

**Raju Kolati**  
E-mail: raj.kolati@gmail.com  
Phone: +91-9886872285.

#82/A, Basavasamithi layout,  
Vidyaranpura,  
Bangalore 560097.

- 
- Having 6.8 years Professional experience in C/C++ Development under GNU Linux Environment.
  - Worked in LTE eNB transport protocol stack and GTP-F protocol
  - Adequate knowledge in source control tools like Clear case (Linux), Visual Source Safe, Tortoise SVN, Wireshark, Doors.
  - Good knowledge of networking concepts and IPV4 concepts.
  - Good knowledge on Networking device driver concepts.
  - Experience in Embedded and Telecom product development.
  - Experience in Agile methodology way of working
  - Experience in development of system software on Unix/Linux system programming.
  - Sound understanding and knowledge of UNIX and Linux OS internals.

#### **Educational Qualification**

- Bachelor of Engineering (BE) in Electronics & Communication Engineering from VTU university in the year 2010 with overall 65 %.
- 12th with 82% from SJVP College Harihar (PU Board Education, Karnataka).
- 10th with 92% from CES High School Ranebennur (KSSC Board Education, Karnataka)

#### **Technical Skills**

- Programming Languages : C , Basic C++.
- Architecture : Intel Core 2, Red Hat Linux.
- Platform : Linux, Windows.
- Familiar with : Python, Bash Shell.
- Debugging Tools & Simulators : Log reader and GDB.
- Development Tools : Clear case (Linux), SVN, GNU Tools (GCC, GDB), Valgrind , Wireshark ,Remote GDB

#### **Employment Details**

**Company** : Nokia (Aug 2014 – Present).  
**Designation** : Software Design Engineer.

**Company** : L & T Technology Service (May 2011 – Aug 2014).  
**Designation** : Software engineer .

## Project Profile

Company: Nokia (Aug 2014 – Present):

1. Inter-site Carrier Aggregation : LTE 4.5G feature (Single core and Dual core eNBs)	
Role Played	Developer (Had been to china Hangzhou for deployment support )
Skills Used	C/C++, IPC mechanism, Pthreads , Socket programming ,SVN, GNU Tools IPV4 & IPV6
Networking Protocols	UDP, GTP-F,
Environment	Intel Core 2, Red Hat Linux.
Platform	AXM, Flexi platform, Cavium
Team Size	3
Period	6 Months.
Description	The benefit for the customer lie in leveraging the advantages of carrier aggregation like fast U-plane load balancing or increased peak throughput. Interworking of user plane packets for inter-site Carrier Aggregation over X2 interface (ICOM GW). In Uplink direction once packet reaches to ICOMGW from UE. ICOM GW parses the BIP header to get the device id then device id and Peer Ip lookup will happen in fast path and based on this input, ICOM GW will prepare GTP-F packet and it will forward to peer ENB. In Downlink case ICOM GW will parse the GTP-F packet and forwarding the packet to corresponding baseband modules
Responsibilities	<ul style="list-style-type: none"><li>➤ Design and Implementation of ICOM GW fast path code Stack and implemented firewall rules for ICOM ports</li><li>➤ ICOM GW UT GTest cases and Gmock implementation.</li><li>➤ Involved in customer deployment of this feature</li><li>➤ Testing traffic with IXIA setup</li><li>➤ Debugging and Bug fixing</li></ul>

<b>2. Two Way Active Measurement Protocol stack (TWAMP Stack)</b>	
<b>Role Played</b>	Developer.
<b>Skills Used</b>	C/C++, IPC mechanism, Pthreads , Socket programming ,SVN, GNU Tools IPV4 & IPV6
<b>Networking Protocols</b>	UDP, ICMP, ARP,
<b>Environment</b>	Intel Core 2, Red Hat Linux.
<b>Platform</b>	AXM, Flexi platform
<b>Team Size</b>	2
<b>Period</b>	9 Months.
<b>Description</b>	<p>Two way active measurement protocol (TWAMP) actively measure and supervise the conditions through the mobile backhaul transport network between two points, using RFC863 UDP Echo and RFC5357 TWAMP protocols. These two points can be the eNB on one side and the SEG (third party Security Gateway) or any other (third party) site router or measuring equipment on the other side. If required the measurements can be done for each QoS class separately.</p> <p>For the measurement one point generates test traffic according to the IETF defined Two Way Active Measurement Protocol (TWAMP), specified in [RFC5357] and analyses the results. The other point must at least be able to send back the test traffic. In an advanced solution the other point shall add Time stamps when the packet was received and transmitted back thus allowing the sender to deduct the internal processing time from the calculated delay time according to RFC5357.</p>
<b>Responsibilities</b>	<ul style="list-style-type: none"> <li>➤ Design and Implementation of complete TWAMP Stack and implemented firewall rules for TWAMP chain</li> <li>➤ Twamp Module Test cases and script implementation.</li> <li>➤ Development Testing with VLAN configuration and IPSEC configuration</li> <li>➤ Debugging and Bug fixing</li> </ul>

