Project Cluster

“Reconfigurability”

Vision on Reconfiguration in Mobile Networks

_Draft 5.1, 16.10.01_
## Contributing projects

So far the following projects contributed to the cluster report (names in bold letter are the contact points for the relevant projects):

<table>
<thead>
<tr>
<th>Project</th>
<th>Contact</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAST</td>
<td>Dave Lund</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:DLund@hwcomms.com">DLund@hwcomms.com</a></td>
</tr>
<tr>
<td></td>
<td>Mirsad Halimic</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:Mirsad.Halimic@panasonic-pmdc.co.uk">Mirsad.Halimic@panasonic-pmdc.co.uk</a></td>
</tr>
<tr>
<td></td>
<td>Kambiz Madani</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:madani@cmsa.wmin.ac.uk">madani@cmsa.wmin.ac.uk</a></td>
</tr>
<tr>
<td></td>
<td>Parbhu Patel</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:parbhu.patel@panasonic-pmdc.co.uk">parbhu.patel@panasonic-pmdc.co.uk</a></td>
</tr>
<tr>
<td>DriVE</td>
<td>Ralf Tönjes</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:ralf.toenjes@ericsson.com">ralf.toenjes@ericsson.com</a></td>
</tr>
<tr>
<td></td>
<td>Wolfgang Stahl</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:Wolfgang.Stahl@eed.ericsson.se">Wolfgang.Stahl@eed.ericsson.se</a></td>
</tr>
<tr>
<td>EMERGENCE (KA II)</td>
<td>Ursula Huws</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:analyica@dial.pipex.com">analyica@dial.pipex.com</a></td>
</tr>
<tr>
<td>MOBIVAS</td>
<td>Spyros Panagiotakis</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:spanag@di.uoa.gr">spanag@di.uoa.gr</a></td>
</tr>
<tr>
<td></td>
<td>Maria Koutsopoulou</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:mkoutsop@di.uoa.gr">mkoutsop@di.uoa.gr</a></td>
</tr>
<tr>
<td></td>
<td>Nancy Alonistioti</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:nancy@di.uoa.gr">nancy@di.uoa.gr</a></td>
</tr>
<tr>
<td></td>
<td>Nikos Houssos</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:nhoussos@di.uoa.gr">nhoussos@di.uoa.gr</a></td>
</tr>
<tr>
<td></td>
<td>Vangelis Gazis</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:gazis@di.uoa.gr">gazis@di.uoa.gr</a></td>
</tr>
<tr>
<td></td>
<td>Zahos Boufidis</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:boufidis@di.uoa.gr">boufidis@di.uoa.gr</a></td>
</tr>
<tr>
<td>MOTION (KA II)</td>
<td>Marco Boero</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:marcob@softeco.it">marcob@softeco.it</a></td>
</tr>
<tr>
<td>TRUST</td>
<td>Markus Dillinger</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:Markus.Dillinger@icn.siemens.de">Markus.Dillinger@icn.siemens.de</a></td>
</tr>
<tr>
<td></td>
<td>Thomas Wiebke</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:wiebke@panasonic.de">wiebke@panasonic.de</a></td>
</tr>
<tr>
<td></td>
<td>Nigel Drew</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:Nigel.Drew@motorola.com">Nigel.Drew@motorola.com</a></td>
</tr>
<tr>
<td></td>
<td>John MacLeod</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:John.MacLeod@bristol.ac.uk">John.MacLeod@bristol.ac.uk</a></td>
</tr>
<tr>
<td></td>
<td>Mehul Metha</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:Mehul.Metha@roke.co.uk">Mehul.Metha@roke.co.uk</a></td>
</tr>
<tr>
<td></td>
<td>Tim Farnham</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:Tim.Farnham@toshiba-trel.com">Tim.Farnham@toshiba-trel.com</a></td>
</tr>
<tr>
<td></td>
<td>Ja Garcia</td>
</tr>
<tr>
<td></td>
<td><a href="mailto:Jagarcia@ttinorte.es">Jagarcia@ttinorte.es</a></td>
</tr>
</tbody>
</table>
1. Vision on Reconfigurability ................................................................. 5
   1.1 Requirements and Regulatory Issues ............................................. 6
      1.1.1 Requirements for Future Mobile Communications from the User Perspective .... 6
      1.1.2 Regulatory and Standardisation Considerations .............................. 7
   1.2 Mapping requirements to system concepts and technologies .......... 8
      1.2.1 Creation and provisioning of services over converging networks and different radio
            access modes ............................................................................. 8
      1.2.2 User Environment Management and Distributed Processing Framework ........ 9
      1.2.3 Radio Reconfiguration Control ................................................... 10
   1.3 Constraining Considerations .......................................................... 10
   1.4 Time Table for Future Mobile Networks .......................................... 10

2. Service provider and user requirements ............................................. 12
   2.1 Background and aims ..................................................................... 12
   2.2 IST 1999-Call Projects: Approaches ............................................. 12
   2.3 Results so far .................................................................................. 14
      2.3.1 TRUST ................................................................................... 14
      2.3.2 MOTION ............................................................................. 14
      2.3.3 EMERGENCE .................................................................... 14
      2.3.4 MOBIVAS .......................................................................... 14
      2.3.5 DRIVE ............................................................................. 14
   2.4 Issues to be addressed in the future .................................................. 15

3. Enabling technologies technology requirements on reconfigurable
   networks and terminals ..................................................................... 16
   3.1 TRUST Enabling Technologies ....................................................... 16
      3.1.1 RF Design in TRUST .............................................................. 16
      3.1.2 Design methodology for base band ......................................... 17
   3.2 Drive Enabling Technologies .......................................................... 18
   3.3 MOBIVAS Enabling Technologies ................................................ 18
      3.3.1 Embedded Beans .................................................................. 18
      3.3.2 Processor Expert™ .............................................................. 19
      3.3.3 Reconfigurable terminal design example ................................. 19
   3.4 CAST Enabling Technologies .......................................................... 21
      3.4.1 RF front-end ........................................................................ 21
      3.4.2 Reconfigurable baseband processing .................................. 22
      3.4.3 Intelligent Sub-System .......................................................... 22

4. System Aspects .................................................................................. 24
   4.1 TRUST: System Aspects ............................................................... 24
      4.1.1 Framework for Reconfigurable Terminals ................................ 24
      4.1.2 System and Network Architectures Supporting Reconfigurable Terminals .... 25
      4.1.3 Overview on the system architecture .................................... 26
   4.2 Mobivas: System Aspects .............................................................. 34
      4.2.1 Service provision requirements and management .................... 34
      4.2.2 The need for network/mobile systems reconfigurability management .... 34
      4.2.3 Introduction of open APIs for enabling flexible service provision and systems
            reconfiguration management ....................................................... 35
      4.2.4 Charging in an integrated service provision and reconfigurability management
            framework .............................................................................. 39

5. References ....................................................................................... 42
Executive Summary

The projects contributing to this report are members of a project cluster called “reconfigurability” which aims to support the cooperation among the projects and to develop a vision on reconfigurable systems by describing evolutionary scenarios based on existing-systems and revolutionary scenarios deploying leading-edge technologies. In this report the current research areas of IST projects dealing with reconfiguration are described enriched with the current trends experienced in the research and standardisation communities. The projects give their vision on future scenarios and open issues to be addressed in future EU projects or somewhere else. All research topics inherently bear the open point to what extent reconfigurability concepts need to be standardised and what can remain proprietary, i.e. in the responsibility of a manufacturer, operator or third party. The projects are aware of the ongoing discussions in MExE, SDR-F, OISP (formerly OSA), etc., but until now no standardisation body systematically addresses the questions to be tackled.

The first chapter of this report addresses the vision on reconfigurability and gives an overview on user requirements, regulatory demands, system methodology its and constraints. Concepts for reconfigurability must be compatible to legacy networks and must be embedded into future networks based on IP but also future network topologies as ad-hoc networks must be supported e.g. by reconfigurable routers. The remainder of this report summarises the research areas of contributing projects structured into three research areas which help identify the important points for standardisation or regulatory bodies.

The first research area is to determine the user and service provider requirements for reconfigurability by introducing a methodology how to understand and to promote the demands on such systems. This important input for the complete research field must be understood at first before designing any concepts or developing any technology. For this area a discussion between end user, operator and manufacturer must lead to a common understanding of the real needs.

The second research field is the definition of a world-wide accepted system concept which can be only established by standardisation across the network and terminals. The understanding of an architecture enabling a Distributed Processing Environment [1] and the methods for spectrum sharing, mode detection, seamless service provision by mode switching and the secure software download lead to combination of the currently defined methods known from classical mobile radio design and computer science. This synergy will emerge many novel concepts but also new questions currently not formulated. But only the full understanding of till now more or less independent research fields and the consideration of the real user needs will avoid a failure of the SDR technology.

The third field is called Enabling Technologies for reconfigurability, which addresses the analogue RF domain, but also base band technology and the needed programming languages for the software running on the processor platforms. This field bears many aspects which can be categorised as proprietary, i.e. manufacturer dependent. It is to be noted that programming language itself might be proprietary but its classification according to concurrency, persistence and distributed is to be commonly adopted and in this respect it is a part of the overall system concept.

