Continuous Integration Tool for Load Development in Femto Gateway

Ebin.P.Eldho[1], Ananthakrishnan[2], K N Raja Rao[3]

[1]M.Tech Student, Digital Communication, R.V College of Engg.
[2]Technical Leader, Nokia Siemens Networks, Bangalore-45
[3]Professor, R.V College of Engg., Bangalore-59

Abstract—A 3G Femtocell allows service providers to extend service coverage indoors, especially where access would otherwise be limited or unavailable. The project deals with designing and implementing an automatic testing tool, using Visual Studio C#. The development of tool named CI GUI Tool is aimed at achieving the continuous integration of Test cases in Femto Gateway. The project involves analysis of manual execution of test procedures in femto gateway and automating the same. The automatic testing tool is designed for 'Agile Model' instead of conventional 'Waterfall Model'.

General Terms—3g, Umts, Femtocell.

Keywords—Femtocell, Femto Access Point, Femto Gateway, Continuous Integration,TR69 Protocol.

I. INTRODUCTION

In telecommunications, a femtocell is a small cellular base station, typically designed for use in a home or small business. It is designed to extend mobile operators' coverage area and improve their capacity. A femtocell produces a coverage range up to 30-50 meters and connects to the service provider's network via broadband such as DSL or cable [1]. Current designs typically support 2 to 4 active mobile phones in a residential setting and 8 to 16 active mobile phones in enterprise settings.

Once plugged in, the femtocell connects to the MNO's mobile network, and provides extra coverage. The user must then declare which mobile phone numbers the femtocell must allow/ authenticate, via a web interface provided by the MNO. When these mobile phones arrive under coverage of the femtocell, they switch over from the macro cell to the femtocell automatically. All the communications will then automatically go through the femtocell. When the user leaves femtocell coverage area, phone will hand over seamlessly to the macro network. Femtocells require specific hardware, middle ware and software, which needed to be developed and tested for their functionality.

Femtocells offer benefits for both subscribers and operators. Femtocells can be used to off-load traffic from the macro cellular network there by reducing infrastructure cost. Thus femtocells enable the mobile operators to move from conventional single-macro base station with high number of users into small coverage area with limited number of users [2]. Also, Operators can ensure that their mobile services are used in the home despite the availability of competing technologies (e.g. Wi-Fi, VOIP using Skype). The subscriber experiences better coverage, voice service and higher data throughput [3].Although much attention is focused on UMTS, the concept is applicable to all standards, including GSM(EDGE), CDMA2000, TDSCDMA, WiMax and LTE solutions.

II. FEMTO NETWORK COMPONENTS

The essential network elements in the femtocell solution are the femtocell itself, which consist of femto Access Point (FAP), Femto Gateway (FGW) and Core Network. A FAP usually allow authorized users to connect to the femtocell and to utilize services other than voice, such as text or real time multimedia streaming etc. For development purpose in industry, we simulate the FAP's and Core Network. These simulated FAP's & Core network are together known as Functionality Tester (FT). This simulated FT is interacting with FGW using 'Load'. It utilizes TR69 protocol specifications for communication.

2.1 Femto Access Point

A Femto Access Point is a small wireless base station that is installed to the user premises, serving as a femtocell. The FAP offer them the same services as if they were operating under a regular NodeB. The device is low cost and relatively small in size and can be installed to the user's home or office. The operator has no exact control of the location. The FAP is powered from the user's electric network using most likely an external power adapter.

2.2   Femto Gateway

The Femto Gateway is the device used to connect the FAP's to the network. FGW aggregate FAP's traffic backhauled over fixed line broadband links such as cable modem, DSL or Optical fiber.ie, FGW functions as a concentrator for all traffic received from the FAP. The FGW's main responsibility is to function as a single RNC towards the Core Network and hide the details of different FAP'S connected to it. Also the FGW functions as the core network towards each FAP. In the femtocell logical architecture designed by 3GPP, the FGW is placed in the operator's premises [4].
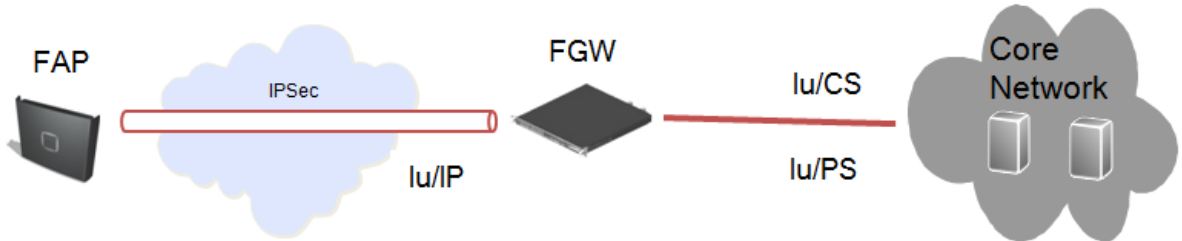

