RECONFIGURATION MECHANISMS AND PROCESSES IN RMA CONTROLLED SOFT-RADIOS

Stoytcho Gultchev, Klaus Moessner, Rahim Tafazolli
Mobile Communications Research Group
CCSR, University of Surrey Guildford, Surrey, GU2 7XH
(S.Gultchev,K.Moessner,R.Tafazolli@surrey.ac.uk)

ABSTRACT

The number of configurable hardware platforms and adaptive or parameterised platform solutions is constantly rising, even the long neglected aspects of reconfiguration of higher layers of the protocol stack have finally found entry into research projects. On the physical level, any functional part within a communication node can be identified as a module; reconfiguration of such module may affect the functionality of others, neighbouring functional parts within reconfigurable radios. The same applies for the upper levels, where the protocol stack, with its stratified structure, provides also a modular scheme that isolates neighbouring modules. Despite such abstractions, configuration and reconfiguration of future commercial Software Defined Radio Terminals remain being complex tasks and procedures, numerous parameters will influence reconfiguration procedures and programmability of terminals, additionally they open the possibility for un-wanted and unsolicited radio configurations. In this paper some of the issues related to reconfigurability are investigated: it highlights possible security threats to reconfigurable communication systems and describes an approach, based on the RMA, that will ensure the correct functionality of communication networks during reconfiguration processes.

1. INTRODUCTION

Any reconfigurable nodes can be expected to have, intrinsically, the ability to accommodate innumerable possible configurations enabling access to both terrestrial and satellite communication networks. These configurations will be implemented purely by application of different combinations of radio configuration software. A management architecture capable to prevent mis-configurations that may occur and also to ensure reliable reconfiguration of SDR terminals is required. This paper describes research pursued towards the design aspects and implementation of processes and mechanisms for a reconfiguration management architecture for 'Mobile Soft Terminals'. The processes and mechanisms necessary to implement controlled and managed reconfigurations in software definable radio equipment. Furthermore, it highlights possible threats to reconfigurable communication systems and describes an approach that will ensure the correct functionality of communication networks during reconfiguration processes.

A brief description of reconfiguration issues, management, control and requirements will be followed by a description of the mechanisms employed in the Reconfiguration Management Architecture (RMA) to implement reconfiguration procedures in reconfigurable equipment.

2. SOFTWARE RECONFIGURABLE TERMINAL

Research into software radio technology is coming to age [1] where the number of configurable hardware platforms [2] and adaptive or parameterised [3] platform solutions are steadily rising.

Configuration of such software radio platforms takes place during system initialisation (i.e. boot-time) however, to make use of the programmable features of such platforms, the matter of 'how to exchange or replace single functional modules during run-time' has to be considered.

The differences between dissimilar configurations may range from, comparatively, minor changes in the RF part (e.g. use of different frequency bands, like GSM 900/1800) to a complete reconfiguration from one access scheme to another (e.g. GSM to IS-95). Accordingly, reconfiguration procedures may alter one particular module or may affect all entities within a terminal, including all protocol stack levels from application to physical layer (i.e. the radio processing platform) [4]. This may require introduction of additional signalling channels as well as an expanded network infrastructure to support the reconfiguration mechanisms. This diversity of reconfiguration scenarios requires strong management functionality to control the actual reconfiguration procedures within the terminal [5].

2.1. Network Reconfiguration Control

Reconfiguration of communication systems is not only a matter of the air interface between terminal and base station,

reconfigurations also affects the network nodes along communication paths. This requires that the network has to be aware of the terminal configuration and its current settings at any time. To ensure standard compliance of intended configurations, a responsible network authority (e.g. mobile an fixed network provider/operator) needs to be able to interact with any Software Radio Terminal requesting reconfiguration [6] and if necessary to prevent such reconfiguration.

2.2. Reconfiguration Management

Legacy radio systems support user-, control- and management planes, however this does not suffice for the support of the additional requirements reconfigurability introduces; a new 'reconfiguration (support) plane' is needed. This additional functional plane is being realised in the Reconfiguration Management Architecture, see figure 1. RMA has been designed to enable, manages and support secure and reliable reconfiguration of terminals and network nodes, and also to facilitate the download of trusted and approved reconfiguration software. The main objective of the RMA is to prevent unsolicited radio access scheme configurations in Software Radio Equipment [7].

