

Application of messaging in place of interprocess communications

Htut Khine Htay Win
Research Programmer
NCSA

An Archaic Code

```
#include <boost/interprocess/anonymous_shared_memory.hpp>
#include <boost/interprocess/mapped_region.hpp>
#include <iostream>
#include <cstring>
Int main() {
    Using namespace boost::interprocess;
    Try {
        Mapped_region region(anonymous_shared_memory(1000));
        std::memset(region.get_address(), 1, region.get_size());
    }
    Catch (interprocess_exception &exception) {
        Std::cout << exception.what() << endl;
        Return 1;
    }
    Return 0;
}
```

Types of Interprocess Communications(IPC)

- Signals
- Shared Memory
- Named Pipes
- ...

AMQP Messaging system

- What is AMQP Protocols?
- What is an example of product that supports AMQP?
- An interesting fact about AMQP conception.
- Interfacing between multiple programming languages

RabbitMQ

Send:

```
Import pika
```

```
Url = "amqp://guest:guest@localhost:5672/%2f"
```

```
Connection = pika.BlockingConnection(pika.URLParameters(Url))
```

```
Channel = connection.channel()
```

```
channel.basic_publish(queue="hello_queue", body="HELLO WORLD")
```

Receive:

```
Import ...
```

```
Connection =
```

```
Channel =
```

```
Def callback(channel, method, properties, body):
```

```
    print("Message Received is %s" % body)
```

```
# Start IOloop
```

Pitfalls of Rabbitmq



Pitfalls of Rabbitmq

- ConnectionClosed and heartbeat timeout issues
- Blocking vs Non-blocking connections
- Message queues setup, cleanup
- A designated message server

Summary

- AMQP is definitely better than IPC.
- Overheads(developer time + **management stress**) are significantly lower by using rabbitmq than using IPC.

THANK YOU!!!