Fig 1: **Femtocell Overview**

FGW communicates with the FAP using the Iu/IP interface and with the network using the standard Iu (Iu/CS and Iu/PS) interface [5]. The FGW has the following functions:

- It provides authentication and certification to allow only data to and from authorized FAPs
- The FAP-GW aggregates traffic from a large number of FAPs and provides an entry point into the operator core network.

2.3   Core Network Elements

The core network involves all the elements used in the UMTS architecture. In addition, it involves the new network element Operational & Management System (OMS). The Asynchronous Transfer Mode (ATM) is defined for UMTS core transmission.
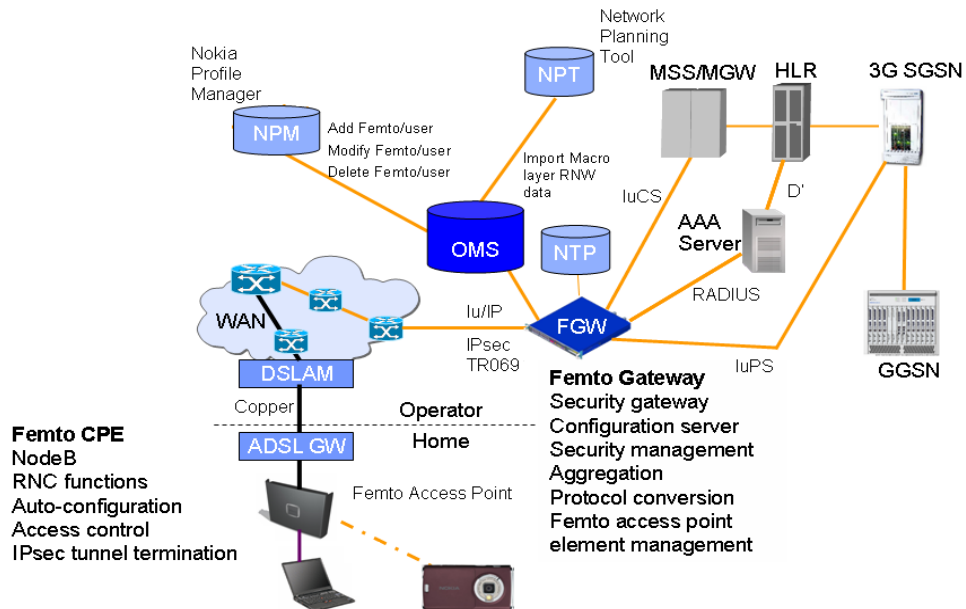

Fig 2: **Femto Network Architecture**

The Core Network is divided in to circuit switched and packet switched domains. Some of the circuit switched elements are Mobile services Switching Centre (MSC), Visitor location register (VLR) and Gateway MSC. Packet switched elements are Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN). Some network elements, like EIR, HLR, VLR and AUC are shared by both domains.

III.   CONTINUOUS INTEGRATION (CI)

We use Continuous integration to frequently integrate one's new or changed code with the existing code repository. Continuous integration aims to improve the quality of 'Load', by replacing the traditional practice of applying quality control after completing all development with continuous processes of applying quality control which is implemented frequently.

In the present contest, each of the possible interactions between the FT and FGW is a Test case. For each and every scenario in the n/w (Circuit Switched call, Packet Switched call, hand over, authentication, node b registration etc), there should be a corresponding test case. A 'Load' is defined as a combination of test cases. The simulated Functionality Tester is interacting with FGW using 'Load'.

3.1 Load Development Model

A Load development methodology is a framework that is used to structure, plan, and control the process of developing loads. Each model has its own recognized strengths and weaknesses. One load development methodology is not necessarily suitable for use by all projects. Depending on the model used, various process associated with the development cycle also varies.