SDR concepts are part of future networks which may be called “Systems beyond 3G”. But for cost-effectiveness it must be understood that such concepts must be backwards compatible to 2G and 3G systems. An overview on the compelling needs for SDR concepts can be found in [2]. This report on reconfigurability should be later merged with the other project clusters “Systems beyond 3G” and “Smart Antennas”.
1. Vision on Reconfigurability

As we look beyond the Third Generation of Mobile Communications, we can initially perceive the convergence towards an IP-based core network and ubiquitous, seamless access between 2G, 3G, broadband and broadcast wireless access schemes, augmented by self-organizing network schemes and short-range connectivity between intelligent communicating appliances. In this ‘composite radio environment’ where several highly standardized legacy radio transport schemes exist, the medium term goal would be to develop reconfigurable network and terminal techniques to enable interworking and so to deliver diverse and exciting applications using the most appropriate radio access scheme(s). Appropriate in this sense refers to the dynamic choice of access scheme(s) to achieve seamless, uninterrupted delivery to the user, customized to the user needs in terms of content, quality of service and cost. In such an environment, vertical handover may take place between different access systems (cellular layer down to personal network layer, e.g. Bluetooth), combined with real time service and resource negotiations to seamlessly achieve desired quality of service. The interworking, mobility management and roaming would be handled via the medium access systems and the IP based core network.

Reconfigurable systems must be embedded into a network environment composed of legacy PLMN networks and future networks based on IP transport mechanisms as illustrated in the below figure:

![Figure 1: Future Mobile Networks](image)

The exploitation of second generation cellular networks has been essentially based on voice services in spite of their native capabilities to handle several modes of data services. Because of this reason, mobile network operation model has been offered under a close model. Nowadays a revolutionary change of this model may be changing this scenario. Some alternative service capabilities like SMS are growing quite quickly and WAP enabled devices are flooding the market. Probably, the near introduction of GPRS capabilities will foster the use of this kind of mobile services. All that trends are announcing that a radical change in the way to provide mobile services may be ongoing. Reconfigurable radio is a concept that embraces all these new capabilities and extends them to cover more challenging possibilities including the dynamical adaptation of the communication protocol stacks or even the redefinition of the physical layer of the PLMN.

Mobile operators and manufacturers will exploit reconfigurable equipment for three main purposes. First of all, these techniques will allow to reduce the exploitation costs for mobile communications provisioning. This reduction may come chiefly from the capability of adapting the systems to the improved capabilities that are continuously appearing in the mobile arena. Reconfigurable systems may allow, for instance, to provision for newest voice codecs to older terminals enabling the improvement of
the global capacity of the system or the service quality. The second way customers will exploit reconfigurable equipment is related to the need to deploy new networks in a near future to exploit third generation wireless communication systems based upon the UMTS standard whose licensing stage is now being completed all around Europe and whose deployment will be started in a near future. Such deployment process will require quite extensive investments in new equipment whose awarding processes may be strongly influenced by the technical merits of the solutions proposed by the different providers.

Eventually, there is a strategic impact that is related to the positioning of the network operator and service providers in the new value chain. The new service provisioning chain has an open character, and new business opportunities are being anticipated. It is still unclear the market acceptance for such opportunities but the research of IST project dealing with reconfigurability will be very valuable to establish the technical foundations for the service provisioning models and to acquire a rough knowledge of the time that will be required to allow such solutions to leave development laboratories and disrupt the market.

1.1 Requirements and Regulatory Issues

1.1.1 Requirements for Future Mobile Communications from the User Perspective

Given the increasing demand of flexibility and individuality in the society, what does all of this mean to the end-user? Potentially, the value would be in the diversity of mobile applications, hidden from the complexity of the underlying communications schemes. This complexity would be abstracted into an intelligent personality management mechanism, learning and understanding the needs of the user, and controlling the behaviour of their reconfigurable terminal accordingly in terms of application behaviour and access to the supporting services.

TRUST attempts to rationalise this ‘seamless wireless utopia’ by studying the ‘real’ requirements for reconfigurable terminals and creating realistic working scenarios. Technology research will identify the system support concepts, enabling technologies and standardisation required to realise the scenarios, and through subjective evaluation, system modeling and simulation, will evaluate the feasibility of the proposed solutions.

As can be seen, reconfigurability touches the PCS network in many places. Each of the potential ‘features’ interact in a way that affects all users. Rather than presenting a technology utopia, TRUST’s user requirements approach [3] focuses on the study of ‘real’ early adopter users and operators. Throughout the project these user groups will be used to generate real usage scenarios and requirements.

Derived from early TRUST qualitative work [3], [4], Table 1 shows the dominant high level requirements for reconfigurable radio from the perspectives of users, application/content providers, service providers, network operators and equipment manufacturers. Based on these requirements, the demands on the end-to-end system concepts and corresponding enabling technologies may be derived.

<table>
<thead>
<tr>
<th>END USER HIGH LEVEL REQUIREMENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ubiquitous mobile access</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Quality expectations vary with task</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Ease of access to applications and services</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Low cost and relevant services and meaningful billing</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Technology comfort</td>
</tr>
</tbody>
</table>

END USER HIGH LEVEL REQUIREMENTS
User-friendly handling of delays, disconnections and new connections via meaningful feedback to the user;
Transparent handling of version/configuration control for application and system software (including radio access stack software);
Accountability of system to user for reconfiguration changes.
Support expected from the service provider and operator in finding services and updating software.
Intelligent use of battery resource, both locally (local application, display, sound) and in network access
Simple UI and appealing aesthetic.
Reasonable equipment life
Expectation that terminal equipment will offer support for fast-evolving complexity and diversity of applications and services

APPLICATION DEVELOPER/CONTENT PROVIDER
Common Execution Environment
Allowing development of applications and associated content independently of underlying network services and terminal capabilities; self-configuration via capability exchange

Application Diversity
Terminals capable of supporting fast-evolving complexity and diversity of applications and services;
Utilisation of increasing terminal resources to enrich application (eg spare DSP processing capacity)

SERVICE PROVIDER
Fast, open service creation, validation and provisioning
Allowing development of services independently of underlying network services;
Provisioning of validated services configured to underlying network and terminal capabilities

Inform user of services available
Requirement for an effective scheme to ‘advertise’ available services in a service discovery negotiation

Maintain connections and adapt to required QoS
Ability to seamlessly switch connections to alternate radio access schemes or alternate network operators both in call and in standby
Dynamically modify resource allocation to maintain desired QoS over radio channels

NETWORK OPERATOR
Maximise utilisation of allocated spectrum
Flexible allocation of spectrum according to differing user demands.
Radio resource and network management to support coexistence of access schemes within allocated bands and spectrum sharing between operators.

Maintaining QoS
Maintenance of Quality of Service is a fundamental measure of network operator performance

Longevity and flexibility of network equipment
Supporting reconfiguration in the radio access equipment and the media access fabric interfacing to the core network

Owning customers
Mechanisms to support operator control of terminals, at all levels

TERMINAL AND COMPONENT MANUFACTURER
Economies of scale
Consolidation of product variants onto reconfigurable product platforms

Bug fix and software enhancement provisioning
Ability to download and install software to overcome bugs and enhance functionality/performance reduces recall costs and increases differentiation and revenue stream

Fast product creation
Reconfigurable IP authoring fostering maximised reuse, hardware/software codesign and platform-based IP integration methodology

Table 1: Reconfigurable Mobile Communications Requirements Study

1.1.2 Regulatory and Standardisation Considerations
Reconfigurable radio technology is acknowledged by FCC, the US regulatory authority, as an important mechanism to allow the modernisation of spectrum engineering practices to improve spectrum efficiency [5]. For example, software radio terminals and basestations may enable “spectrum on demand” thus improving spectrum utilisation. This will require a novel regulatory framework that should be jointly defined by industry and regulation.

Evident in the recent International Workshop on Reconfigurable Mobile Communications Systems organised by the TRUST project [6], European regulators field views and proposals. For the longer-term vision, deregulation of spectrum and minimal standardisation are key to breaking down the ‘Generation’ cycle in mobile communications, where years of standardisation work, resulting in the release of the ‘next’ generation system, may well be already outdated by technology advances. The vision of a flexible, scalable system into which new technology developments may be easily integrated, could potentially minimise standardisation requirements, specifying only fundamental communication mechanisms such as:

- An interface for connecting new radio access points to the converged backbone network;
- A communications channel to the terminal over which spectrum access is negotiated, or evolution of an adequate spectrum-access etiquette;
- A mechanism by which spectrum access may be policed (for example, consideration of the case where a software radio pollutes local spectrum, having downloaded and installed software which causes rogue emission).
Furthermore, the role of regulators will require evolution, possibly focussing on:

- Policing user privacy and security in a scheme where user profiles, configuration information, and a significant quantity of personal data is stored and widely distributed. Security and privacy measures must be jointly agreed by industry and regulations to resolve potential conflicts, for example between users and the advertisement industry.
- Ensuring fair access to spectrum through real-time policing.
- Prevention of service monopolies.

