Industrial Placement Report

DALE LANE

(ma8cdl@bath.ac.uk)

GSM RADIO SUB-SYSTEMS SOFTWARE ENGINEER - MOTOROLA

THE ORGANISATION (PART ONE)

INTRODUCTION

Motorola is a large multi-national corporation, with businesses spanning several industries. The main focus of the business is in "providing integrated communications solutions and embedded electronic solutions"[1]. This covers a wide variety of products and services, and the corporation is organised in a rigid hierarchy to efficiently manage the development of such diverse technologies.

The corporation is divided into three main sectors:

- Integrated Electronic Systems Sector
- Semiconductor Products Sector
- Communications Enterprise

I worked within the last of these: the Communications Enterprise (CE). The largest of the three sectors, CE accounts for approximately 70% of Motorola's business.

CE itself is divided into seven major business units:

- Personal Communications Sector
- Broadband Communications Sector
- Commercial Solutions Sector
- Government Solutions Sector
- Industrial Solutions Sector
- Global Telecom Solutions Sector
- Internet and Networking Group

I worked for the Global Telecom Solutions Sector (GTSS) the telecoms part of CE, responsible for developing a wide variety of telecommunications technologies, including GSM (Global System for Mobile Communication) and GPRS (General Packet Radio Service).

Because of the wide scope of technologies developed by GTSS, the sector is further split into organisations, allowing departments to specialise in a certain field. I worked within the GSM part of the business, responsible for maintaining and developing Motorola's GSM solutions.

OVERVIEW OF GSM

GSM stands for Global System for Mobile Communication, and is a digital mobile telecommunication technology that is widely used in Europe and other parts of the world. It has over 120 million users worldwide and is available in 120 countries[2], forming a billion dollar industry.

It was developed to replace the early analogue cellular telephone systems of the early 1980s. It is often referred to as *second generation*, or 2G, because of this.

The architecture of a GSM network is hierarchical in design, allowing for several mobile phones to be serviced by a single point of contact with the terrestrial telephone network (PSTN).

Figure 1 below shows a diagrammatical overview of the GSM architecture. The diagram shows the way that the architecture can be divided into four main areas.

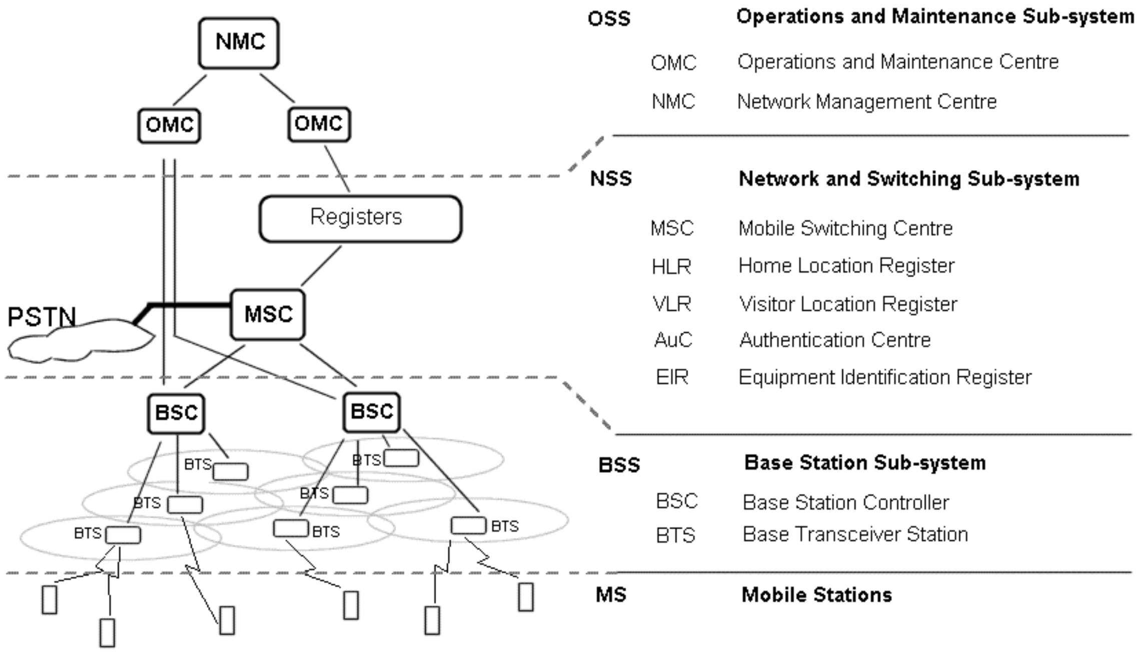

Figure 1 - GSM System Architecture

The OSS allows the network operator to control and maintain their network, allowing access to any site. It connects to both the BSS and NSS, allowing direct manipulation of any site when required.

The NSS has two main functions – firstly, it is responsible for the connection to the terrestrial phone network (PSTN), which is handled by the MSC (Mobile Switching Centre), and secondly, it is responsible for keeping track of the location of customers' mobiles, and verifying whether or not a customer should be allowed to make calls, which is handled by the AuC (Authentication Centre) using the Registers.

My department was responsible for BSS – Base Station Sub-systems. This is the part of the system that communicates with mobiles and handles calls. Each BSC (Base Station Controller) is responsible for controlling a number of BTS (Base Transceiver Stations), which are where the radios are located.

BSS SOFTWARE DEPARTMENT

Each sub-system detailed above (MS, BSS, NSS and OSS) is further divided into three departments: Firmware (responsible for developing the code that is stored in the programmable read-only memory), Hardware and Software.

I worked in **BSS Software**, responsible for maintaining and developing the BSS Software system – the real-time embedded software system that controls the Motorola Base Station platforms. The department has over 75 software engineers and is based mostly in Swindon, Wiltshire.

The BSS Software system is very complex, and includes nearly one hundred processes. To make this more manageable, the processes are logically divided into a number of groups, called *Functional Areas* (FA). The department is divided into sub-departments, also called FA's, with each FA responsible for a particular group of processes. Each FA operates as a small department with a *Team Leader* who fulfils a Line Manager role, responsible for managing the FA on a day-to-day basis.