Presently in femto Project, we are manually integrating the test cases in to the gateway. Here the Load development utilizes a linear and sequential approach accomplished by making use of Waterfall model. However, Load development model required for achieve continuous integration is Agile model. Agile is a load development methodology based on iterative and incremental development, where requirements and solutions evolve through collaboration between self-organizing, cross-functional teams.

Agile methods break tasks into small increments with minimal planning. Here, at each step of development, the build is checked and verified for bugs. Once a bug is found in a particular test case code, it is either fixed or passed back to the previous teams which are responsible for fixing the bug. Then the team will fix that bug and the code is passed down again and the test cases are tested for bugs again on its way down.

3.2 Continuous Integration Of Test Case

A test case may be defined as a set of conditions under which a tester will determine whether an application or software system is working correctly or not. The test cases are written and modified using TTCN. Test cases will be written for existing scenarios in a femto Gateway. These test cases are to be integrated into femto Gateway to check for its functionality. Instead of manually integrating and analyzing each of the test cases, we can do automation of test cases.

'Jira' is the repository or database for managing the test cases. The test case from the jira has to be fed to the CI GUI Tool as its input. We make use of CI to automatically integrate these testcases into the FGW & FT. Thus by developing a CI Tool, we are to achieve the continuous integration of test cases, as well as changing the load development model from Waterfall to Agile.

3.2.1 Advantages

1. When unit tests fail or bugs emerge, developers can try revert the code base to a bug-free state, without wasting time for debugging.
2. Developers detect and fix integration problems continuously - avoiding last-minute chaos at release dates.
3. Early warning of broken/incompatible code.
4. Early warning of conflicting changes.
5. Immediate unit testing of all changes.
6. Constant availability of a "current" build for testing, demo, or release purposes.
7. Immediate feedback to developers on the quality functionality, or system-wide impact of code they are writing.
8. Frequent code check-in pushes developers to create modular, less complex code.

IV.   FT TESTER

FT is a tool to simulate OMS, FAP and Core Network. It consists of ETS and PEU. The Executable Test Suite (ETS) and Protocol Execution Unit (PEU) are constructed by using Telelogic TAU Tester, which includes the TTCN-3 compiler (t3cg) and other ASN tools. Test cases are written in TTCN3 language, TTCN-3 compiler (t3cg) is used to convert them to C and GNU tools to compile into executable binaries.

The Protocol Execution Unit (PEU) runs the lower protocol layers, which act as service providers towards upper layers in TTCN. PEU is a server and ETS is a client through which TTCN test case are executed.

4.1 TEST CASE EXECUTION
1. Establish IPsec tunnels between FT machine and FGW.
2. Start the PEU binary and then start the ETS
3. Once ETS is up, execute "list" command to see the list of test cases available to execute.
4. To run one test case execute "run <test case name>" on the ETS prompt.
5. Verdict could be PASS, FAIL, ERROR or test case execution hang.
6. ETS generates log.txt in log directory to know more on the test case execution flow.
7. Stop ETS by execution command "shutdown" on the ETS prompt.
8. Stop PEU by pressing "Enter" followed by "E".

4.2 METHODOLOGY
1. Read all the information about the test cases from 'Jira' database and write into an XML file.
2. Using tags, grep the necessary data from XML file and serve it as input to the CI GUI tool.
3. In CI GUI Tool, the test cases are classified according to their types.
4. The CI GUI Tool initiates the continuous integration of test cases.
5. IPsec tunnel is established. PEU is started and later ETS is started. ETS binaries are written into log.txt file.
6. CI GUI Tool retrieves the data available in the log.txt created by ETS.
7. Thus the status of test cases can be made available at CI GUI Tool.
7. The failed test cases are modified and are again continuously integrated until errors are null.
8. A continuous integration with no errors will lead to the development of a successful load.

V.   RESULTS

Depending on the Priority, test cases are classified as Blocker, Critical, Major or Minor. For a successful load build, all test cases with priority as Blocker & Critical should pass. However, a pass percent of 80 for Major and Minor test cases is acceptable for successful Load built.

For a usual 3g femtocell currently in deployment, there are around 5000 test cases and will increase depending on the complexity of deployment. Among this, there will be around 2000 mandatory test cases which should pass for the successful completion of load. While we are doing manual integration of these test cases, we are to run each of these test cases and check whether it passes or fails.