However, the huge investment in legacy systems will ensure that 2G and 3G cellular schemes will be retained for many years. The key is to establish the business model and technologies through which an evolved uniform, scalable system may be adopted as and when segments of spectrum are released, such as RegTP in Germany are also considering evolution of the regulatory position in the light of these technologies, and are encouraging the industry to:

### 1.2 Mapping requirements to system concepts and technologies

Analysis of the needs and regulatory issues reveals a mapping of requirements and constraints onto a set of key system support functions and enabling technologies. These fall into three distinct groups:

- Creation and provisioning of services over converging networks and different radio access modes
- User environment management and distributed processing framework supported by appropriate middleware(s)
- Radio reconfiguration control

#### 1.2.1 Creation and provisioning of services over converging networks and different radio access modes

A key enabler for reconfigurable radio systems is the fast creation and provision of scalable services, developed independently of, and adaptive to the underlying network technologies, environment and traffic conditions, allowing the convergence of fixed, wireless and broadcast networks. In conjunction with an appropriate management framework (distributed processing environment for reconfigurable terminals facilitated by e.g. a set of middleware) including capability negotiation and secure software download for reconfigurable terminals, services may be provisioned by adaptation to available resources, considering the terminal itself as a processing resource. This scheme requires the existence of a service description framework, possibly describing services in terms of reusable components and open interfaces, and a distributed processing control framework. The ‘Open Interface Service Provision (OISP) is an example of such a scheme being developed under the 3G Partnership Programme (3GPP). A common terminal execution environment for applications namely MExE (Mobile Execution Environment), supports WAP, personal-JAVA and CLDC/MIDP Java environments independent of the access scheme (2G, 3G, cordless, wired), and is under development in 3GPP TSG-T2. [7]. Common to each application execution environment is a terminal capability negotiation mechanism and defined security domains through which applications may be downloaded and executed with appropriate access to terminal resources. TRUST will investigate extensions to the MExE environment required to support software download of components and parameters below the application layer.

The management framework for reconfiguration must address the following functions:

- Service creation and service mobility
- Secure software download
- Scalable service provision with scalable QoS
- Common execution environment
- Consolidated billing mechanism
1.2.2 User Environment Management and Distributed Processing Framework

These software concepts essentially provide the mechanisms to support requirements for user friendliness, transparent reconfiguration and distributed application processing. In computer networks, object-oriented technologies for distributed systems provide scalability and modularity, resulting in efficient software provisioning, faster deployment and bug fixing, higher flexibility and therefore improved cost efficiency. A framework is required to coordinate and manage communications and interworking within the distributed, object oriented environment. With the continuing convergence of computer and mobile communications technologies, such a framework could supply much of the support needed to realise the user-friendly, ubiquitous environment demanded by the future user. QoS management, reliability, mobility, security, radio resource management, distributed configuration management and user-preference agents may be considered as independent distributed objects operating within the overall network concept. The distributed run time system must support following functions:

- Secure end-to-end configuration negotiation between (distributed) parties with different domains of responsibility (user, service provider, network operator, manufacturer)
- Intelligent configuration (creation/update and secure management) of applications according to terminal and service capabilities and user preferences
- Secure software download
- Distributed processing for applications and configuration management

A mobile framework for distributed processing could then be viewed as a virtual backplane supporting objects distributed between the terminal, Node B and RNC. It would be able to handle facets associated with a radio link between terminal-resident and network-resident objects, namely temporary disconnections, multiple simultaneous connections, terminal mobility, migration of objects between terminal and network via download (statically or dynamically, i.e. during run time), and should potentially support functionality such as:

- Maintenance of QoS for real-time applications
- Safe interworking between disparate software components from different sources
- Adaptation of applications to dynamic availability of resources (processing, bearer services)
- Managing handoff between air-interfaces where required services are only available to certain air interfaces
- Concurrency and persistence, allowing the implementation of distributed processing and for the support of seamless handoff between different radio access schemes

The virtual backplane concept may be schematically described as below, where objects for terminal management and user applications can migrate during run time. With migration of objects, mobile agents can be realised.

---

Figure 2: Virtual Backplane in a Distributed Processing Environment [1]
1.2.3 Radio Reconfiguration Control

The functions described in previous section represent the key enablers for reconfigurable terminals to provide the connectivity, mobility, reliability, security and QoS required to access the potential of the evolved network scenario and thus to deliver the user requirements. Some terminal-centric solutions exist today and represent the state-of-the-art, however maximum benefit requires terminal/network cooperation and this is a central theme of TRUST system aspects research. Novel solutions will be proposed and evaluated in terms of complexity, performance, overhead, tradeoff and feasibility through development of system models and simulation. In particular, the reconfiguration control has to reflect following functions:

- Radio access mode identification, blind and assisted
- Radio access mode switching management
- Simultaneous connection to multiple services
- Secure software download including authentication, hierarchical capability exchange and integrity assurance
- Efficient algorithms to realise flexible, robust radio access schemes
- Flexible, reconfigurable terminal/basestation software and hardware architectures

1.3 Constraining Considerations

The concept of reconfigurable mobile communications by its very nature implies ever increasing flexibility, resulting in a potential system management nightmare. TRUST will constrain its research by careful consideration of realistic scenarios and constraints, with particular emphasis on:

- Regulatory and system integrity considerations
- Privacy, anti-trust issues and security functions for reliable download
- Complexity tradeoffs
- Feasibility of migration from existing legacy solutions and implementations

These may lead to considerations such as:

- Software download of radio-specific software (baseband, radio protocol stack objects, parameters) will be limited to manufacturer approved builds downloaded from a manufacturer’s secure server to protect manufacturer’s regulatory liability for system integrity;
- Terminal resource sharing between application and radio-specific software will be feasible only if terminal resources used by radio control software are adequately protected;
- Software updates not specifically requested by the user may adversely modify terminal behaviour due to incompatibility with other installed software
- Anti-trust protection: Transparent introduction of software may hide or disable functions provided by a competitors application;
- Use of intelligent user preference establishment and maintenance schemes to customise the terminal’s behaviour must be secure such that user profile data is not accessible to untrusted third parties
- Volume of downloadable software and over-air service negotiation for reconfiguration must be managed to avoid network impairment
- Reconfiguration mechanisms must be backward compatible with existing radio access standards

Regarding consideration of a distributed processing environment to support reconfiguration management aspects, the cost functions are the additional terminal resources (processing power, memory, IO) and over-the-air communication overhead (throughput, delay). Whilst not specifically developing framework/middleware solutions [8], TRUST will identify the framework functionality needed to support the key reconfigurable terminal system technologies, estimate key tradeoffs and examine feasibility.

1.4 Time Table for Future Mobile Networks

We have identified an evolution path for future developments, depicted in Figure 3. This presents the medium term requirement for reconfigurable terminals to address optimum service delivery and
synergy in a composite radio environment, and the longer term view of dynamic spectrum access in a scalable communications environment.

Figure 3: Roadmap for future reconfigurability
2. Service provider and user requirements

2.1 Background and aims

Reconfigurable radio technology can impact mobile systems in a number of different areas, and it is the complexity of interactions between technology fields that makes effective design a challenge. It is hoped that reconfigurable radio will enable a completely connected world with seamless interoperability between systems and services. A number of advantages are envisaged for the wireless end user (enhanced roaming capabilities; seamless and transparent interoperability between different communication standards; optimised radio transmission characteristics according to the environment; over-the-air download of application software as and when needed), and for the network operator (advanced applications; interoperability; standard platform improvement). The reconfiguration vision is indeed a compelling one. However, mobile reconfigurable radio systems will only be adopted if they satisfy the needs of end users, operators, service providers and regulators. To address this need, a user-centred design process is advocated whereby user requirements are considered throughout, together with the impact of the emerging system(s) upon the wider organisation, stakeholders and society.

A number of industries are embracing the social, human and cognitive sciences as an integral part of interactive system design, and communication technologies are no exception ([9][10][11][12][13]). Ostensibly, this work proposes that the physical, social and cognitive needs of users should be addressed when designing all aspects of interactive systems. A user-centred design process should be followed that analyses user needs by eliciting data directly from users, and considers the impact of emerging designs upon those needs.

The projects associated with this Reconfigurability cluster present technologies that will support interactive mobile activities. The following issues are therefore important:

**Support of the IST User Focus:** The Information Society Technology programme of the Fifth Framework (European Commission, 1999) aims to empower the European citizen. Such a laudable goal can only be achieved through incorporation of user-centred research methods with the project life cycle.

**Studies of the Usage of Reconfigurable Mobile Systems are Needed:** Although a small (but growing) amount of work has been conducted in the mobile and wireless domains, there has been little work specifically addressing what reconfigurable systems can offer to users. Given that the Enabling Technologies required for reconfiguration are rapidly developing, e.g. high speed DSPs and processor cores, adaptive antennas, multimedia codecs; it is crucial that detailed end user requirements work is done before the technology mould is set.

**Operators Need to be Addressed Specifically:** Reconfiguration creates a much more complex operating model than that currently seen in the wireless telecommunications world. However, at the heart of the future SDR world will still be the network operator: the organisations that will have to deploy SDR networks and support SDR terminals. If SDR is to emerge as an important enabling technology within the wireless market, the operators’ requirements must be understood.

User requirements are being considered in several cluster projects. The aim of clustering activities in this area is to draw together the different approaches and perspectives taken in each of the contributing projects in order that all projects can benefit from the combined research effort in this area. As the research matures, the emerging user and service provider requirements can be consolidated and provide a focus for technology developments within future IST programmes.