I worked in the **Radio Sub-systems** (RSS) FA, responsible for the RSS group of processes. The Team Leader was Peter Gill. My manager, responsible for a number of FA's, was Paul Armstrong, and the department manager for BSS Software was Robert Wicks.

EUROWAY

As mentioned previously, BSS Software is based in Swindon, in the Euroway Industrial Estate in Blagrove. The site also houses several other departments, including Test Groups, BSS Firmware, and non-technical groups. Euroway is an out-of-town site, but the site itself contains some basic facilities that compensate for the distance from town, including a coffee/sandwich bar, restaurant and shop.

BSS Software is based on one large floor of the Euroway building. The office is divided into single-person cubicles, each with whiteboard, cupboard space and a PC laptop with docking station. Most work is done on Dell laptops, on a Windows NT environment. The servers are all UNIX, and most development is done through Exceed PC X software.[3]

Adjacent to the cubicle-space is a large, purpose-built testing lab, with GSM Base Station equipment allowing testing of developed software, investigation of problems. This is required because of the embedded nature of the software. All of the hardware in the lab that our software runs on is produced by Motorola, and the platforms contain a variety of chipsets, including Motorola 68000 series processors.

Work in the lab is done on a variety of platforms, including our laptops, SunOS UNIX workstations, and other specialized GSM testing tools, both in-house and commercial.

THE WORK (PART TWO)

JOB DESCRIPTION

My job title at Motorola was GSM Radio Sub-systems Software Engineer. As explained previously, this involved working in the development of the real-time embedded software that runs in Motorola GSM Base Stations, working in the RSS team.

My responsibilities included:

- Maintenance of legacy software
- Implementing new software features in accordance with the Software Development Lifecycle
- Participating in technical reviews of my own and others work
- Mentoring of new staff
- Preparing and delivering technical presentations
- Content Manager of departmental intranet site
- Development of web pages
- Involvement in process improvement activities
- Development of in-house tools
- Providing technical assistance to other departments
- Representing the company at graduate recruitment events

MAINTENTANCE OF LEGACY SOFTWARE

BSS Software is an extensive system that has been developed by the department over several years. The latest GSM software load is labelled GSR 5, first going through field trials during my placement in 2001.

Upgrading to a new software load is a lengthy, complicated and expensive process. This reason among others means that not all customers upgrade to a new load when it becomes available. As a result, the department is also responsible for maintaining the existing loads already in the field, such as GSR 4. Different software loads will contain different features and functionality, and part of the job is to provide software support for these multiple live software loads.

As well as identifying system defects in-house, defects are often encountered in the field. The role of Software Engineer includes providing customer support to correct these defects. This customer support spans from problem investigation to provision of software fixes.

In addition to correcting defects as they are reported to the department, some situations (such as Field Trials and Software Upgrades) require more formal support. Once experienced with the system, the job involves helping to provide remote support for customers at sites worldwide. Responsibilities include fault debug, generation and support of software patches.

Identifying the defect is a significant part of this maintenance process, requiring overall system knowledge to recognise in which Functional Area(s) the defect is located. From a high-level description of system behaviour, the Software Engineer is responsible for identifying a low-level cause of the problem. Once identified, if the defect is contained within the Engineers' own FA (such as Radio Sub-systems), then the defect can then be investigated, requiring an in-depth, low-level knowledge of the group of processes in their own FA. If the defect is contained within another FA, then it is passed on to an engineer from that FA for investigation. Some problems will include defects in more than one FA, requiring collaboration with engineers in other groups.

Once the defect is identified, and a patch or fix created, it is the responsibility of the Engineer to test the fix, to ensure that the defect is corrected, and that no additional defects have been introduced. This can include testing using a software simulator, and/or GSM hardware in the department's purpose built test laboratory. This requires competence with a wide variety of specialised GSM testing equipment, as well as the ability to build and configure a GSM Base Station system so that it can be used to make and receive calls.

In addition, the Engineer is responsible for maintaining the documentation, making changes to the relevant documents to reflect changes made to the code.

IMPLEMENTING NEW FEATURES

As mentioned previously, the department is responsible for maintaining several live software loads. In addition to identifying and correcting defects, this includes the development of new features and functionality within the legacy code. These are all added in strict accordance with the Software Development Lifecycle – with each new feature following the process of Requirements Analysis (both System and Feature), High and Low Level Design, Coding, and Testing (both System and Feature). Software Engineers are involved in all stages of the Software Development Lifecycle.

An example of a new feature that I was involved in is AMR (Adaptive Multi-rate) – the single most fundamental change to the GSM system since it was first designed. A critical project, this required a wide variety of work and skills, from using high-level system knowledge to help with the identification of feature requirements, to a low level knowledge of the RSS processes to help with the design and coding of the AMR functionality and their RSS impacts.

TECHNICAL REVIEWS

All work produced in the BSS Software department is subject to a process known as the Fagan Defect-Free Inspection Process. This is a formal technical inspection technique, involving peer examination and review of work, which can be code and/or documentation.

The process, developed by Michael Fagan Associates[4], is learnt on an intensive three-day course attended by all engineers in the department. It involves a four-person inspection team meeting to formally review a code or documentation item line-by-line, looking for local errors and possible wider implications and/or problems.

Attending these technical reviews is a part of the job, and requires considerable preparation. Before attending an inspection meeting, it is necessary to thoroughly familiarise oneself with the work being inspected, and the steps taken by the engineer who produced the work. Likewise, engineers must attend technical reviews of their own work, which requires preparation sufficient to enable the engineer to present their work to a group of their peers.

MENTORING

Mentoring became a significant part of the job, particularly in the last third of the placement year. The BSS Software department has a formal mentoring scheme for new starters, which involves each new starter being assigned to an experienced engineer to act as mentor.

Mentors are responsible for training the new starter in all aspects of the job. This includes familiarising them with the GSM architecture, the BSS Software system, the structure of the code and the equipment in the lab. The GSM system is quite complex, and it is the mentor's responsibility to help the new starter learn how it works. Where formal technical training is required, the mentor helps to identify the training needs and match these up with available courses.

