data transmission

In this test, by serializing the same object, we compare the difference of data produced by different serial technologies. Seen from the Fig. 2 about the information data transfer capacity, this structure (Pigeon) has evident point of interest. Since the majority of serial advances contain field name data, it could bring about transmitting extra information, which is no need in custom convention.



**Fig 2: Test response time.**

In this test, we just send a basic solicitation to server each settled time, and the server returns prepared result. Through the information under same conditions, this system (Pigeon) has clear point of preference, whose ongoing execution is normal 40 times of Red5 server. Purposes behind such an outcome are: on one hand, complex conventions, because of unique business needs, keep up countless data and could bring about expanding the measure of information handled; On the other hand, because of its more confused capacities, advanced conventions usage turn out to be more bloated, and would bring about lower preparing effectiveness. Be that as it may, these elements are a bit much in amusement improvement. In this manner, in those applications with high ongoing prerequisites, utilizing custom convention has more noteworthy execution advantage.

### 4. Conclusion:

Protocol bean is used to developed inter communication process between two different towers in mobile communication. This protocol bean uses class concept ,so that java programming can be implemented to transfer and convert the data into required format from sender to receiver. In particular, we have described a simple but very powerful and flexible protocol which provides for variation in individual network packet sizes, transmission failures, sequencing, flow control, and the creation and destruction of process to-process associations. We have considered some of the implementation issues that arise and found that the proposed protocol is implementable by HOST'S of

widely varying capacity. The next important step is to produce a detailed specification of the protocol so that some initial experiments with it can be performed.

**References:**

1. T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Singapore: Pearson Education, Inc., 2002.
2. K. Feher, *Wireless Digital Communications: Modulation and Spread Spectrum Applications*. Upper Saddle River, NJ: Prentice Hall, 1995.
3. J. G. Proakis, *Digital Communications*, 4th ed. NY: McGraw Hill, 2000.
4. G. R. Cooper and C. D. McGillem, *Modern Communications and Spread Spectrum*, NY: McGraw Hill, 1986
5. Communication framework for MMOG based on custom protocol [August 2013, 20].