2.2 IST 1999-Call Projects: Approaches

As part of the clustering activity, TRUST’s user requirements experts approached a number of projects within IST to investigate their approaches to user requirements. The user spectrum is broad, covering end-users, operators and service providers. The nature of activities under discussion relate to work and leisure activities and the locations can be in-building, outside or in-vehicle. The results of this study are presented in Table 2.1.
<table>
<thead>
<tr>
<th>Project</th>
<th>Relevance to Cluster</th>
<th>User group</th>
<th>Approach</th>
<th>Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUST</td>
<td>Contributing towards system and hardware concepts for reconfigurability. This work is scoped by the user scenarios and requirements.</td>
<td>Focus on Lead User Group for end users and operators to generate scenarios and use cases and identify user requirements.</td>
<td>Questionnaires, Interviews, Focus groups. Core usage scenarios will be used to drive technology simulation and testing, and to generate detailed requirements for reconfigurable radio systems. Requirements will be refined through storyboarding, prototyping and simulation.</td>
<td>First deliverable (D2.1) June 2000. Interim evaluations. Second deliverable (D2.5) end 2001.</td>
</tr>
<tr>
<td>MOTION (KA II)</td>
<td>Group working is an important application of reconfigurable mobile systems.</td>
<td>User needs within future business models.</td>
<td>Identified mobile scenarios and fundamental requirements. Focus on two end-users (Electrolux Zanussi and Nokia). Developing prototypes of mobile services.</td>
<td>Reports.</td>
</tr>
<tr>
<td>EMERGENCE (KA II)</td>
<td>Will provide information on the scale of mobile work, occupations and activities involved.</td>
<td>Mobile/nomadic work Home-based teleworking Relocated back offices Telecentres and call centres.</td>
<td>Survey of 22 countries and in-depth case-studies.</td>
<td>Results at WEDGE conference, October 2000.</td>
</tr>
<tr>
<td>MOBIVAS</td>
<td>Ensuring availability and integrity of service provision, and considering suitable business models and pricing strategies are all important issues for reconfigurability.</td>
<td>Users, service providers, terminal and network technologies.</td>
<td>Development of new business models.</td>
<td></td>
</tr>
<tr>
<td>DRiVE</td>
<td>In vehicle broadband systems will need reconfigurable components and supporting user interfaces.</td>
<td>User requirements work focuses upon the needs of network operators/service providers.</td>
<td>Requirements studied mainly through market research.</td>
<td>Deliverable (D02) October 2000. Interim user surveys and trials. Analysis – April 2000.</td>
</tr>
</tbody>
</table>

Table 2: User Requirements research in Reconfigurability Cluster
2.3 Results so far

2.3.1 TRUST

The initial phase of gathering user requirements has identified 31 business and personal use case scenarios based around three exemplar users of software reconfigurable radio. These exemplar users represent a travelling salesman, a digital film producer, and a musician. Questionnaires, interviews and focus groups were used to gather initial end user requirements. From this study, two clear user types emerged: “Gadget Crazy” and “Don’t Touch it”. The terminal design may need to differ functionally for these groups, but there is an increasing trend for the distinction between work and personal requirements to be blurred. A telecommunications market study, questionnaires and an interview with a UK operator provided data on operator requirements, and possible business models were identified. Results indicate that operators need reassuring about software reconfigurable radio: they already have much to consider with the evolution of 2G to 3G. The results from the initial study of user requirements are reported in [14]. Four key use case scenarios have been selected as the basis for further user requirements gathering (from both end users and operators), and for simulation activities within the System Concepts research within TRUST.

2.3.2 MOTION

Three mobile scenarios have been identified. One, a software engineer working in isolation while away on a trip dials in to a virtual community server to access an expert to help solve a problem. Two, an engineer visits a plant in an area where no WAN connectivity is available, but is able to use a single wireless access point to connect between his/her laptop and a knowledge base in the plant’s information service. Three, a team of plant experts from three different countries meet in a site where a new plant must be established, and are able to cooperate by sharing their knowledge base locally and thus access a unified knowledge base seamlessly. From these scenarios, fundamental requirements have been elicited relating to: personal and terminal mobility; seamless access to information; notification and reconciliation; asynchronous and synchronous communication; business process support; archival; authentication and access control.

2.3.3 EMERGENCE

A major survey in 22 countries has been carried out. This will produce reliable quantitative data about the relocation of knowledge work both within countries and across national boundaries, and whether this work is outsourced or carried out in-house. Detailed case studies conducted in developed and developing countries will provide qualitative information on the impact of new information and communication technologies upon the location of employment. First 'headline' results due to be launched at the 'Where in the World? E-work Location in a Global Digital Economy' conference, October 2000.

2.3.4 MOBIVAS

The opportunities for a wide range of innovative new products and services will only be exploited with the development of business models and pricing strategies that prove attractive to the market. A new business model is proposed, whereby the Value Added Service Providers offer their services to users after coming to a contractual agreement with the Network Operator. The users themselves have no service agreement with the Value Added Service Providers. In order to introduce content-rich value-added services in such a network environment, the MOBIVAS project introduces a number of additional entities in the network infrastructure. These entities comprise an integrated platform that will be deployed on the whole system, including the core and access networks as well as the mobile terminal and the value-added service provider domain.

2.3.5 DRiVE

A market research study has been conducted to investigate the requirements for multimedia services provided in vehicular environments. The objective was to identify user needs, to investigate the market potential for DRiVE services, and to use the results of the analysis for designing the services of DRiVE. A Web-based questionnaire enabled an end-user survey of business and private users, and interviews were carried out with business customers. The main outcomes of the analysis are provided in [15]. Firstly, there is a big interest in systems like DRiVE. Secondly, the most interesting applications for mobile services are messaging, email, chat, traffic information and emergency services. Thirdly, the users would pay for services like messaging, email, chat, Web browsing and file downloading. The end-user survey provides data on acceptable cost levels to the user for installation of such a system and for monthly charges of services. Interviews with business customers indicate a strong interest in mobile
online services, with a need for strategic partnerships to be formed. A range of business models are possible, but there are many different risk factors that may impact on the success of the business model.

2.4 Issues to be addressed in the future

Cluster projects are considering end user and service provider requirements from a number of different perspectives. TRUST and DRiVE are identifying user requirements in order that the system concepts and services can be designed to meet those requirements. MOTION also considers a number of scenarios in order that fundamental user requirements can be identified. The goals of EMERGENCE are wider than reconfigurable systems, but an understanding of future ways of working will add to our understanding of how these systems will be used. MOBIVAS is concerned with future business models, and recognises that new business models are likely to be very different from the approaches used for voice telephony.

Cluster activities in this area should focus on consolidating results from each perspective in two areas: end-user requirements (including new ways of working); and new business models (including network operator requirements). By consolidating emerging results from different perspectives, all projects can benefit from combined efforts to address user requirements as they evolve in this rapidly changing and complex domain.
3. Enabling technologies technology requirements on reconfigurable networks and terminals

3.1 TRUST Enabling Technologies

3.1.1 RF Design in TRUST

The objective of this work package is to investigate, develop and evaluate a suitable technology for these three functions appropriate for use in a flexible mobile terminal. The front-end bandwidth should be in excess of 0.8 - 2.4GHz in order to cover present and planned mobile air interface standards, with consideration an extension to 5.2GHz in order to cover the Hyperlan standard. The pre-selection of a particular standard within that front-end bandwidth should be online, flexible, and controllable via digital base-band operations.

Assuring a proper transmission over such a wide bandwidth is a state-of-the art challenge. A direct up-conversion would be the most flexible solution, but currently it is not really feasible. As it was mentioned in the receiver part, it is extremely difficult to design a quadrature local oscillator over such a frequency interval. Other problems, mainly related to the direct generation of such a high output power, support the previous conclusion.

An architecture with two or three frequency conversions seems to be the most adequate solution. In Figure 4 the general diagram of our preferred transmitter solution is proposed:

![Proposed TRUST transmitter diagram](image)

**Figure 4: Proposed TRUST transmitter**

With reference to this diagram it is easy to detect those components that are critical from a design point of view. The second local oscillator, for example, has to provide wideband operation with good phase noise characteristics. In order to employ a single local oscillator, a low Q resonator and a wideband varactor would have to be used, reducing the possibility of getting the desired phase noise figures. This problem could be solved by combining a wideband negative resistance device and a number of high Q resonators. Extreme care must be exercised when designing a switching arrangement for these resonators.

The operation of the last mixer is also quite critical. In order to assure good conversion over the whole frequency band, the local oscillator and the IF signal would have to be conveniently conditioned through the use of particular networks. “Good conversion” implies reduced mixer spurious and intermodulation products.

The power amplifier finally demands a lot of attention. Most commercially available products deal with narrower bandwidth. For the transmitter part of an SDR we are forced to design utilising a wideband power transistor. An amplifier with good efficiency, high output power and high linearity implies an
optimum device biasing and loading, the use of multiband matching networks, and a combination of predistortion and feedforward techniques. Getting all these techniques to function together is not simple, especially when each air interface standard imposes its particular conditions, and when some of these standards occupy spectral positions coinciding with harmonics of others.

The subjects already commented in the receiver about the suppression of unwanted output products through a proper IF filtering are also present in the transmitter conception. It is probable that those common problems may finally lead to some common solutions.