2.3. Radio Standard Compliance

In current commercial implementations of terminal and network equipment, the manufacturers are responsible that the compliance to the radio access standards of their terminals/network nodes is ensured. However, with the future provision of open programmable and reconfigurable radio platforms (i.e. based on SDR technology and defined using object oriented design methodologies as described in [1]), the responsibility may as well shift to other parties involved.

The design and provision of reconfiguration software through third party vendors and therein the possibility to combine various (software) implementations, from different sources, within one radio configuration requires the introduction of management and control mechanisms that allow the enforcement of defined rules which regulate the configuration of any software radio platform and ensure the compliance to given radio access standards. These rules have to cover three main subjects: a) standard compliance of the reconfigurable equipment, b) definition of the contractual limitations between subscriber and operator and c) the manufacturer specific reconfiguration constraints for a terminal platform [7]. The distributed reconfiguration management scheme (i.e. the RMA) uses this rule-based approach to implement the shared responsibility for reconfigurations, and to control the boundaries of reconfigurability in future open programmable radio platforms. The Reconfiguration Management Architecture (RMA) implements the rules by describing the structure of the intended configuration using a 'configuration tag-file'. This tag-file then implements the intended configuration in the configurable radio part of a terminal (see later section "Handling the Tag-File"). Figure 1 depicts the RMA and its functional parts as well as the various modules and their distribution between network and reconfigurable terminal.

3. RECONFIGURATION MANAGEMENT ARCHITECTURE (RMA)

Reconfigurable terminals have to perform many additional tasks stretching beyond mere transport related signalling, call control, connection management, etc.. To implement these additional tasks, the RMA consists of a number of functional parts distributed across the network, these parts complement the associated RAT architectures, and handle all issues related to terminal reconfiguration and basic connectivity for reconfigurable terminals.

Reconfiguration management and its related signalling may require additional signalling channels as well as expanded network infrastructure [8]. Eventually this requirement may lead towards the introduction of a universal/global radio control channel as presented in [10]. A reconfiguration support plane, as defined in the RMA, will ensure not only secure and reliable reconfiguration of terminals and network nodes. Moreover, it (i.e. the 'Reconfiguration Management Plane') provides the means for a trustworthy and secure software download.

Situations which require terminal or network reconfiguration may occur at any time, triggers for this may be changes of network conditions and coverage or alternating application requirements, like varying bandwidth, QoS provision (or demands) etc. Therefore, it is inevitable that a configuration managing entity continuously oversees possible requests from the network and monitors the terminal conditions. The reconfiguration manager works alongside and interacts with all other parts of the reconfigurable terminal (i.e. network node) and other network nodes.

Terminal reconfigurability requires reconfiguration management that considers both the mobile station and also corresponding management units within the network. The parts of the RMA located within the network are responsible for the coordination of the configurations of single nodes and also for the approval of anticipated terminal configurations. The main objectives and tasks for the RMA are:

- To enable full or partial reconfiguration of all protocol stack layers;
- Control and monitoring of the configurations of network nodes;
- Control and management of reconfiguration processes at both terminal and network side.

Reconfiguration from a current to a new configuration may be caused by various reasons; three different situations have to be handled:

1. The network node (terminal/base station) has no inherent configuration and a complete, initial, configuration has to be performed (i.e. the manufacturer, network provider or vendor will initiate the configuration of the terminal).
2. The network node is already configured to a radio standard (e.g. GSM) but needs to be reconfigured to a different standard (e.g. UMTS) (i.e. either the user/application or the network may request a reconfiguration, but this requires the agreement of the network provider).
3. The network node is already configured but a minor, partial reconfiguration (e.g. an update or bug-fix) of one or more modules is required.

As aforementioned the responsibility for reconfiguration management needs to be shared between network and terminal, the modules of the RMA are distributed into three major parts: a) Reconfiguration Control, b) Reconfiguration Management and c) Radio Module part.