As well as technical training, the mentor helps to familiarise the new starter with the company and the department, as well as the systems and procedures used in the department.

In addition to formal mentoring responsibilities of experienced engineers, I was also responsible for training summer placement students working in the department on 8-week projects.

TECHNICAL PRESENTATIONS

As BSS Software is a large and complex system, there are many features and areas of functionality that periodically require research and investigation. Prior to investigating a fault or defect raised either in-house or from a customer, it is often necessary to research the area of functionality that the defect is located in. This research can involve reading through code, reading documentation, speaking to other engineers and looking up GSM specifications from ETSI[5] (European Telecommunications Standards Institute), or a combination of these.

Once the engineer is thoroughly familiar with an area of functionality, and the defect has been identified and corrected, the engineer is then responsible for preparing and delivering a technical presentation on the feature to the rest of the group. This has several benefits for the department, such as allowing all engineers in a FA to become familiar with areas of functionality with only one engineer needing to spend the time researching the topic. It broadens the knowledge of all the engineers in the group, preventing any engineer becoming too specialised in just a single area of the system. It also allows the engineers in the group to be kept informed of what other members in the FA are doing, promoting teamwork and sharing of ideas and expertise.

WEB DEVELOPMENT

BSS Software maintains an extensive intranet site, used to disseminate information throughout the department, and as a source of reference when carrying out investigation and testing work. In addition to the central department intranet, each FA is responsible for maintaining a mini-site where information and tools useful for that Functional Area can be found.

Developing pages for department and FA intranet sites is a part of the Software Engineer job, with web development training offered to those engineers without extensive experience with HTML and other web technologies such as JavaScript, CGI and others.

Each intranet site and FA mini-site has a Content Manager, responsible for the pages developed by engineers within their area. I was given the role of Content Manager for the RSS site, because of my experience developing web sites for other groups. This involved taking ownership of the existing site, which was a disjointed collection of incomplete and out-of-date pages.

Figure 2 - RSS Intranet Site

My first step was to completely redesign both the style and content of the site, and create tools for maintaining it. My main tasks since have included maintaining content, developing new uses for the site and managing content created by other members of the department. The RSS is now quite large, with over sixty pages, and is widely used within the RSS group. Figure 2 above shows the design of the site that I created.[6]

PROCESS IMPROVEMENT

As mentioned previously, there are over 75 engineers working in the BSS Software department. Some of the code and documentation maintained by the department is also worked on by engineers working at the Motorola site in Arlington Heights, Chicago. The large number of engineers involved in development requires a rigid development process, to ensure that there are no conflicts between tasks carried out by different engineers.

Although the procedures used in the department are quite specific, there is an ongoing Process Improvement programme, which attempts to identify problems with the processes used in development, and to find new, more efficient ways to do things.

Involvement in process improvement activities is part of the Software Engineer job, which is done in many ways, including classifying code and documentation errors found during technical reviews using an Orthogonal Defect Classification – so the point at which the defect was introduced can be identified and, hopefully, avoided in future.

I was also responsible for designing and producing a training course that is now used for all new starts in the Software Department, which explains the different procedures for making changes to the code in the multiple software loads that we support.

DEVELOPMENT OF IN-HOUSE TOOLS

Many of the tools used in the BSS Software department are tools produced by engineers within the department. These include tools created using Tcl/Tk, Perl, Java applets and UNIX scripts. Many of these tools are used by the whole department, others by just a single FA, although some are created by engineers for their own use to help with daily work.

Creation of these tools is a part of the job, both formally and informally. These tools vary in size and complexity, from small Perl scripts used to convert numbers between different bases for use during lab work, to more complex applications for decoding GSM message logs.

ASSISTING OTHER GROUPS

The division of BSS Software into Functional Areas means that many engineers develop a detailed knowledge of the processes in their area, but have only a limited knowledge of other Functional Areas, or of the interaction between them.

This means that part of the job in RSS is to provide advice and assistance to engineers in other groups and departments. This can include explaining functionality of RSS features, diagnosing problems or erratic system behaviour, assisting in the investigation of reported defects, and giving training in the usage of various GSM lab tools.

Other examples of this include helping the test group SITG (Systems Integration and Test Group) to develop tests for RSS areas of functionality. This can include explaining the objective and functionality of features to Test Engineers, and working with the test group to develop effective test cases.

GRADUATE RECRUITMENT

Motorola attends several of the large graduate fairs in the country as part of the overall recruitment strategy. In addition to staff from Human Resources, Software Engineers are also selected to represent the Software departments and answer any questions that potential applicants might have.

I attended the Target Live! IT & Engineering Fair, held at Imperial College, London. Open to all students, not just those from Imperial, the Fair was attended by most of the large high-technology firms in the UK. The work involved talking to graduates about Motorola, and explaining what sort of vacancies were available. It also involved doing a brief assessment on each candidate, both verbally and by looking through his or her CV. I ended up with about 40 CVs, and had to select which people were worth interviewing. I chose six of the people that I had seen, and passed their CVs onto my manager for interviewing.

CASE STUDY (PART THREE)

INTRODUCTION

The role of Software Engineer is a very varied and diverse one, involving many different types of work. As such, it is difficult to describe a *typical* job that I worked on, as I worked on so many different things. Instead, I have chosen a problem that I worked on that can be explained to someone without extensive knowledge of GSM. While not one of the more technically complicated problems that I did, it is indicative of the type of investigative work that I did.

GSM OVERVIEW

To describe the problem, it is necessary to first explain some GSM theory. These descriptions are greatly simplified, and should not be considered a correct technical description of a GSM system.

BTS – Mobile Channel Types (BCCH, SDCCH, TCH)

As explained in Part One, GSM mobiles communicate with "Base Stations". The radio transmitters in *Base Transceiver Stations* (BTS) generate *cells* – an area in which mobiles can receive service from the network. Software running on the BTS manages mobiles in the cell, and controls calls made by them.

Mobile phones and Base Stations communicate using a variety of logical channel types, depending on the type of communication. Three examples of these channel types are BCCH, SDCCH and TCH.