Considering the impact the evolution of technology may have over the RF optimum architecture, is one of the most important conclusions that this work package may provide. In this sense, the current research on direct conversions (up and down), on highly linear non-standard topologies, etc, may establish the future evolution of these “generic radios”.

### 3.1.2 Design methodology for base band

It is the understanding that the TRUST baseband is *adaptive* because of its ability to *re-configure* itself. The software architecture of this R-BB sub-system is based on Object-Oriented methodology. Each module of the baseband transceiver chain is re-configurable by instantiation of an appropriate class and/or re-initialisation of module(s) with new parameters. It is assumed that the software (i.e., class) of each module (e.g., modulator, FEC decoder, etc.) is available (downloaded), error free, and compatible. In order to re-configure the baseband transparently, i.e., without switching the terminal off, the baseband architecture must support dynamic creation and binding of new/modified modules in the baseband. As a result, instantiation of downloaded classes must be administered through dynamic binding, whereby the behaviour of the class is only required at run-time, however, the structure of the downloaded class is known *a priori*.

From the baseband perspective, there are two levels of re-configuration,

1. **Total Re-Configuration** – this refers to when the baseband is re-configured from one standard to another, i.e., inter-standard re-configuration. For example, from GSM to UMTS. Such re-configuration corresponds to exhaustive changes in the functionality, behaviour, and interfaces of the constituent modules.

2. **Partial Re-Configuration** – this refers to when one or more baseband modules are re-configured without changing the operating standard, i.e., intra-standard re-configuration. For example, certain modules may be re-configured in order to improve QoS whilst remaining on the current operating standard.

The Re-Configurable Baseband (R-BB) consists of the following components, as shown in Figure 5.

![Figure 5: TRUST Baseband Sub-System](image)
1. Re-Configurable Baseband Management Module (RMM) – this is the overall authority of the baseband sub-system. It is responsible for negotiating re-configuration, creating active and shadow transceiver chains, and controlling the run-time behaviour of each module in the baseband.

2. Active Baseband Transceiver Chain – this is the currently operating baseband chain. Each baseband chain is made up of Baseband Processing Cells (BPC). The configuration map defines the interconnections between constituent BPCs, which is stored in the local baseband library.

3. Shadow Baseband Transceiver Chain – this is the target baseband transceiver chain. It contains references of BPCs, which are kept unchanged from the active chain, and one/more new BPCs. The shadow transceiver chain complies with the configuration map as part of the agreed re-configuration strategy.

4. Baseband Software Library – it contains the configuration map of the baseband. The configuration map is a list of BPCs (with appropriate process identity for each) and their interface definitions and their respective inter-connections. The configuration map is the overall definition of the baseband, and is a piece of software in itself, which is downloaded when a new standard is implemented. In addition, the library will also store all the baseband module classes that are currently in use and also were in use for the previously operating standard. The extent of class storage in terms of current standard (and variants), previous standard (and variants) and any other newly downloaded classes is under investigation. However, it is clear that the library must store either one or both of the following,

   a. A ‘read-only’ default configuration map together with all the associated module classes and associated parameter lists. This would allow the baseband to confidently re-configure to a known, working standard, which is compliant with the user profile, and is a valid operating standard. The availability of such a default copy could be used for a full baseband installation, with the confidence of a fully working, standard compliant configuration.

   b. A complete copy of the previously working baseband configuration. This should include the configuration map, associated baseband class definition, parameter lists, operating standard/network identity, host network registrations etc. Such a store would allow the terminal to return back to a fully working baseband configuration, without the possibility of returning the terminal off the network for a full installation.

5. Re-Configuration Switch – is a typical ON/OFF switch. It implements the ON/OFF signal from the RMM, by switching the shadow chain ON and the active chain OFF.

3.2 Drive Enabling Technologies

In a multi radio scenario the efficient and dynamic use of fragmented spectrum is seen as one of the major pre-conditions to allow a cost efficient provision of multi media services. Currently the use of spectrum is fixed allocated to radio access technologies in the out-door mobile communication area.

Therefore spectrum should be allocated and released dynamically depending on local and temporal needs.

3.3 MOBIVAS Enabling Technologies

3.3.1 Embedded Beans

Similarly as JavaBeans, Embedded Beans can be characterised as components with a set of properties, methods and events. The difference between the JavaBeans and Embedded Beans is that Embedded Beans are aimed mainly to the embedded systems area. The embedded systems are known to have limitations regarding the application code size a execution speed, which had to be taken into account when the basic concepts of the Embedded Beans were created. Therefore, in comparison with Java Beans, the implementation of Embedded Beans is not limited to the Java language only – in fact, any programming language can be used. This allows the developer to create a low-level, highly optimised components in ASM language as well as a C++ or Java components with much higher level of abstraction. As the programming language is not restricted anyhow, also HW-descriptive languages like VHDL can be used in Embedded Beans. This way, a HW component can be encapsulated by a bean, which is a great feature when designed mixed HW-SW systems (systems, containing a programmable CPU and FPGA or other programmable logic device).
Within the Embedded Beans (in the following text, the term “Bean” will be used sometimes instead of “Embedded Bean”), the definition of methods is the same as in the object-oriented languages – a method is a procedure whose calling is the only way of interfacing the bean. An event is a special empty method that encapsulates the interrupt mechanism. This method is defined after the bean is instantiated in the design. A property of a bean can be used to define its internal or initialisation state and any other attributes that are needed for the description of bean functionality (e.g., for a pure HW component, the signals of the component interface are captured as its properties). The bean properties can be accessed only by bean methods and events (but their initial value can be also set in the environment before the application is compiled). The list of accessible methods then forms the interface of the bean.

A bean can also contain an unlimited number of internal language constructs that are part of its implementation (these corresponds to private methods of OOP languages). These constructs can not be accessed outside of the bean. The beans can be created as platform (hardware) dependent or independent. For the reusability of the beans, the hardware independence is the primary target when creating a bean. However, there are beans that must be architecture dependent – these beans encapsulate for example the used CPU (MCU) and its peripherals or FPGA architecture. In order to create hardware independent beans, the bean inheritance must be used.

3.3.2 Processor Expert™

Processor Expert™ is component based embedded application CASE development tool for 8/16/32bit microcontrollers and DSP processors with high level of generated application code portability, component reusability, inheritance and short learning curve using expert knowledge system assistance. Object oriented, component-based rapid application development brought revolution to nowadays programming technology. MS Visual Basic and Delphi started the new era of MS Windows applications development. In contrary of this approach the programming tools for embedded systems with 8,16,32bit microprocessors are still based on “classic” compilers, which does not reflect the modern silicon aspects. Processor Expert™ represents first product that was designed to provide unique features of rapid component oriented application development with virtual prototyping for the embedded systems designers and sharing IP by embedded components exchange. Processor Expert™ framework uses components (Embedded Beans) as building blocks for user application. Beans are “black boxes”, which encapsulate the functionality of basic elements of embedded systems like CPU core, CPU on-chip peripherals, standalone peripherals, virtual devices, programmable arrays, pure software algorithms and express these facilities as using properties, methods, and events (like objects in OOP or UML).

During the design time user defines both the low level and high level application behaviour using beans and adds his own high level code to serve application events. This process is similar to creation of UML deployment/component diagrams. Behaviour of each bean is defined through its interface and all settings are checked in background by MCU expert knowledge systems, which assists user the same way as and MCU vendor application engineer would. MCU expert knowledge system calculates overall system timing propagation in dependency of current MCU clocks, MCU peripherals usage, suggests automatic connection of peripherals to suitable MCU pins, precalculates sets of safe values for runtime usage, verifies the application timing, etc. Through this design time checkout, all beans are verified to be used properly.

In the next phase, Processor Expert™ generates resulting source code (in selected language – typically C, ASM, C++ or VHDL). The source code comes from the beans and from the user side. This source code is then translated by conventional tools (compilers, technology mappers, …), linked, downloaded and ran to the destination system with much more safety and less requirements for debugging.

Because the Embedded Beans are not restricted to any programming language, they can be implemented in Java as well. Moreover, Embedded Beans as a superset to the Java Beans allow the Java Beans to be generated from them. The usage of Java in a mobile terminal brings an evident advantage of a platform independent SW and also the possibility of adding a new functionality to the terminal by the usage of Java RMI.

3.3.3 Reconfigurable terminal design example

One of the basic requirements when designing a mobile terminal is its power consumption. Although it may be not so evident, the power consumption is a limiting factor in sense of the HW reconfigurability. For example, it is impossible to use a ten-million-gate FPGA, integrating all the DSP and MCU functions in one package with absolute reconfigurability, since its power consumption would make such a terminal unusable. A good compromise would be to use a low-power DSP and MCU processor in combination
with a small, low power programmable logic device (PLD). For example, the DSP16000 and ARM7 MCU, that can be found in Lucent’s Sceptre®3 GSM chipset are powerful enough to handle the functions needed for GSM Phase 2+ or UMTS terminal implementations. The Xilinx’s CoolRunner PLD with extreme low power consumption (<0.5mA) could then serve as HW co-processor for MCU, DSP or both. This reconfigurable coprocessor would be used in tasks, that the SW could not handle or compute efficiently (e.g., bit manipulations or data encryption algorithms needed by some part of a future mobile protocol). This HW architecture could be seen on Figure 6.