Considering that execution of a test case takes approximately 2 minutes, a total of 2000 test cases take 2000*2=4000 minutes = 2.7 days. In case of errors, we need to fix the bugs and again start the load build from the beginning. Thus it may take days for the completion of a build.

However, using continuous integration we are continuously integrating the test cases. Here, since the process is automated, the process of load development can be completed in very few hours. Also, in cases of an error, we can fix the code, integrate it to the repository and continue the load build. There is no need to start the process from the beginning.

Fig 3: **Sample Testcase kept in database 'Jira' with Priority as Major**

Fig 4: **Screen Shot Of Input XML file created**

5.1 PEU Configuration

1. Place the tar ball in the PEU folder <root>/PEU. Eg. FGWFT_REL2.0_RIS5_DELIVERY1.0_PEU.tar
2. Untar the tar file. tar –xvf FGWFT_REL2.0_RIS5_DELIVERY1.0_PEU.tar
3. Base folder <2.5.1> is created under the PEU folder.
4. <2.5.1/TP3> the following folders are present

"axis2c-src-1.4.0"- Shared library used to send/receive messages for TR069 session.

"bin"   - PEU binary is stored.

"peu"   - Binary is run and configuration files are present.

"scripts"   - ini files and script to install PEU.

"Template" - Templates of the configuration files to be generated.

5.2 ETS Configuration

1. ETS tar placed in the required path
2. install.sh run from the <fgw-ft-ets> folder.
3. Provide path for 'TauTester', 'ttcn3libs', 'logviewer' & 'user makefile' as and when install script asks for these

5.3 Pre defined Configuration Parameters

rnctesterParam.mp_SrcUdpIpAddress_iuFapCS	— Source Udp ip address for sending CS data on FAP
rnctesterParam.mp_UserPlane_StartingSrcPort_iuFapCS1	-Starting Source Port for sending CS userplane data on FAP1
rnctesterParam.mp_UserPlane_DestPort_iuFapPS1	—Destination Port of the peer for PS U-plane data on FAP1
rnctesterParam.mp_UserPlane_DestPort_iuPS1	—Destination Port of the peer for PS U-plane data on CN1
rnctesterParam.mp_TLA_Header_Fap	—Header is not added to TLA if value is set to 0.(for IP)
rnctesterParam.mp_TLA_Header_ps	—Header is not added to TLA if value is set to 0.(for ATM)
rnctesterParam.mp_pcFgw_cs	—Point code of FGW towards CS FAP
rnctesterParam.mp_pcFgw_ps	—Point code of FGW towards PS FAP
rnctesterParam.mp_pcFgw_cn_cs	—Point code of FGW towards CS CN
rnctesterParam.mp_pcFgw_cn_ps	—Point code of FGW towards PS CN
rnctesterParam.mp_pcFap_cs_1	—Point code of CS FAP 1
rnctesterParam.mp_pcFap_ps_1	—Point code of PS FAP 1
rnctesterParam.mp_pcCN_cs_1	—Point code of CS CN 1
rnctesterParam.mp_pcCN_ps_1	—Point code of PS CN 1
rnctesterParam.mp_VPI_UP_IU_CS_1	—VPI of CS CN 1 with the configured alcap path id as 1
rnctesterParam.mp_VCI_UP_IU_CS_1	—VCI of CS CN 1 with the configured alcap path id as 1
rnctesterParam.mp_SrcUdpIpAddress_iuCS1	—Source Udp ip address for sending CS data on CS CN1
rnctesterParam.mp_SrcUdpIpAddress_iuPS1	—Source Udp ip address for sending PS data on PS CN1
rnctesterParam.mp_transportType_iuCS1	—Transport type on CS CN1. Either e_IP or e_ATM
rnctesterParam.mp_transportType_iuPS1	—Transport type on PS CN1. Either e_IP or e_ATM
rnctesterParam.mp_continueOnMismatch	—When this module parameter is set to true, the BTSOM test cases will continue on template mismatch.

5.4 Fap Side Configuration

FGWM3UAPORT_IU_FAP_PS   - Used in the file m3ua_IuFap*.cfg where destination sctp port of the peer
                                                              (FGW) is defined.

FGWM3UAHOST   - Used in the file m3ua_IuFap*.cfg where destination sctp IP address of peer
(FGW) is defined.

M3UAHOST_IU_FAP_PS*   - Used in the file m3ua_IuFap*.cfg where source sctp IP address of host
                                                              (FAP) is defined.