TCH stands for Traffic Channel, and is the channel type used for the transfer of voice and data traffic. The sound of the mobile user speaking is transmitted over this channel. Likewise, the sound of the person they are talking to is received over this channel.

SDCCH is a signalling channel, used for non-voice communication. SDCCH is a stand-alone, dedicated channel, used in such situations as during the setting up a new call, when the mobile and BTS have to communicate the various parameters required for the call.

BCCH stands for Broadcast Control Channel. It is transmitted constantly, acting as a form of repeating beacon for the BTS. It is this channel that allows mobiles to display a signal strength indicator, normally shown on the mobile display as a number of segments or bars. The mobile "listens" to the BCCH and calculates the strength of the signal received.

Cell neighbours and "Sys Infos"

In addition to providing the mobile with a way of determining the signal strength from a particular BTS, the BCCH channel also provides the mobile with a lot of other information.

Information from the BTS is contained in System Information messages, or "Sys Infos", embedded in the channel. These contain information about the cell and the network required for the mobile to use the network.

An example of the information contained in the Sys Infos are "Neighbour Lists" – which provide the mobile with a list of the cells adjacent to the cell which the mobile is listening to.

For example, in Figure 3 below, while in idle mode (phone switched on, but not making a call), the mobile will be receiving Sys Infos on the BCCH transmitted by the serving cell BTS (shown in black). The Sys Info will contain a neighbour list, giving details such as frequency numbers and cell IDs for the cells labelled A, B, C, D, E and F.

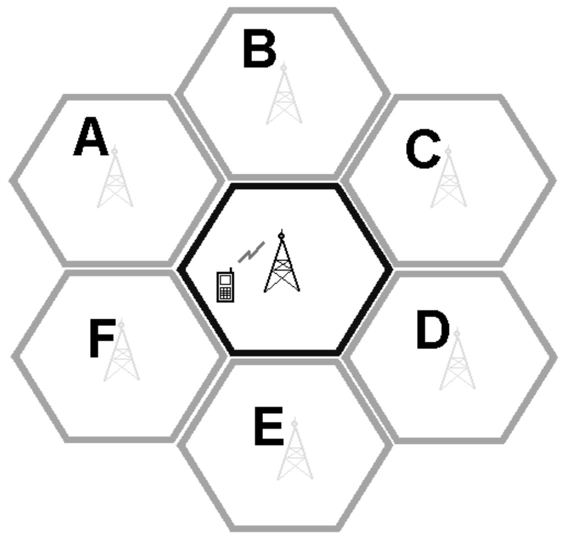

Figure 3 - Cell neighbours

The mobile uses this information to know what cells it can be transferred to if it moves out of range of the serving cell, or if some other reason causes the signal to degrade.

Handovers

One of the features of GSM is Handover – passing an active call from one BTS to another. The portable nature of mobile phones means that the cell used to start a call might not be the best cell to use ten minutes into the call.

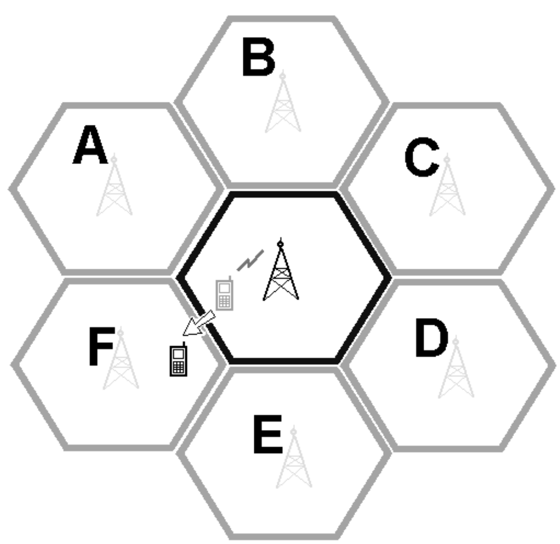

Figure 4 – Handover

As the phone moves further away from the BTS that it is communicating with, and closer to another cell, it is necessary to hand the call over to the other cell, and to do this without affecting the call in any way. The mobile user should remain unaware that anything has changed.

For example, assume that the mobile in Figure 3 above started a call. Then, during the call, the user had moved to the position shown in Figure 4 above. The signal from the original BTS would likely be quite weak, while it could receive a strong signal from the BTS in cell F.

The call will be "handed over" to cell F, with the call and all related information being passed over to the BTS in cell F, and without the mobile user noticing the transfer. Although this involves ending the transmission on the TCH from the original BTS, and establishing on a new TCH in cell F, this transition should be done seamlessly.

Handovers enable mobile users to make calls on the move, without sacrificing call quality.

Measurement Reports

The decision whether to handover a call, and where to hand it to, is made by BSS Software. The mobile should do as instructed by the BTS, and cannot make a decision to handover by itself.

BSS Software makes decisions to handover based on information received from the mobile in "Measurement Reports". These are sent regularly by the mobile (every 480ms) during a call, and inform the BTS of the quality and strength of the signal received by the mobile. BSS Software uses this data in complex Handover and Power Control algorithms.

Measurement Reports also contain information about the signal strength of the neighbours. The mobile measures the strength of the BCCH signal received from each of the neighbours given to it in the neighbour list (described above). It repeatedly checks each neighbour, and the signal strength values are included in the measurement reports sent to the serving BTS.

In this way, during a call the BTS constantly receives information from the mobile telling it how good the signal between the mobile and BTS is, as well as how strong the signal is between the mobile and other BTS's. This allows it to make the decision as to which BTS the call should be handed over to.

For example, if the BTS sees call quality begin to drop significantly, and that the mobile is reporting a very strong BCCH signal from Cell F, it will initiate a handover to Cell F.

Call Establishment

At the start of a new call (or during a handover – effectively a very quick, seamless end to one call, and start of a new one) the mobile needs the frequency of the BCCH on the serving cell, and the frequency of the TCH on the serving cell. This will allow it to transmit and receive as required to make the call, and monitor the strength of the beacon signal from the cell for comparison with neighbours.