![Figure 6: Reconfigurable terminal HW architecture](image)

The SW part of such a terminal could be easily designed in Processor Expert™ with the usage of Embedded Beans library. The whole design would be done as 2 independent projects – one for the DSP part and the other for the MCU side. The PLD part (VHDL files for PLD configuration and C files for accessing the PLD HW from the processor) would be generated from either the DSP project, the MCU project or from both projects. The structure of these two example projects is shown on the Figure 7.

![Figure 7: Beans used in the MCU-side and DSP-side Processor Expert™ example projects](image)

The “OS beans” in the MCU-side project are “optional” – they would be present (and linked to) the project only in the case when a static-task, simple operating system like OSEK is used. If a complex operating system like Linux is used, it is not included in the MCU-side project, but in an independent project. As for the Java support in the terminal: the JRE can be considered as a part of the OS, so its beans are not shown on the figure. Also, some beans are connected together - e.g. the GUI, LCD and Keyboard beans. This connection illustrates the inheritance relation – the GUI bean inherits the LCD and Keyboard beans, so it can use their functionality. The “Other beans” can be for example interface beans for communication with the DSP or asynchronous serial interface for connecting to the PC.
The DSP-side project is quite self-illustrating. Since the DSP tasks does usually not require task scheduling or switching, which would bring unnecessary overhead, all the concurrency is achieved only by the interrupts handlers of the included beans.

3.4 CAST Enabling Technologies

3.4.1 RF front-end

In software defined radio, the RF, IF and A/D/A stages are not designed for any particular air interface standards, and therefore are required to be ideally ‘transparent’ in the mobile terminal to basestation linkage. This requires highly linear modules with very wide dynamic range, to cover all possible air interface requirements.

At present, we are investigating the applications of sub-harmonic sampling down-conversion for broadband RF front-end in software radio. Even though this well known technique has been used extensively in wideband instruments, the applications in software radio has not been explored but appear to be very promising.

The subharmonic sampling technique has been illustrated in Figure 8. The RF input signal is fed to a sampling switch via a series resistor R1. A narrow width pulse generator controls the sampling RF switch. The pulse generator sampling period is in turn controlled by a LO sinusoidal generator, the frequency of which is a subharmonic of the RF carrier.

![Subharmonic Sampling Downconversion Technique](image)

Figure 8: Subharmonic Sampling Downconversion Technique

The subharmonic sampling process produces copies of the modulated RF signal at the LO frequency intervals. It is important to make sure that the single sideband modulated RF bandwidth is smaller than half of the LO frequency, to avoid overlapping of the copies. The modulated RF information can be recovered by bandpass filtering at any multiple of the LO frequency, or a lowpass filter for recovering the baseband directly. The subharmonic sampling technique can be used for both the upconversion & the downconversion of the input signal. In both cases some conversion loss will occur due to harmonic mixing operation, and because most of the converted signal is filtered out and only the required harmonic is selected. The measured conversion loss and LO to RF leakage of a typical microwave sampling diode circuit, for RF frequencies of up to 20 GHz is shown in Figure 9.
3.4.2 Reconfigurable baseband processing

In recent years the Digital Signal Processor (DSP) has been the essential tool for efficient programmable processing in communications systems. The DSP, being software programmable, can offer an almost unlimited function capability although the serial processing nature of these, and indeed any microprocessor, limits the performance. A relatively new technology is, however, set to become a large competitor to the DSP. The Field Programmable Gate Array (FPGA) combines reconfigurability with processing performance comparable to Application Specific Integrated Circuit (ASIC) designs. The FPGA is now capable of tackling intensive mathematical algorithms and can also be reconfigurable in system and, in some cases, real time.

To develop algorithms for reconfigurable processing on the FPGA we require a suitable hardware platform. Many FPGA development kits are available but none of them actually incorporate the facility required to utilise the new configuration mechanisms which make these procedures possible. We have developed a reconfigurable board based on four Xilinx Virtex FPGA, which incorporates an intelligent configuration mechanism for fast and partial reconfiguration. This board is shown in Figure 10. A dedicated port is used to transfer configuration data into each device and debugging can be achieved by reading from this port. Internal registers in each device can be set to specify which part of the device is to be configured or debugged. The development system consists of software simulation, hardware verification tools and several FPGA configuration aids.

3.4.3 Intelligent Sub-System

The approach we intend to follow in this project is a new architectural model that can combine the above technologies in a coherent architecture in order to provide the most effective information. We call this architecture CODA (Complex Organic Distributed Architecture). CODA is a distributed intelligent information system, which combines data warehousing, data mining and agent technology. The general structure of CODA is illustrated in Figure 11, which shows that CODA separates the intelligent
information system into five functionally distinct layers, each supported by a data warehouse component and an intelligent component.

Each data warehouse component is separated by filter components, which restructure data into formats suitable for the information processing functions to be performed. This approach is based on the way that organic systems manage complex and adaptive behaviour. The advantages of CODA are that it is intuitively easy to manage, and supports complex evolution by using data intelligently. At the same time CODA minimises information flow between the system components and provides sufficient component autonomy. CODA will achieve this by refining and adapting Beer's Viable Systems Model (VSM with reference to intelligent information systems. The Viable Systems Model is based on the way a biological organism, such as the human nervous system, processes data in terms of objectives.

CODA uses standard distributed object technology based on the OMG (Object Management Group) CORBA (Common Object Request Broker), and CODA components will be designed using Java (JDK 1.3 or a newer version, if available) and CORBA, using VisiBroker 4. Access to the other CAST components can be done using either the CORBA ORB (Object Request Broker), based on CORBA Internet Inter-ORB Protocol, or Java RMI (Remote Method Invocation).
4. System Aspects

4.1 TRUST: System Aspects

4.1.1 Framework for Reconfigurable Terminals

Software Defined Radio is considered as a key technology but a framework is needed how the interaction between the reconfigurable terminal and network must be defined. Such a framework for enabling a standardized negotiation procedure describes the interaction between network operator, service provider and user and is illustrated in Unified Modeling Language (UML) below:

![Diagram](image)

Figure 12: Reconfigurable Terminal Interaction

Derived from this top-level use-case diagram are defined detailed use-cases, class, collaboration, message sequence and state transition diagrams. UML is used to support the system architecture definition and the detailed design of modules in terms of functionalities and interactions. SDL is also used for specific modeling schemes.

**Identify Alternative Modes**

Identify Alternative Modes has two main subordinate use cases, which are Unassisted Scanning and Assisted Scanning, both of which can be controlled by the user. The Unassisted Scanning is only utilised when the user turns the terminal on and has no prior knowledge of available modes (as a fallback). The Assisted Scanning exploits information stored in the terminal or services in the network to provide information regarding alternative mode availability. The network operators, service providers and even other terminals can provide this additional information. The user can also have preferences for particular modes and this can also be taken into account during Assisted Scanning.

**Negotiate Most Appropriate Mode**

Negotiate Most Appropriate Mode consists of two main subordinate use-cases, which are Network Capability Negotiation and Service Negotiation, (that uses Assess Terminal Capability in order to decide whether the mode can be supported). Network Capability Negotiation utilizes network bearer service profiles and user preferences (which includes the cost and performance preferences) to decide whether a particular network is suitable. The Service Negotiation utilizes service requirements, availability and capabilities together with
terminal capabilities (including the terminal resource availability) to decide which mode of operation is most suitable to support the required services. **Assess Terminal Capabilities** takes into account user preferences in terms of resource utilization (for example power consumption) and also the availability of software.

**Download Software Modules**

The main subordinate use-case for **Download Software Modules** is the actual performing of the **Software Download**, which includes the **Download Method Identification** and **Download Planning**. The **Download Method Identification** takes into account the terminal resource availability, user preferences in terms of urgency and cost of the mode change, network capabilities in the current mode (and alternative modes) and download service availability. This enables the most appropriate method of software download to be selected. The **Download Planning** takes into account the resource availability in the terminal, the user preferences for timeliness of download and cost, and the network loading to decide when to download the software.

The **Software Download** itself can then occur at the scheduled time from the selected service provider and using the selected network. The user preferences also need to be taken into account during the download especially if the demand for resources on the terminal become higher than anticipated. The user may also want to suspend the download and resume it later. Once the download has occurred, checks must be performed to ensure the integrity and validity of the software.

**Switch To New Mode**

The **Switch To New Mode** consists of **Activation** of software and hardware components that form the configuration to support the mode. This uses **Installation** and **Testing** and also includes a **Location Update** if a new network has been selected. The **Activation** of software must be performed in a controlled manner to ensure that the configuration of the terminal required for the new mode can be supported without conflict and without causing malfunctioning or abnormal operation. It is also necessary to deactivate software in the existing configuration of the terminal that cannot be supported simultaneously with the new configuration. The software module **Activation** must be authorized by responsible entities to ensure this high degree of integrity. The **Activation** of a new mode can be triggered by the terminal itself after performing mode selection, but can also be triggered by the network operator or user directly when necessary. The **Installation** of the software into the correct location and **Testing** is performed via the platform specific resource management system.

**4.1.2 System and Network Architectures Supporting Reconfigurable Terminals**

The network architecture is based on a network-centric architecture (Figure 13) involving the association of Home Reconfiguration Manager (HRM), Serving Reconfiguration Manager (SRM) and Proxy Reconfiguration Manager (PRM). This architecture extends the classical cellular Radio Access Networks.