M3UAIUPORT_IU_FAP_PS*   - Used in the file m3ua_IuFap*.cfg where source sctp port of the host (FAP) is
defined.

OPC_IU_FAP_CS*   - Originating (Own) point code of the FAP used in m3ua_IuFap*.cfg
                                          and sccp_IuFapCS*.cfg.

OSSN_IU_FAP_CS*   - Originating Sub system number used in the sccp_IuFapCS*.cfg.

OPC_IU_FAP_PS*   - Originating (Own) point code of the FAP used in m3ua_IuFap*.cfg
                                        and sccp_IuFapPS*.cfg.
OSSN_IU_FAP_PS*   - Originating Sub system number used in the sccp_IuFapPS*.cfg.
DPC_IU_FAP_CS*   - Destination (remote) point code of the FAP used in m3ua_IuFap*.cfg
                                        and sccp_IuFapCS*.cfg.
DSSN_IU_FAP_CS*   - Destination Sub system number used in the sccp_IuFapCS*.cfg.
DPC_IU_FAP_PS*   - Destination (remote) point code of the FAP used in m3ua_IuFap*.cfg
                                        and sccp_IuFapPS*.cfg.
DSSN_IU_FAP_PS*   - Destination Sub system number used in the sccp_IuFapPS*.cfg
FAP_FGWIUPORT   - Destination UDP port of the FAP (local for FGW) used in the file struct.db.
FAPIUPORT_IU_PS_1   - Local UDP port of the FAP used in the file struct.db.
FAPHOST_IU_PS_1   - Local UDP IP address of the FAP.
RTP_SRC_MULTIPLEX_PORT_IU_CS_FAP_*   – RTP source multiplex port of the FAP(Local)
RTP_DST_MULTIPLEX_PORT_IU_CS_FAP_*   – RTP destination multiplex port of the FGW (remote)