It also needs a list of BCCH frequencies of the neighbour cells, in order to measure their signal strength for inclusion in the Measurement Reports.

Power Control

Another feature of GSM is the ability of the phone to alter power usage during a call –known as Mobile Power Control. The principle is that the mobile transmits at a low power when it is receiving a strong signal, and at a higher power when the signal is weak. This dynamic use of power during calls has a number of benefits, including reduced power consumption, which extends battery life. Transmitting at reduced power also reduces radio interference in the area for other mobile users, allowing for increased cell capacity. The aim is to do this without sacrificing call quality, by retaining the ability to increase the mobile's transmission power when required.

This is controlled by the BTS, which sends commands to the mobile to instruct it what power level to transmit at. The BTS decides the power level using the Measurement Reports received from the mobile.

FEATURE: COINCIDENT MULTIBAND

Frequency Bands

In the UK, the GSM system we use is GSM 900 – so called because it operates on a frequency of 900MHz. In some countries, the system used is DCS 1800 – based on GSM, but using a frequency of 1800MHz. However, the two systems are both very similar.

As a result, mobile phone manufacturers soon began to manufacture phones capable of using either frequency band. This enabled mobile phone users to travel abroad with their phone, without having to use a different handset. These handsets are known as MultiBand-capable handsets, or more precisely as either Dual Band (handsets which are capable of using two frequency bands), or Tri Band (handsets capable of using three frequency bands including PCS 1900 for use in the United States).

MultiBand

The GSM 900 frequency spectrum is divided between the mobile phone service providers. In the UK, the frequency spectrum is divided between companies such as Vodafone and Orange, with each company allocated a range of frequencies. This means that capacity (how many mobile phone users a cell can support at once) is a consideration for them – as they are limited to a finite range of frequencies.

As more MultiBand-capable mobiles entered the market, it was a logical progression for network providers to consider using other frequency bands to further increase the capacity of their networks.

Adding DCS 1800 cells to a network requires the addition of new radios that transmit on 1800MHz frequencies (although similar, the technology is different enough to require different RF hardware), as well as paying for the license for the new frequencies. However, in areas where the network use has reached maximum capacity, and where there is no more room for channels in the allotted frequencies, then using an entirely separate frequency band to supplement the network is a consideration.

It allows the operator to increase the network's capacity, by moving MultiBand-capable handsets to the new frequency band, leaving room for more users on the existing band.

Obstacles to MultiBand networks

Setting up cells is a complex and costly process. Each cell requires hundreds of parameters and algorithms to be individually configured, so as not to cause interference with neighbour cells, or to leave areas of poor coverage. As network operators have developed GSM 900 networks over several years, starting this process anew for DCS 1800 is a daunting prospect.

The higher frequencies used by DCS 1800 have different propagation properties, which means that the settings required will be different to a GSM 900 network. In addition, to enable Handovers between the different frequency bands will require extensive reconfiguring of the neighbour lists of cells on the existing 900 network – a lengthy, complicated process.

These reasons made many network operators reluctant to purchase DCS 1800 equipment, despite the opportunities for increased capacity. As a result, manufacturers such as Motorola develop features that make it simpler to add a new frequency band to a network – as it adds another source of sales for the company. One such feature is "Coincident Multiband".

Coincident MultiBand

Coincident MultiBand allows the operator to easily add MultiBand capabilities without complicated reconfiguring of an established network. The principle is for the new frequency band to compliment, rather than alter, the existing infrastructure. The new frequency cells are configured with the same cell boundaries established for the old frequency band, forming an additional "layer" of coverage.

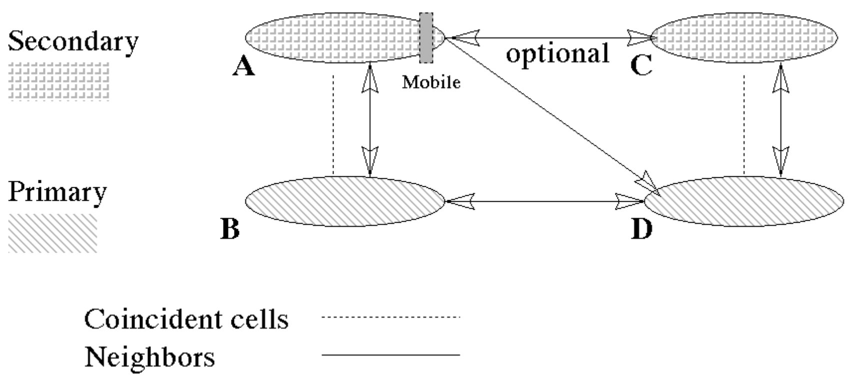

Figure 5 - Coincident MultiBand

Figure 5 above shows a side-on view of the structure of a coincident MultiBand arrangement, with the existing GSM 900 cells (B & D) shown as the lower "Primary" layer, and the new DCS 1800 frequency cells (A & C) shown as the higher "Secondary" layer.

The secondary cell has all the same neighbours as the primary cell as well as the co-located primary cell as a neighbour. The secondary cell may also have other secondary cells specified as neighbours.

For example, the new secondary layer cell A has cell D as a neighbour (the same as it's coincident neighbour B does), as well as other secondary layer cell C. Cell D does **not** have cell A as a neighbour – so extensive reconfiguring of the existing cell parameters is avoided.

During calls on the new band, Measurement Reports are still made based on cells in the original frequency band. This allows the mobile to be handled as if it were on the primary frequency band, while not using primary frequency band channels.

For example, a mobile using a TCH on Cell A, will measure the strength of Cells B and D (because they are defined in the neighbour list). When Cell A receives the Measurement Report from the mobile, the BSS will treat the BCCH strength of the coincident neighbour Cell B as the strength of the serving cell when deciding if a handover is needed.

To summarize, the BSS uses the signal strength reports of cell B from the mobile to make a decision as to whether a handover is needed. It uses the signal strength reports of cell D from the mobile to determine if the neighbour is a viable candidate for the needed handover.