---

Figure 13: Reconfiguration Managers enlarging Classical Cellular Networks
Interactions between terminal and network are crucial as the available bandwidth on the wireless link is a limited resource that should be used for services rather than negotiations. Furthermore, resources on the terminal itself are usually also limited. In order to relieve the terminal from the burden of frequent interactions with network entities, information from the network could be generally obtained via the PRM, which is located in the radio access network. It serves as a proxy instance for negotiations with other network entities, in particular the SRM and the HRM.

4.1.3 Overview on the system architecture

The proposed functional and network architectures have been developed from a user-orientated perspective. This has been achieved by eliciting requirements from users and developing user scenarios. From these user scenarios, the key stages (or use-cases) for terminal reconfiguration are:

- Identify alternative modes – in the geographic locality and estimate performance
- Negotiate most appropriate mode - based on criteria for selection (user preferences)
- Download software modules - using the most appropriate mechanisms switch to new mode – reconfigure the terminal to the new mode

System design roadmap resides on the following generic stages. Starting from Use Case scenarios, therefore stressing user-centric necessities, process requirements are derived supporting reconfigurability task. The previous section outlined the mayor steps of such a process. Those requirements are the basis to identify final functionalities within the process enabling terminal reconfiguration. Afterwards, system architecture is structured supplying the terminal with all the required entities providing aforementioned functionalities in a consistent manner. Thus, from the very beginning, the SDR System Aspects working group has clearly followed a Top-Down approach at system design time, which subsequent granular architecture eases upgrades and changes in radio dependent/independent functions.

Figure 14 depicts high-level Overall Functional Architecture and it shows main modular entities, each entity grouping specific functionalities and the interaction between them. It only gives a summarized feeling of what can be the final outcome of the research group, since the group is immersed on the refinement and improvement of the first general draft at actual stage.

Two main groups form the basis for the architecture. On one hand reconfigurable hardware and terminal resource system related part, depicted with horizontal lines; and on the other hand reconfiguration process control and configuration side, shown in clear grey, which is the core of this architecture and focus of the SDR System Aspects group. The later provides the terminal with the necessary functionality required during the reconfiguration process. Additionally, network implication during the reconfiguration process has also been addressed. It falls under the SDR terminal and network interactions section and those entities belonging to the field appear shadowed. Few more components appear in the architecture referring to actors who may trigger a reconfiguration process, namely User, Network Operator and Service Provider, and the Application, which is active part during the reconfiguration process as it may impose several requirements regarding RAT to be finally chosen.

Arrows drawn between entities represent interactions between those entities. They indicate either some kind of negotiation, a store/retrieve type of interaction if one of the entities represents a collection of data (depicted with wide vertical fringe), a kind of trigger, or a query operation.

Once the process requirements are understood, the main modules and their associated functionalities are presented following the reconfiguration process in chronological order. Accompanying Figures model entities by means of Unified Modeling Language (UML) at analysis stage, showing main class structures related to principal management modules involved throughout the reconfiguration process.
4.1.3.1 Modules

**Mode Identification and Monitoring Module (MIMM)**

DisCOVERS, identifies and monitors existing alternative modes within the constraints imposed by both terminal resources and the current mode. During the detection process the terminal will have to scan different frequency bands in order to detect available modes in the surrounding geographical area of the user. Therefore, the RF front-end must provide accurate dynamic behavior; i.e. be able to synchronize and retune without causing unpleasant side effects in the RF spectrum. Detection of alternative modes is very complex as a result of the constraints imposed by the current mode, in terms of the amount of “free time” available. Furthermore, if the terminal is in-session, the service must not be disrupted during the detection process, so the amount of time available is even more constrained. After this difficult task is completed and modes have been identified, monitoring of these modes is vital to ensure that a sufficient level of service and link quality is being offered before that mode can be considered as a potential roaming destination. Here again severe time restrictions are imposed by the current mode and by any active services. If the terminal can be assisted by some external entity (i.e. proxy, other terminals, third party) then the demands placed on the terminal by detection and monitoring activities can be alleviated to a certain degree, although external support cannot always be relied upon. MIMM has to determine what external support is available (if any), and the constraints of the current mode, from information provided by the Reconfiguration Management Module (RMM). It also has to determine, from information provided by the Resource System Management Module (RSMM) whether there are sufficient resources (battery power, processing power, memory, etc.) available on the terminal to be able to achieve the required mode detection and monitoring without adversely affecting the level of service in the current mode.

**Mode Negotiation and Switching Module (MNSM)**

This module guides the mode negotiation process. It checks service availability in the attempted mode, negotiates Bearer Services with the network and makes sure the terminal can provide desired performance levels in the attempted mode, regarding load of the system and link quality. Later on, and keeping in mind outputs from other modules a decision is made to change to the attempted mode or not. For decision making it is important to identify User preferences stored in the Profile Databases, link level quality provided by the MIMM, reconfiguration complexity given by CMM, time expected for software download provided by the Software Download Module (SDM), resource availability and user interaction (agreement) in case the User
Profile is not set up in detail. Results of these negotiations are stored in Lookup tables, which should be used for potential future access to the same mode and network. They would save time and resources, as negotiation results would be probably still valid.

**Configuration Management Module (CMM)**

Controls the current and possible future configurations of the terminal, which includes the configuration of the system modules that control the mode switching, software download and mode identification and monitoring. In order to enable the configuration of different distinct parts of the terminal core software and hardware to be coordinated without restricting the flexibility of the implementation, a configuration management architecture with different domains of responsibility is proposed. Two types of entity are introduced to enable interaction between these domains of responsibility, which are domain managers and terminal agents. The domain managers are responsible for control of the interactions between terminal agents in different domains and provide generic configuration information that is required by all terminal agents. The terminal agents are responsible for performing specific configuration management operations, which include the retrieval of detailed information regarding current and possible future terminal configurations and controlling the reconfiguration of distinct parts of the terminal software and hardware.

Terminal agents interact with the RSMM in order to obtain information regarding the resources available before reconfiguration is performed. Each terminal agent is also responsible for requesting (via domain managers) other configurations to be activated or deactivated and for notifying other terminal agents (via the domain managers) of any change in the configuration status controlled by the terminal agent. In addition, the terminal agents perform the necessary reconfiguration management functions in the appropriate way. For example, falling under the Reconfiguration Management Module (RMM) section, one terminal agent could be responsible for the baseband section and could create active and shadow transceiver chains and control the switching between the two chains, while another could be responsible for the RF hardware section (both of these example terminal agents are likely to be within the manufacturer domain of responsibility). Furthermore, although not currently being addressed here, terminal agents responsible for protocol stack components may fall within the network operator domain of responsibility and control different parts of the communication protocol software, whereas terminal agents responsible for transport layer protocols and higher layer protocols may fall within the service provider domain of responsibility.

The configuration management module also provides information regarding the current and possible future modes to the mode switching module and other terminal modules. Therefore, it provides the mapping between modes (which are abstract representations of the RAT air interface and higher layer communication protocol combination) and the configurations that provide the support for those modes. This will include the resource consumption and reconfiguration time estimates for a given mode, which can be a very important criterion for making a decision in the mode switching process.

**Proxy Re-configuration Manager (PRM)**

The network provided supporting entity serves as a proxy instance for negotiations with other network entities, in particular the Serving Reconfiguration Manager and the Home Reconfiguration Manager. It provides information for Mode Identification and Monitoring, Mode Switch and Software Download processes and additionally for Configuration tasks.

It takes a crucial role in the mode negotiation and also for the software download task, since it provides the mechanism for network centric software download. PRM can function as an agent for autonomous service discovery and negotiations, reducing the (bandwidth) load and need for signalling over the wireless link, decreasing the workload (CPU and battery load) of the terminal. A direct connection of the proxy to a fixed (IP) infrastructure also saves cost and enables support for partial disconnected operation of the terminal. Thereby, PRM plays an additional function as an information broker for the terminal, which distributes data from different terminal entities to a variety of other entities and vice versa.

**Software Download Module (SDM)**

SDM contains the Bandwidth Management Module (BMM), which depending on several variable values calculates the optimum download strategy and sends it to the Software Download Module. SDM then takes charge of either downloading the software from a certain entity in the network, making the most of decentralised download schemes, or from the Lookup tables, i.e. libraries, which may contain the required module from a previous reconfiguration. It is also responsible for defining who can initiate, enable and disable software downloading.
**QoS Manager**

QoS Manager takes care of firstly mapping QoS parameters from user/application requirements to the levels of underlying technology components, which are formed by the Operating System (gathering both the software and the hardware platform) and Communication components; and secondly obtaining resources from the aforementioned components. Additionally it monitors status of service, if any, regarding user/application satisfaction level and dynamically adapts the requirements to potential changes on resource availability of each component.

Mapping of parameters, subject to mapping filters availability, is preceded by acquirement of parameters from several sources. Since each component may incorporate its own QoS control mechanism, the QoS Manager only addresses matters involved with the current and target Radio Access Technologies (RAT). Thus, the sources to be checked are restricted to the capabilities of the target RAT, compared to the current one, and the capabilities of the terminal in terms of its reconfigurable modules’ performance in the new mode.