5.5 Core Network Configuration

FGWM3UAPORT_IU_PS* - Used in the file m3ua_IuPS*.cfg where destination sctp port of the peer
                                (FGW) is defined.
FGWM3UAPORT_IU_CS* - Used in the file m3ua_IuCS*.cfg where destination sctp port of the peer (FGW) is
                                defined.
M3UAHOST_IU_CS_*   - Used in the file m3ua_IuCS*.cfg where local sctp IP address of the host is defined.
M3UAIUPORT_IU_CS*   - Used in the file m3ua_IuCS*.cfg where local sctp port of the host is defined.
OPC_IU_CS_*   - Originating (Own) point code of the MSC used in m3ua_IuCS*.cfg
                                    and sccp_IuCS*.cfg.
DPC_IU_CS_*   - Destination (remote) point code of the MSC used in m3ua_IuCS*.cfg
                                    and sccp_IuCS*.cfg
OSSN_IU_CS_*   - Originating Subsystem number used in the file sccp_IuCS*.cfg, m3ua_IuCS*.cfg
DSSN_IU_CS_*   - Destination Subsystem number used in the file sccp_IuCS*.cfg, m3ua_IuCS*.cfg
M3UAHOST_IU_PS*   - Used in the file m3ua_IuPS where host sctp IP address of the host is defined.
M3UAIUPORT_IU_PS*   - Used in the file m3ua_IuPS where host SCTP port of the host is defined.
OPC_IU_PS_*   - Originating (Own) point code of the SGSN used in m3ua_IuPS*.cfg
                                    and sccp_IuPS*.cfg.
DPC_IU_PS_*   - Destination (remote) point code of the MSC used in m3ua_IuPS*.cfg
                                    and sccp_IuPS*.cfg
OSSN_IU_PS_*   - Originating Subsystem number used in the file sccp_IuPS*.cfg, m3ua_IuPS*.cfg
DSSN_IU_PS_*   - Destination Subsystem number used in the file sccp_IuPS*.cfg, m3ua_IuPS*.cfg
SGSNIUPORT_IU_PS*   - Local UDP port of the SGSN used in the file struct.db.
CN_FGWIUPORT   - Destination UDP port of the peer (FGW) used in the file struct.db.
SGSNHOST_IU_PS_1   - Local UDP IP address of the SGSN used in the file struct.db.
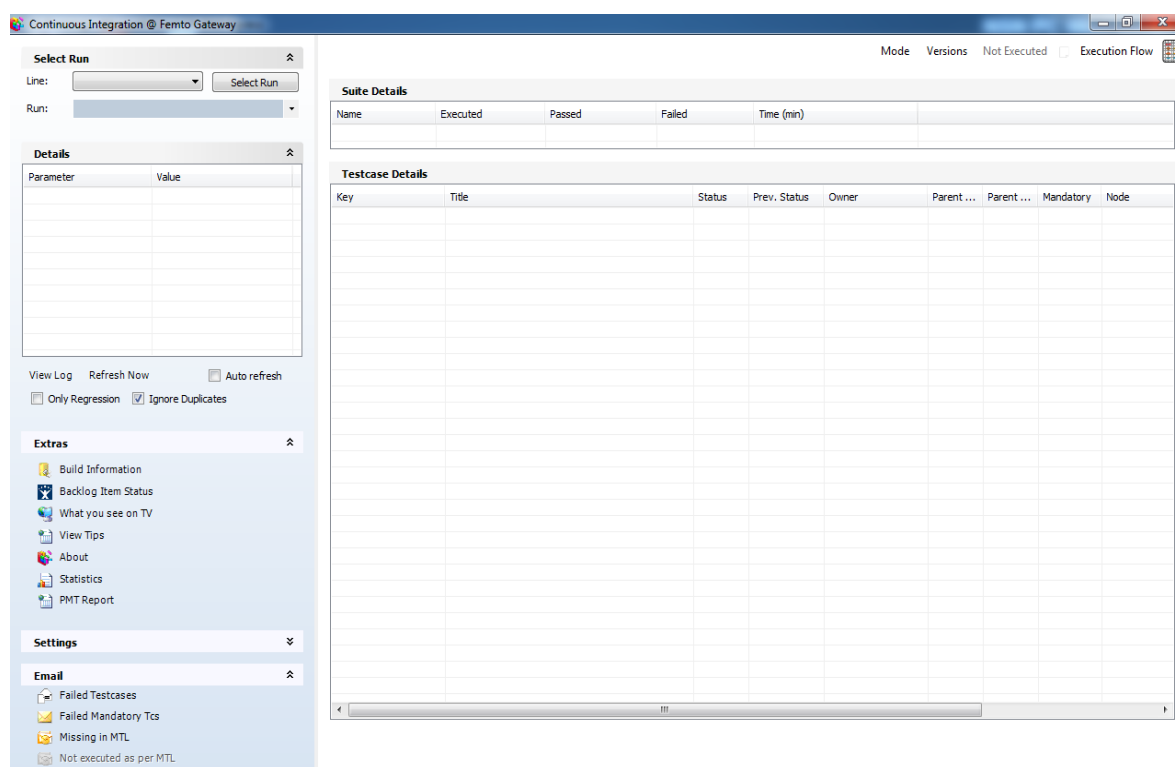
Fig 5: **Screen Shot Of CI GUI Tool**

VI.   TECHNICAL ASPECTS OF FEMTOCELLS

The capacity potential of femtocells can be verified rapidly from Shannon's law, which relates the wireless link capacity (in bits per second) in a bandwidth to the SINR. The SINR is a function of the transmission powers of the desired and interfering transmitters, path losses, and shadowing during terrestrial propagation. Path losses cause the transmitted signal to decay as $Ad^{-\alpha}$, where A is a fixed loss, d is the distance between the transmitter and receiver, and α is the path loss exponent. The key to increasing capacity is to enhance reception between intended transmitter-receiver pairs by minimizing d and α . Simultaneously, additional benefits in network-wide spatial reuse can be obtained by, but not restricted to, exploiting diversity, and employing interference cancellation, interference suppression, and interference avoidance techniques. Femtocells enable reduced transmit power while maintaining good indoor coverage. Penetration losses insulate the femtocell from surrounding femtocell transmissions. Assuming a fixed receive power target with a path loss propagation model (no fading), and denoting $\alpha(\beta)$ as the outdoor (indoor) path loss exponent, overlaying an area L^2 with N femtocells results in a transmit power reduction on the order of $[10(\alpha-\beta) \log 10 \, L + 5 \, \beta \log 10 \, N]$ dB.