Coincident Handover to unreported neighbour

Figure 6 shows the Coincident Cell Handover. If the BSS decides that cell D is a viable candidate for a Handover for a mobile occupying a TCH on Cell A, it will detect that Cell C is a coincident cell of Cell D, and will redirect the handover to cell C. This is done because the secondary layer is the preferred band, leaving the primary band available for mobiles without MultiBand capabilities.

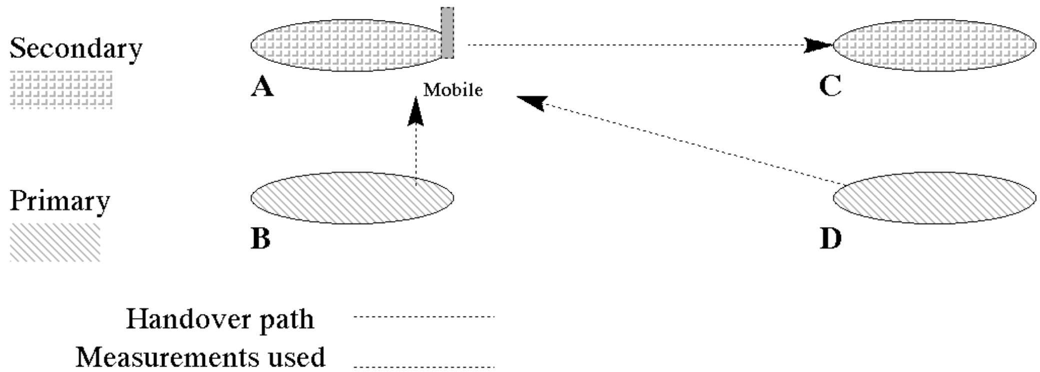

Figure 6 - Handover to unreported neighbour

THE PROBLEM

Tele2 Sweden[7] is a GSM network operator who uses Motorola BSS equipment for their GSM infrastructure. During my placement, they upgraded to a newer version of BSS Software. Part of this upgrade included the introduction of the Coincident MultiBand feature explained above.

After the upgrade, their customers began to report problems with the network – an increase in the number of dropped calls (calls where the mobile user is cut off before the end of the call), and a decrease in the audio quality during calls.

Initial investigations showed an increase in average number of handovers per call. Handovers are detrimental to call quality. They are a last resort, used after all other techniques (such as Power Control), have failed to improve quality and when not to Handover would lead to the call being dropped. This increase would partly explain the reported poor audio quality.

BSS statistics also showed a decrease in the average uplink signal strength – the signal from the mobile to the BTS. This would also explain the poor audio quality, and the increase in dropped calls.

As the algorithms controlling Handovers and Power Control are contained in the Radio Sub-systems group of processes, this problem was passed on to me for further investigation.

INVESTIGATION

My first step was to look at the changes made in the version of BSS Software that TELE2 had upgraded to. The problems had begun to occur soon enough after the upgrade to suggest that the upgrade was the cause. I looked at the major changes, and listed those that might have some effect on Handover or Power Control functionality – which included Coincident MultiBand.

My next step was to produce filters to give to the customer. Filters are a debugging tool that can be used to selectively trap and display internal system information and interprocess messaging. I produced filters to examine what was happening in the features and areas of functionality that I had identified as having changed significantly. Once collected, the filter logs were sent to me for analysis.

One of the first things that I noticed was that RSS was making a lot of Handovers because of poor uplink signal strength – RSS can decide that a Handover is required for any one of a number of reasons, and the logs showed a distinct increase in Handovers made because the signal that the BTS was receiving from the mobile was not strong enough to maintain the call.

This was consistent with the increase in calls with poor uplink signal strength shown in the BSS statistics. However, I could not find any significant changes to the Power Control algorithms. Trying to recreate the problem in the lab was also unsuccessful – calls were successfully managed, with Power Levels altered as required.

I could see no reason for an increase in the number of calls where the signal received from the mobile was very weak – so my next step was to look more closely at what RSS was doing, or trying to do, about the poor uplink signal strength. Again, I supplied filters to the customer that would display the internal information and messaging, this time to display the Power Control algorithms data.

As explained previously, the strength of the mobile transmission is controlled by the BTS, which uses complex Power Control algorithms to determine the optimum power level. The mobile informs the BTS of the power level it is currently using by including it in Measurement Reports. If the BTS determines that a change is required, it sends an instruction to the mobile to make the adjustment.

The filter output showed two significant things – firstly, measurement reports received from mobiles with poor reported uplink signal strength showed that they were not transmitting at their maximum power. In other words, they were transmitting at a level not powerful enough to reach the BTS with a strong signal, while they could have used more power to transmit a stronger signal.

Secondly, RSS was correctly sending Power Control commands to the mobiles, instructing them to increase their transmission power. Measurement Reports received from the mobile showed that these commands were ignored, with no change to the mobile transmission strength. While this explained the increase in dropped calls, the increase in number of Handovers, and the reported poor audio quality, this still didn't give a reason, or suggest how the problem could be fixed.

My next approach was to look at to try and find a pattern in when this was happening. Although this was happening in a significant number of calls, it was not happening to every call, and there had to be a reason why some calls were unaffected. I looked through the history of messages for affected calls, and noticed that nearly all of the affected calls were calls that had been passed from frequency band to another in a Coincident MultiBand handover.

This was a turning point in the investigation, in that it was the first time that I was able to link the problem with an area that had changed in the upgrade. From this, I was able to isolate the cause.

PROBLEM TECHNICAL BACKGROUND

When the GSM specifications were first defined, provision was made for several different classes of mobile – to allow for mobiles of varying sizes. Table 1 below is an excerpt from the GSM specifications[5], showing the 5 different power classes for GSM mobiles, from "4" and "5" for small hand-held units to "1" for large, powerful transmitters such as those found in car-mounted phones.

Power Class	GSM-900 Maximum peak power	DCS-1800 Maximum peak power
1	20W (43dBm)	1W (30dBm)
2	8W (39dBm)	0.25W (24dBm)
3	5W (37dBm)	4W (36dBm)
4	2W (33dBm)	
5	0.8W (29dBm)	

Table 1 - Power Class Definitions (taken from GSM 05.05)

Note that DCS-1800 mobiles are classified using a different scale, due to the different characteristics of the higher frequency. A MultiBand-capable mobile has a different class for each frequency band.