The 'Configuration Control Part' (CCP) executes those reconfiguration related tasks that affect the network or air interface and therefore require the approval of the responsible authority. It provides the means to host the authority responsible for reconfiguration of terminals, the tasks of the CCP include:

- reconfiguration software download provision and negotiation,
- evaluation and approval of intended configurations, using a virtual configuration process,
- assurance of standard compliance,
- monitoring of configurations throughout the network,
- provision of configuration rules for different reconfigurable radio platforms,
- registration of the current/new configuration.

To perform these tasks, there are a number of functional modules within the CCP: the AcA-Server performs most of the aforementioned tasks, it monitors the configurations of the network neighbourhood, manages the registration of terminal configurations, handles the software download, validates new configurations by executing a virtual configuration (VC) procedure and ensures the adherence to given standards. Physically, the CCP may be located either within the access or even the core network. 'Rules&Policies' is a tool used by the network provider to specify certain platform dependent parameters and reconfiguration policies of the network provider. The 'SW-Store' is a database hosting approved configuration software and the terminal configuration register.

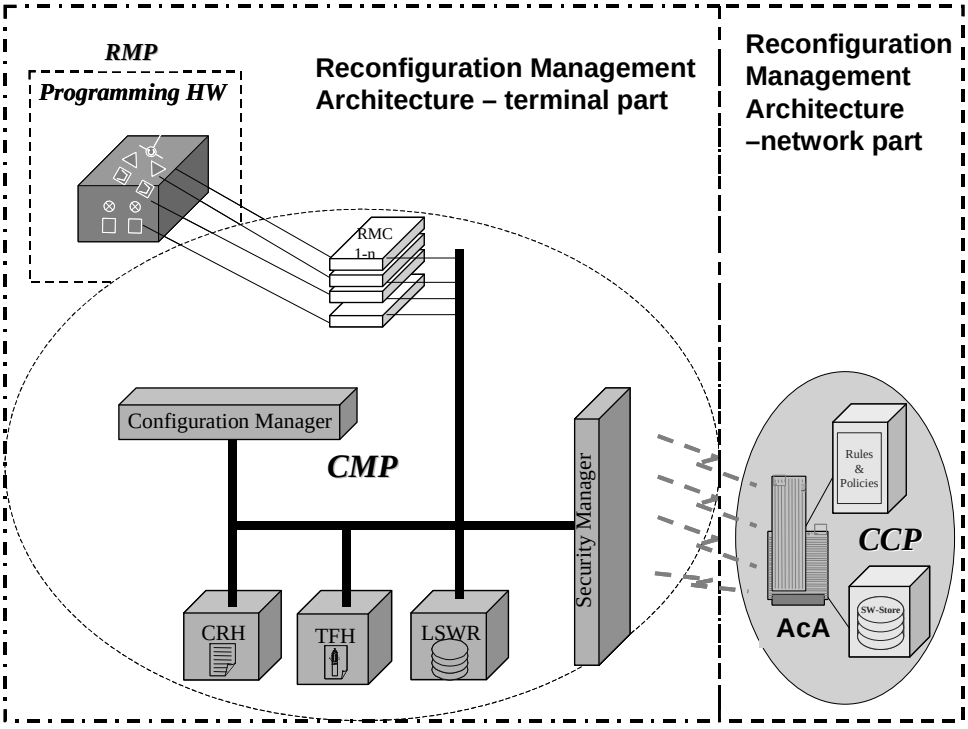

Figure 1 Reconfiguration Management Architecture [9]

Across the air interface, there are two parts within the reconfigurable terminal, a) the 'Configuration Management Part (CMP)' coordinating the configuration and reconfiguration processes of b) the configurable 'Radio Module Part' (RMP).

Tasks of the Configuration Management Part (CMP) include the procurement of configuration software, handling of configuration rules, generation and compilation of tag-files, implementation of new configurations and finally reconfiguration related signalling.

The CMP contains a number of functional modules, firstly the 'Configuration Manager', which manages the communication between the modules within the CMP and also the signalling between CMP and CCP. A variable number of 'Reconfiguration Management Controllers' (RMC) acts as interfaces between the managing domain (i.e. the CMP) and the radio execution domain (i.e. the RMP).