For example, choosing a cell dimension of $L = 1000$ m and $N = 50$ femtocells, with equal path loss exponents $\alpha = \beta = 4$, femtocells give a transmit power saving of nearly 34 dB. When the indoor path loss exponent is smaller, say $\beta = 2$, the transmit power savings increase to nearly 77 dB. To   summarize, the capacity benefits of femtocells are attributed to:
   • Reduced distance between the femtocell and the user, which leads to higher received signal strength.
   • Lowered transmit power, and mitigation of interference from neighboring macro cell and femtocell users due to outdoor propagation and penetration losses.

As femtocells serve only around one to four users, they can devote a larger portion of their resources (transmit power and bandwidth) to each subscriber. A macro cell, on the other hand, has a larger coverage area (500 m–1 km radius) and a larger number of users; providing quality of service (QoS) for data users is more difficult. The first two points illustrate the dual improvements in capacity through increased signal strength and reduced interference. The third point shows that deploying femtocells will enable more efficient usage of precious power and frequency resources. The assumption here is that the wired broadband operator provides sufficient QoS over the backhaul. Otherwise, backhaul capacity limitations could reduce the indoor capacity gains provided by femtocells.

VII.   FEMTOCELLS ATTRIBUTES AND FUTURE SCOPE

Femtocells use over-the-air standard protocols, to communicate with standard mobile devices. In flexible deployment scenarios, technology specific femtocells can be deployed easily with any existing and future technologies like GSM, LTE, WiMAX, etc [6]. Femtocells are gaining popularity because they can be used to access the radio link where radio service through traditional macrocells coverage cannot be provided. Furthermore, femtocells provide higher capacity by significantly minimizing the level of interference in home environments.

Femtocells create extra network capacity with restricted user access; which allows users to enjoy higher data rate services like multimedia streams, online gaming, video and TV broadcasts. The use of femtocells will definitely save UE battery. This is because femtocells offer much smaller distance between the UE and HNB compared to the much larger distance typically between the UEs and macrocells. The femtocell operates in the licensed spectrum, and assures no interference from unauthorized users, using DSL, broadband connections and optical fiber technologies as backhaul to transport data over the Internet protocol.

Applications of femtocells are not limited to only home users. In a broader perspective, femtocells can be utilized for a wider set of applications. One application of femtocells is to use them in corporate environments. Unlike home installations, in offices more than one femtocell could be used depending upon the size of the organization. In this type of arrangement the femtocell can also be connected with other IP-enabled devices such as Private Branch Exchange (PBX), printers and LANs etc. Additionally in enterprises, handover between femtocells will be needed. Another model of femtocell application could be to use them in aircrafts, trains or in passenger ferries. This type of femtocell deployment can use satellite as a backhaul. From operator's point of view, significant amount of traffic can be offloaded from existing macro cell sites after femtocell deployment. Operators in future will need a fewer number of macro cell sites in the presence of the femtocell.

VIII.   CONCLUSIONS

Femtocells have the potential to provide high quality network access to indoor users at low cost, while simultaneously reducing the burden on the whole system. This article has identified the key benefits of femtocells, the technological and business challenges, and research opportunities. Presently, testing of these femtocells are carried out by manually integrating the test cases. This testing procedure has following disadvantages:

1. Time required for testing is more.
2. Full functionality test not possible due to time constraint
3. Poor reliability.
4. If the testing is to be carried out during design stage, co-ordination between teams is difficult, as the implementation or realization of different teams could be different.
5. Man hours wasted because of (4).

The project will overcome all the lacuna of the present waterfall model of Load Development. By implementing proposed test tool, the Load development can be done using Agile model.

IX.   ACKNOWLEDGMENTS

REFERENCES

[1].   Simon R. Saunders,2009, "FEMTOCELLS Opporunies and Challenges for Business And Technology", John Wiley & Sons.

[2].   F. Chiussi, D. Logothetis, I. Widjaja, and D.Kataria, 2010, "Femtocells [Guest Editorial]", IEEE Communications Magazine.

[3].   Jen Chen, Peter Rauber, Damanjit Singh, Chandru Sundarraman, Peerapol Tinnakornsrisuphap, Mehmet Yavuz, Jan 28, 2010, Femtocells – Architecture & Network Aspects.

[4].   White Paper,Version 0.2.2,Nokia, Femto Gateway Definition.

[5].   V. Chandrasekhar and J. Andrews, September 2008, Femtocell Networks: A Survey, IEEE Communications Magazine.

[6].   Release 9, TS 22.220, 3GPP, Technical Specification Group Services and System Aspects; Service requirements for Home Node B (HNB) and Home eNode B (HeNB).