<b>3. Plug and Play (Auto connection with DHCP Server and without DHCP Server )</b>	
<b>Role Played</b>	Developer.
<b>Skills Used</b>	C/C++, IPC mechanism , Pthreads , Socket programming ,SVN, GNU Tools IPV4
<b>Networking Protocols</b>	DHCP,ICMP,ARP,RARP
<b>Environment</b>	Intel Core 2, Red Hat Linux.
<b>Platform</b>	AXM Flexi platform
<b>Team Size</b>	4
<b>Period</b>	6 Months.
<b>Description</b>	<p>Installing new sites is an expensive, complicated and time consuming procedure for network operators. After hardware installation (including cabling and powering), a specially trained commissioning engineer needs to be present on site in order to commission and integrate the new BTS into the operator's network. Once the BTS is integrated into the operator's network, it can be operated and maintained remotely from within the operator's network.</p> <p>The idea of the "Plug-and-play" feature is to make new site installations easier, cheaper, faster and more reliable by letting the BTS automatically integrate itself into the operator's network after it has been powered up for the first time. By doing this, the commissioning engineer could operate and maintain all the base stations from a central place within the operator's network right from the beginning.</p>
<b>Responsibilities</b>	<ul style="list-style-type: none"> <li>➤ Involved in the complete Implementation of Auto connection with DHCP and without DHCP Server</li> <li>➤ Development Testing with VLAN configuration and IPSEC configuration</li> <li>➤ Debugging and Bug fixing</li> <li>➤ Part of agile team in iterative development</li> </ul>

**Company : L & T Technology Service (May 2011 – Aug 2014).**

<b>1.GSM and GPRS Protocol Development for GSM BSS</b>	
<b>Role Played</b>	Developer.
<b>Skill/Tools</b>	C/C++, Clear Case, SVN, GNU Tools.
<b>Environment</b>	Intel Core 2, Red Hat Linux.
<b>Team Size</b>	10
<b>Period</b>	8 Months
<b>Description</b>	To test GSM BSS MS and SGSN simulator have been developed. As part of this simulator GSM Protocol (BSSMAP, BTSMAP, CC, MM, RR,GMM, SM) Parser, pack and unpack library have been developed.
<b>Responsibilities</b>	<ul style="list-style-type: none"> <li>• Team member</li> <li>• Implementation of CC,RR,BSSMAP ,GMM,SM protocol library</li> </ul>

	<ul style="list-style-type: none"> <li>• Implementation of CC,RR, BSSMAP,GMM,SM parser library</li> <li>• Integration and testing of CC,RR,BSSMAP with MS simulator and SGSN Simulator</li> </ul>
--	---

2. BTS-MS GPRS & DSP Simulator	
<b>Role Played</b>	Developer.
<b>Skills Used</b>	C/C++, Clear Case, SVN, GNU Tools.
<b>Environment</b>	Intel Core 2, Red Hat Linux.
<b>Team Size</b>	5
<b>Period</b>	5 months
<b>Description</b>	<p>To test GSM BSS BTS-MS GPRS simulator has been developed. As part of this simulator pack, unpack parser libraries have been developed &amp; GTP protocol has been implemented as a part of Simulator.</p> <p>DSP Simulator module is independent module which provides interface between MS Simulator and BTS, when a signaling or traffic message received from BTS, the application will call the appropriate API from DSP simulator and Process them and take appropriate actions.</p>
<b>Responsibilities</b>	<ul style="list-style-type: none"> <li>• GMM, SM protocol library</li> <li>➤ Implementation of GMM,SM parser library</li> <li>➤ Implementation of Interface messages between DSP and LAPDm.</li> <li>➤ Implementation of interface message structure between DSP Simulator and MS simulator.</li> <li>➤ Implementation of API between BTS,DSP Simulator and MS simulator</li> <li>➤ Implementation of DSP Simulator</li> </ul>

### Personal profile

Fathers Name : Raju Kolati  
 Date of Birth : 20 JUL 1988  
 Maternity : Male.  
 Marital Status : Single.  
 Language Known : English, Kannada, Hindi.

I solemnly declare that the information stated above is correct to the best of my knowledge and belief.

**Place: Bangalore**  
**Date:**

**Raju Kolati**