The RMCs implement the actual configuration of radio modules within the RMP[1], whilst the Configuration Manager (CM) controls and coordinates the whole of the reconfiguration process. Other modules within CMP are a Local SoftWare Repository (LSWR) (to store radio configuration software), a Configuration Rule Handler (CRH) (this unit maintains the list of rules for reconfiguration, these rules depend on policies set by the network provider and also on the terminal type), a Tag-File Handler (TFH) (to store, interpret, generate and alter tag-files) and a Security Manager (SM) (responsible for establishment, maintenance and termination of secure connections between the different management and control units and to prevent malicious reconfiguration requests and tampering during the download of reconfiguration software). A configuration software bus, based on CORBA[2] facilitates the transport between the modules within elements; it carries the signalling traffic between the distributed reconfiguration management and control parts. The functionality of the 'security manager' is required to ensure secure, trusted and authorised exchange/download of reconfiguration information and of configuration software between different parts of the architecture, see [11].

A reconfiguration process (i.e. reconfiguration of the RMP, controlled by the CMP) consists of three major steps that need to be completed for the terminal to be able to apply its new configuration. These stages are:

1. Download of rules and software – respectively CRH and LSWR;
2. The creation of a tag-file (n.b. a tag-file contains the 'blueprint' of the terminal created by the Tag-File Handler (TFH) module);
3. The validation of the intended terminal configuration by the CCP (i.e. the verification, within the network/AcA server, of the configuration described in the tag-file);

The implementation of the radio modules in the RMP, relying on the SW structure (i.e. radio blueprint) defined within the tag-file, is done by the Reconfiguration Module Controllers (RMC).

3.1. CRH-managing rules and configuration policies

Within the configuration management part of a reconfigurable terminal, the CRH stores and handles a list of agreed reconfiguration rules (i.e. these rules are agreed to ensure that the terminal will eventually comply to the existing radio standards, operator needs and user requirements), there are various aspects to be considered when defining rules for reconfigurable platforms, some of these rules depend on parameters defined within the operator/user contract whilst others relate to the platform specifications provided by the terminal (equipment) manufacturer. This latter type of rules, whilst depending on the parameters of the reconfigurable platform, will be defined and set by the network provider/operator and may vary from one configurable terminal type to another but as well from one user to another.

There are three categories of reconfiguration rules to regulate different aspects in reconfigurable equipment; the list of rules defines the degree of permitted changes to the current configuration (i.e. this may range from minor changes such as bug-fixes, to complete reconfiguration of the configurable radio). These 'rules categories' can be divided into: General-, Tag- and Software- Rules. Each of these categories defines a restricted set of limitations to be considered during the configuration of a terminal, the rules directly influence the definition of a terminal configuration by directing the generation of the configuration tag-file.

The General rules define and retain the policies of service provider, operator, and manufacturer. These policies apply for all configurations of any available reconfigurable terminal type/platform. The set of general rules also contains a number of requirements/restrictions (e.g. maximum permitted price for a reconfiguration) given by the terminal user (i.e. defined in the contract between provider and user). The issues and topics represented within this type of reconfiguration rules implement three different policies, these include:

- Operator Policies – i.e. band and permitted use of bandwidth, contractual permissions and agreements regarding reconfiguration, reconfiguration restrictions and permissions, etc.
- Personal Policies – are the contractual restrictions a user may require, this includes the definition of maximum permitted costs for a terminal reconfiguration, etc..
- Software Policies – are set by the manufacturers and define platform specific restrictions, security issues and installation details for reconfiguration software modules.

While the general rules rely on the three policies, tag rules specify the way a terminal has to create the (target) tag file (i.e. the blueprint of the intended software architecture to be implemented on the programmable terminal platform). Tag rules are defined and created by the operator and are stored within the terminal. Once stored, tag-rules will only be replaced if the terminal hardware becomes upgraded (e.g. memory enlargement, etc.). Within the network, the tag rules are created by AcA server based on the specifications of the network operator and the terminal specifications (i.e. manufacturer, platform, hardware settings, etc.).