During the signalling at the start of a call, the mobile informs the BTS of its power class, and the BTS refers to a table such as the one above to determine the maximum power level that it can instruct the mobile to transmit at.

Table 2 below shows the format of the instructions sent from the BTS to the mobile for Power Control, taken from the GSM specifications. The Power Control commands are encoded as a "Power Control level" as this is both more efficient and allows for consistency in instruction formats.

For example, when on a GSM-900 call, a Power Control instruction from the BTS containing a Power Control level 14 is an indication to the mobile to transmit at 15dBm. The mobile refers to an internal table such as the one below to interpret the command, and alters its power consumption accordingly.

In addition to the mobile's maximum, every cell has it's own maximum that it imposes on mobiles while they are in the cell. For example, a cell might want to limit mobiles entering it to using a relatively low power, to reduce interference caused for neighbour cells. This maximum is given to the mobile at the start of the call, in the format of a Power Control Level as shown in Table 2 below.

Power Control Level	GSM-900 Power (dBm)	DCS-1800 Power (dBm)
29	-	36
30	-	34
31	-	32
0	43	30
1	41	28
2	39	26
3	37	24
4	35	22
5	33	20
6	31	18
7	29	16
8	27	14

Power Control Level	GSM-900 Power (dBm)	DCS-1800 Power (dBm)
9	25	12
10	23	10
11	21	8
12	19	6
13	17	4
14	15	2
15	13	0
16	11	-
17	9	-
18	7	-
19	5	-

Table 2 - Power Control level to Power dBm conversion table (taken from GSM 05.05)

THE CAUSE OF THE PROBLEM

My investigation revealed that the TELE2 problem was being caused by Nokia mobiles, when they receive certain kinds of Handover Command. Nokia mobiles seem to use the frequency band of the BCCH given in the Handover Command to determine what power levels to use. This was causing a problem when a Nokia mobile received a Coincident MultiBand Handover (as described previously).

The problem was that during a Coincident MultiBand Handover, the Nokia mobile entered the secondary band TCH given a Handover Command containing a power level of "0". A power level of

“0” indicates the maximum power level to be used by the mobile in the new cell, and in a DCS1800 cell this corresponds to 1W or 30dBm (as shown in the table in Table 2 above).

The Nokia mobile sets a limit on it’s own maximum transmit power, depending on the class of the mobile and the frequency band that it is on. It tries to impose the maximum power as defined in Table 1 above, itself, without relying on the BTS not to order power levels outside the maximum allowed.

However, it is using the frequency band of the BCCH channel given in the Handover Command to determine what this limit it. This means it is using a maximum limit appropriate for a GSM-900 frequency – the table in Table 1 above shows that the maximum level in a GSM-900 frequency band for a class 4 mobile is 2W or 33 dBm.

The mobile therefore “assumes” that it should use a maximum transmit power level of 33dBm. In a GSM-900 frequency, this corresponds to a power control level of “5”, as shown in the table in Table 2 above. Incorrectly using GSM-900 power levels means that the Nokia mobile uses this power level of “5” as an upper limit.

Despite the Handover Command indicating a maximum power level of “0”, the mobile treats this as if it indicates a power level higher than it is capable of, and uses “5” as the maximum power control level.

The problem is that, using a DCS-1800 TCH, "5" actually corresponds to 0.1W or 20dBm. The mobile uses this power level as a maximum, and despite the BSS repeatedly ordering the mobile to set the power level to values up to “0” (30dBm) the mobile rejects this and refuses to adjust its power higher than “5” (0.1W / 20dBm) in the cell.

Preventing the mobile from increasing the power level sufficiently causes poor uplink signal strength. This led to an increased number of handovers, and decreased audio quality up to the point of handover.

SOLVING THE PROBLEM

Most of the work that I have done in RSS has been working with GSM 900 systems. Class 4 mobiles are among the most common, and having spent a lot of time working with logs of Measurement Reports, I am used to seeing the mobiles reach a maximum power level at level 5. I recognised this in the Measurement Reports from the mobiles in the DCS-1800 cell, which provided the connection with GSM-900 levels.

The problem being limited to Nokia mobiles explains why I was unable to reproduce the problem in the lab, as most of my testing is done using Motorola handsets. This was identified after further study of the BSS statistics. Further testing revealed that this problem did not exist in mobiles from other manufacturers.

Although my investigation showed that BSS Software was not causing the problem, this was not a solution. Nokia are the market leaders in mobile handsets in Sweden, which means that many TELE2 customers would be affected if the problem were allowed to persist. TELE2 wanted a solution to satisfy their customers, so I was asked to produce a patch or workaround that would solve this scenario.

However, other than disabling Coincident MultiBand, I could think of nothing that BSS Software could do to remedy this problem. The BTS sends the mobile the power control command, but the Nokia mobiles ignore it, incorrectly believing it to be exceeding the maximum allowed power level. There is nothing more that the BTS can do if the mobile rejects instructions to increase the power level.

So, I was asked to produce a formal Compliance Statement – a technical document stating that Motorola was complying fully with all relevant GSM specifications in this matter, and that the problem was with the Nokia mobiles. An excerpt from this report is included at the end of this Placement Report.[8]

CONCLUSION

It is in Motorola's interest for TELE2 to demand that Nokia correct the problem. Nokia are one of Motorola's largest competitors, and correcting this would likely be a costly and embarrassing situation for Nokia. If they had to resort to doing a recall on affected models, they would be faced the bill of replacing hundreds of thousands of phones.

However, TELE2 simply wants their network to work to the satisfaction of their customers. Even though the problem was not Motorola's fault, a large proportion of TELE2 customers use Nokia phones and would be unhappy at the inconvenience a product recall would cause. It could be argued that the simplest option would be for TELE2 not to use Coincident MultiBand. Standard Handover Commands use BCCH and TCH frequencies in the same frequency band, so it doesn't matter which channel the Nokia mobiles use to determine power levels (which is most likely how the problem was created, and went unnoticed, in the first place).