Software rules will be provided by the software vendors and include information about installation requirements for a

[1] It can be envisaged that possible external extensions to reconfigurable terminals may also be reconfigurable, for which case the architecture foresees the introduction of *'External Module Controllers' (EMCs)* see also [9].

[2] Alternative platforms like RMI or DCOM, etc. may also be applied.

software module on different hardware platforms. Software rules, in general, define the input/output parameters of modules, and contain installation details about the software.
One of the main responsibilities of the CRH is to store the different rules applicable for the software modules stored in the LSWR to make them accessible via an indexing system, and to provide, during the reconfiguration process, both rules and index information to the Tag-File Handler.

3.2. Handling the Tag-File

The RMA uses abstract descriptions of reconfigurable terminals to validate anticipated configurations; this means that the software structure of a terminals RMP configuration is described in a clear-text script (i.e. using a 'readable' syntax similar to HTML/XML, the use of customised subsets of standardised languages would also be possible) called 'configuration tag-file'. The initial radio configuration of a terminal is described in a "boot"-tag-file; usually this file is accessed and interpreted during the initialisation period of the terminal.
The 'Tag-File Handler' is a functional entity that uses its own process and memory; its main purposes are a) to generate a new tag-file (if a reconfiguration is required) and b) to interpret the boot tag-file (i.e. the 'blueprint' of the complete radio part of the terminal) during the boot sequence of a terminal (i.e. the boot tag-file may contain either an initial or the last known terminal configuration). The tag-file handler itself is involved in every configuration or reconfiguration process of the terminal. During a reconfiguration procedure the tag-file handler fulfils following steps:

1. Creating the source code of a new tag-file. It uses therefore a set of rules, which are obtained from the rule handler. These rules consider both specifications of the terminal platform and user specific requirements.
2. After generation of the new tag-source file, the tag-file handler copies the new source code to its own memory space.
3. The tag-file handler compiles the source code into the actual tag-file and stores this within the memory.

The format of the tag-file comprises following details about a radio configuration:

- identifier for radio module structure,
- identifier for module I/O structure,
- location of the reconfiguration software module code in the local software repository,
- size of the code and physical parameters.
- input and output data/parameter types for each radio-module,
- internal parameters that describe the software installation process.

3.3. Implementing the 'tags'

Reconfiguration Module Controllers (RMCs) are addressing the task of how to interpret the information contained within a tag-file and then to implement the complete configuration of the RMP. Therefore, a dual state machine structure is applied within every RMC, the first of these state machines (i.e. the internal state machine) deals with the software installation on the hardware platform (i.e. on the RMP), whilst the second (external) state machine acts as the RMPs signalling endpoint within the CMP. The configuration manager generates all RMCs according to the specification within the tag-file; this means that, depending on the tag file, 1…n RMCs are created and they implement the RMP configuration described within the tag-file. Major advantage of this dual state machine structure is the complete isolation between reconfiguration management architecture and configurable radio part.

3.4. Implementing a Reconfiguration Process

Implementation of a reconfiguration procedure relies on the mechanisms of the tag-file handler and the reconfiguration module controllers. Once the preliminary signalling (i.e. rules and software download) is completed, the configuration manager requires the actual generation of a new tag-file and performance of the subsequent steps of the reconfiguration procedure. Configuration and software rules are then requested from the rule handler and interpreted by the tag-file handler, which generates and compiles the source code of the new tag-file. In case the compilation fails, this process needs to be repeated. Once the tag-file is successfully generated, it becomes uploaded to the AcA server where its validity and correctness will be evaluated during a virtual configuration (VC) procedure. Assuming this evaluation proves the validity (standard compliance) of the file, the AcA server issues the permission (to the terminal) to implement the new configuration. If the validation failed, the tag-file generation process must either be repeated and possible errors removed or the reconfiguration sequence must be abandoned. During the validation process, the tag-file handler still stores the old tag-file and awaits the acknowledgement about the validation from the AcA server.
The next step of the reconfiguration procedure is then the implementation of RMCs (i.e. they are instantiated by the configuration manager within the CMP). During instantiation, the RMCs obtain references to the tag-file handler; whereby these references are required to retrieve the configuration software modules from the local software repository. The first RMC then installs the software specified (within the tag-file) for its functional radio module and performs a basic function test; the configuration manager repeats this implementation sequence until all

specified RMCs and their radio module implementations are installed on the RMP platform.

Upon completion, each RMC notifies the configuration manager about conclusion of the radio module software installation procedure. The configuration manager then connects the modules and verifies the new configuration, followed by a request to register the new terminal configuration with the network (i.e. the AcA).

4. CONCLUSIONS

This paper discusses the issues of reconfiguration and justified the need for reconfiguration management in configurable mobile network environments. It documents the various tasks a reconfiguration management architecture has to perform to ensure reliable and flawless transformation from one to another configuration. Regulatory requirements will have to be met even when communication equipment is potentially completely definable by the means of configuration software. Especially open platforms provide space for both unintended but also deliberate violation of existing radio access standards which needs to be prevented. The paper shows how the various requirements and dependencies between the parties involved (manufacturer, operator/network provider and user) can be captured in rules for reconfiguration of SDR terminals. The RMA is explained and its elements and their functionalities are introduced. The tag-files describing the software structures for the reconfigurable radio part within a software definable terminal and their generation, handling and actual implementation are explained showing the sequence of events between reconfiguration request and registration of the new implemented configuration. The RMA provides the mechanisms necessary to support reconfiguration of software configurable communication nodes in wireless environments and is in fact a complete proposal for a reconfiguration management schema for secure reconfiguration of SDR equipment in reconfigurable radio communication networks.

5. ACKNOWLEDGMENTS

The work presented in this paper has formed part of Core 2 Research Programme of the Virtual Centre of Excellence in Mobile & Personal Communications, Mobile VCE, www.mobilevce.com, whose funding support, is gratefully acknowledged. More detailed technical reports on this research are available to Industrial Members of Mobile VCE.

6. REFERENCES

[1] Mitola J, “Software Radio Architecture–Object Oriented Approaches to Wireless Systems Engineering”, Wiley-Interscience, ISBN 0-471-38492-5, 2000.

[2] Ajluni C, “One World, One RF Front End”, Wireless Systems Design Magazine, pp. 33-34, July 2001.

[3] Wiesler A, Mueller O, Machauer R, Jondral F, “Parameter Representations for Baseband Processing of 2G and 3G Mobile Communication Systems”, 12th Tyrrhenian International Workshop On Digital Communications - Software Radio Technologies and Services, Portoferraio, Italy, 13-16 September 2000.

[4] Moessner K., Vahid S., Tafazolli R., “OPtIMA: An Open Protocol Programming Interface Model and Architecture for Reconfiguration in Soft-Radios”,12th Tyrrhenian International Workshop On Digital Communication – Software Radio Technologies and Services, Portoferraio – Island of Elba, Italy, 13-16 September 2000.

[5] Moessner K, Gultchev S, Tafazolli R, “Software Defined Radio Reconfiguration Management”, PIMRC, San Diego, California, USA, 30 September -04 October 2001.

[6] Gultchev S, Moessner K, Tafazolli R, “Network Based Reconfiguration Support Services for Software Radio Terminals”, IEE 3G2003, London, UK, 25-27 June 2003

[7] Gultchev S, Moessner K, Tafazolli R, “Controlling Reconfiguration”, 3G2002, London, UK, 8-10 May 2002.

[8] Moessner K, Tafazolli R, “Communication Platform Reconfiguration”, Proceedings of the 2nd WWRF Forum meeting, Helsinki, Finland, 10-11 May 2001.

[9] Moessner K, Vahid S, Tafazolli R, “Reconfiguration Management Architectures”, UK patent application no. 0028463.8, 22 November 2000.

[10] Jamadagni N S S, A minimal signaling channel for SDR download support, submission to the 22nd General Meeting of the SDR Forum, Atlanta, Georgia, USA, 5-8 February 2001.

[11] Gultchev S, Mitchell C, Moessner K, Tafazolli R, “Securing Reconfigurable Terminals – mechanisms and protocols”, PIMRC2002, Lisbon, Portugal, 15-18 September 2002.