This would be in Nokia's interest. It would save them the cost and embarrassment of dealing with the problem, and would be costly to Motorola, who made a significant investment in developing Coincident MultiBand. TELE2 were one of the first to trial it, and if they were to decide not to use it, future sales of it would be severely affected, as other network providers followed suit.

The strongest point in Motorola's favour is that BSS Software complies exactly with GSM Specifications. The Compliance Statement that I produced was passed to senior management, who presented it to TELE2 in a meeting in Sweden. However, even this is far from being a guarantee – there is precedent for multinationals such as Nokia putting pressure on ETSI[5] to change the GSM Specifications so that Nokia mobiles conform to specifications.

This problem has now left Development, and is in the hands of senior management and the corporate lawyers. It is likely going to be some time before this is resolved – long after I have left. Success for Motorola would mean one of their largest competitors having the cost and embarrassment of fixing perhaps millions of handsets, as features such as Coincident MultiBand are rolled out elsewhere around the world. Success for Nokia would mean Motorola losing the sale of valuable features to both TELE2 and future customers, as well as the cost of wasted development time.

FINAL NOTE

While this was not the most complicated or technically complex problem that I worked on during my year at Motorola, I decided to use it as my case study for a number of reasons. Firstly, I believe the combination of skills used in working on this problem is indicative of the sort of work that I did while at Motorola. Secondly, it was one of the few problems that I felt could be explained to someone without any previous knowledge of GSM. A lot of the work that I have done is quite difficult to explain at a high level without requiring a lot of background knowledge in telecommunications and GSM in particular. Finally, it is an example of some of the high-profile work that I was fortunate enough to have been given, and the sort of responsibility that I had.

BENEFITS (PART FOUR)

OVERVIEW

One of the most positive aspects of my placement was the wide variety of work involved. The diversity of different tasks made for a very interesting work environment, which allowed me to develop a variety of skills – it wasn't simply a programming or coding job.

Examples include the **training and development** skills required for mentoring, the aesthetic and **user interface design** skills involved in the web design and tools development, the **inter-personal and communication** skills required in collaborating on the code maintenance involving more than one FA, the **design** and **technical coding** skills involved in code maintenance and implementing new features, and the **research and presentation** skills involved in preparing technical presentations for other in the department.

Even within the code maintenance and fault investigation work, the wide variety of features and areas of functionality contained within GSM means that the work was always different, and I was always learning something new. The combination of **research, investigative and technical** work made for an interesting and educational job.

In addition, the **formal software development procedures** that I learnt while working for Motorola will be of great value in any future software development that I do – it taught me a lot of good practices and techniques.

Helping with graduate recruitment was also a useful and valuable experience. It was interesting to see a graduate recruitment fair from a recruiter's point of view, having been to several as a student. It gave me a different insight into the process, in particular regarding what makes an effective CV, having had to read through so many.

ACHEIVEMENTS

I feel that I have achieved a lot during my placement at Motorola. Notable achievements include being assigned to **mentor a new starter** to the department – a responsibility normally given to experienced engineers, being made the **Content Manager** for the RSS intranet site, being assigned as **support engineer** during the Field Trials in Hunan (China) and Malaysia, and being involved in the AMR project – a critical and **high-priority project**.

CONCLUSION

In conclusion, I believe that my year at Motorola has been very valuable. In addition to learning a lot of technical knowledge about GSM and mobile telecommunications, I have had the opportunity to develop many different skills, both technical and social

[1] "About Motorola" – Motorola corporate web-site (http://www.motorola.com)

[2] Whatis.com – statistics correct as at August 2000 (http://www.whatis.com)

[3] "Exceed" – Software produced by Hummingbird Corporation (http://www.hummingbird.com/products/nc/exceed/index.html)

[4] Fagan Defect-Free Inspection Process (http://www.mfagan.com/)

[5] ETSI – a non-profit organisation responsible for developing technical standards for technologies such as GSM. A full description of the organization can be found on their website (http://www.etsi.org/)

[6] The image is blurred to obscure Motorola Confidential Proprietary information.

[7] Tele2 Sweden is the Swedish arm of the international mobile phone service provider Tele2 AB, whose corporate website can be found at http://www.tele2.com

[8] The Motorola Handover Command issued for the inter-band handover is within given specification and should be decoded by the mobile to perform the handover, as well as setting its power correctly.

As an example, the following HO command is used for reference.

```
6 2b 10 5d a 0 5d 2 2 d8
```

Which is decoded as follows:

```
6 - Radio Resource procedure
2b - Handover Command
10 30 - Cell description
BCCH frequency is ARFCN 48 (0x030) - PGSM
0d 02 10 - Description of first channel
ARFCN 528 (0x210) -DCS
4 - HO reference
7 - Power command
d8 - synchronization
```

According to the following GSM specification references this command should be decoded and understood by the Nokia mobiles.

GSM 04.18 (version 8.4.0)

3.4.4.4 – Handover Procedure: Abnormal cases

> "A HANDOVER COMMAND message sent to a multi band mobile station shall not be considered invalid because it indicates target channel frequencies that are all in a different frequency band to that of the ARFCN in the Cell Description IE."

This shows that the target channel frequencies may be in a different band than the BCCH frequency specified in the Cell Description.

In the above example the BCCH frequency is 48 (0x30), which is PGSM, while the channel frequency is 528 (0x210), which is DCS. The above excerpt from GSM 04.18 shows that this is valid.

GSM 05.08 (version 8.4.0)

4.2 – RF Power Control: MS implementation

> "The power control level to be employed by the MS on each uplink channel […] is indicated by means of the power control information sent either in the layer 1 header of each SACCH message block (see GSM 04.04) on the corresponding downlink channel, or in a dedicated signalling block (see GSM 04.08). […]
>
> The MS shall employ the most recently commanded power control level appropriate to each channel for all transmitted bursts on either a TCH (including handover access burst), FACCH, SACCH or SDCCH."

This is suggesting that the MS should use the power control information received with reference to the channel rather than the BCCH frequency. Therefore, if the channel is DCS and the BCCH frequency is PGSM the power control level must be interpreted as per DCS 1800.