Afterwards, mapped parameters are compared against target ones (required by the application and set by the user, either on the User Profile or on demand). The comparison could be preformed at any level of parameter definition, no matter which parameters are mapped.

The QoS Manager uses the outcome information to obtain resources from specific components. With that goal in mind, it negotiates, reserves and adapts the available resources, and thus deploys the requested level of service satisfying as much as possible the user/application. Such procedures are already incorporated within the MNSM and thus, QoS Manager relies on MNSM for those purposes during Vertical Handover processes.

**User Interface / Profile Database**

User Interface and Profile Database dispatch the same type of information, which regards to user preferences for every single parameter user can set up. Profile Database, referring to the User Profile, acts as an agent for the user and keeps him unaware from underlying technology changes by making decisions upon the preferences set by the user previously.

User Interface takes a significant role whenever settings non-specified in the Profile Database are negotiated. Those cases demand user involvement for making a decision and therefore require user involvement via the User Interface.

Even though the System Architecture only shows one connection towards the MNSM, the User Interface is a basic module regarding user interactions during other actions within reconfiguration process. That is certainly true if the User Profile does not contain the specific settings demanded by each process, and therefore it cannot act like an agent for making decisions.

Examples of demanded parameters are the following ones. MIMM requires user interaction to search for certain modes related to specific services and QoS. SDM prompts the user for authorization on downloading software modules from differing sources and charging agreements for the downloading service. CMM requires user input to change terminal system configuration parameters, installation of downloaded modules and several conflicting settings as keeping records of previous configuration in detriment of saving memory.

The logic behind omitting direct connectivity between the User Interface and the mentioned modules resides on the fact that such interactions represent rather smaller amount of information exchange compared to the ones carried out with the MNSM. Thus, System Aspects group view is that remaining modules make use of an open link towards the User Interface, via MNSM.

It is worth mentioning that User Interface refers only to proposed System. It is assumed that external entities, as the Operating System, will require User Interface as well. Anyhow, such requirements are out of the scope of the system architecture and therefore deliberately omitted.

**Access Stratum Module**

Access Stratum Module contains all access specific functionality in each Access Network, which is subject to a certain RAT. It consists of the parts in the infrastructure and in the user equipment as well as the protocols between these parts specific to the RAT.

It provides real time status of resource availability in the access nodes, subject to cell load. Therefore, it trades resource allocation for potential connections via specific Radio Resource Management functionalities.
Furthermore, it integrates signal strength measurement support for link quality assessment purposes, which are essential during Mode Switching decisions.

Additionally incorporates efficient mapping from the traffic attributes used by non-Access Stratum applications, given by dominating external network technologies, to the attributes of the radio access bearer layer of the access stratum.

**Network Bearer Service Profile (NBSP)**

The Access Stratum provides flexible radio access bearers characterised by parameters describing the type of information and QoS transported over the radio interface.

NBSP is a database compiling offered Bearer Services in each network. The catalogue is an essential source for Network Capabilities negotiation and Bearer Services QoS assessment. The list presents static information regardless of the cell status and therefore, although being a key parameter during Mode Switch negotiations, it is insufficient for Mode Switching decisions.

The database is controlled and subsequently updated by the Access Stratum. Appropriateness of distinguishing it from dynamic information, available from Access Stratum, made it to be represented separately from the Access Stratum Module for sake of clarity.

**Authentication Manager**

The Authentication Manager is in charge of the wide plethora of actions related to security during a reconfiguration process, such as authentication, authorization and accounting (AAA). It provides functionalities to handle security procedures prior to proceeding vertical handovers. Thus, Corporate to Public Networks handovers are possible by means of Virtual Private Network (VPN) negotiation methods.

**Location Update Module (LUM)**

Location Update Module supports mobility for reconfigurable terminals changing between different modes; there will be interactions between the mobility management functions of different networks. Thus, it incorporates procedures for inter-working of existing protocols and numbering methods. Procedures regard to managing subscriber location data whilst maintaining confidentiality aspects. Since each network incorporates different means to perform with Location Update process, LUM relies in network side additional entities supporting seamless roaming.

### 4.1.3.2 Interfaces

The architecture definition within UML has identified a number of interfaces between the main system components. These interfaces are described further in this section.

**Configuration Management Module Interfaces**

These are interface classes for the configuration management module. They consist of interfaces from the other components in the system and from user, network operator, service provider and manufacturer applications. The different types of interface (sub classes) determine which domain managers ultimately receive the requests and notifications from the various system modules.

Users of the interfaces are:

- **Software download module** – for requesting information regarding different modes.
- **Mode identification and monitoring module** – for requesting measurement reporting.
- **Mode negotiation and switching module** – for requesting the capability, availability and resource implications of different modes and for actually requesting mode switching.
- **Resource system management module** – for notification of low power conditions.
- **Proxy reconfiguration manager** – for requesting the capability, availability and resource implications of different modes and for actually requesting mode switching.
- **User** – for requesting information about different modes and configurations (via the mode negotiation and switching module).
- **Network operator, service provider and manufacturer applications** – for requesting information about different configurations and performing configuration changes.
**Proxy Reconfiguration Manager Interfaces**

These are interface classes for the proxy reconfiguration manager, which is a network resident entity. These interfaces allow negotiations to take place in the network on behalf of the terminal in order to decide whether terminal mode switching should occur. The users of the interfaces are:

- **Service provider** – to enable the setting of policies and request mode switching within the service provider domain.
- **Network operator** – to enable the setting of policies and request mode switching within the network operator domain. This can include the setting up of network bearer service profiles.
- **MNSM** – to enable the retrieval of bearer service profiles and other cost information.

**Mode Negotiation and Switching Module Interfaces**

The mode negotiation and switching module interfaces are to allow other system modules or user interaction applications to request a change of mode. The users of the interfaces are:

- **Proxy reconfiguration manager** – for instructing mode changes and update of cached data.
- **Software download module** – for mode change requests.
- **Configuration management module** – for notifications of changes in configuration.
- **User** – for all user interaction with the system for requests to change mode, suspend mode identification and monitoring, download software (including suspension and resumption) and obtain information regarding the terminal configuration.

**Mode Identification Module Interfaces**

The Mode Identification and Monitoring Module interfaces enable other system modules to obtain information about possible alternative modes of operation. The system modules that use the interfaces are:

- **Mode negotiation and switching module** – for requesting information about available modes and the likely performance of the modes.
- **Configuration management module** – for notification of changes in mode.
- **User** – to enable the suspension and resumption of mode monitoring operations (via the mode negotiation and switching module).
- **Proxy reconfiguration manager** – for requesting information about available modes and the likely performance of the modes.

**Software Download Module Interfaces**

The Software Download Module interfaces enable other system modules to request the downloading of software and to notify the SDM of changes in configuration. The system modules that use the interfaces are:

- **Configuration management module** – for requesting software download.
- **Mode negotiation and switching module** – for requesting software availability and time required (and resources consumed) for download of software.
- **Proxy reconfiguration manager** – for requesting software availability and time required (and resources consumed) for download of software.
- **User** – to request information regarding the software available and also to request the downloading of additional software and control of the download process (via the mode negotiation and switching module).

**Resource System Management Module Interfaces**

The resource system management module interfaces enable the reconfiguration management modules (within the configuration management module) to request information regarding the resources of the terminal.

**Authentication Module Interfaces**

The authentication manager interface enables the configuration management module to authenticate the sources of requests and notifications.

**Location Update Module Interfaces**
The location update module interface enables the reconfiguration management modules within the configuration management module to notify the LUM when a change in the mode has occurred that requires a location update to be performed.

**Access Stratum Module Interfaces**

The access stratum module interface enables the proxy reconfiguration manager to obtain information about the terminal that is held in the network and is specific to the access network technology.

Reconfigurable terminals must work with different radio access technologies. Therefore, there needs to be a method of comparing different radio access technology parameters. In future standards a clear split between access stratum and non-access stratum (or core network) is likely to occur. By this, all radio specific functions are kept within the radio access network, so the core network is actually independent of the radio access technology that is chosen to be used.

**QoS Manager Interfaces**

The QoS manager interfaces enable the PRM and MNSM to request information regarding the QoS requirements and availability.

QoS Manager information is used both for mode negotiation and mode switching decision making. One of the requirements in mode negotiation would be when a mode supports the required QoS (user and application). Moreover, if the actual mode does not provide acceptable quality of service over a given period, and further negotiations to decrease QoS requirements or to increase system resources (QoS renegotiation) are unsuccessful, a mode switching decision might be triggered.

QoS is a very subjective attribute that depends on the users perception of quality. Users may sense a different quality of service for the same application even if the same bearer attributes are provided by the system. The existing interface between applications and the QoS manager allows both applications and users to provide feedback on actual QoS, e.g. request higher resource reservation.

**4.1.3.3 Deployment**

**Terminal Centric Deployment**

A terminal centric deployment assumes that all of the system modules are deployed in computational units within the terminal. In this type of deployment it is assumed that there is no need for interaction with a proxy reconfiguration manager in the network in order to perform reconfiguration of the terminal. The computational units may reside on the same or different processors within the terminal and may be processes, process threads or any other form of computational unit.

A user scenario will be used to illustrate the operation of this deployment example. Taking the first user scenario as an example:

When the user pauses the download and undocks the terminal, the MIMM is requested (via the MNSM) to enhance mode monitoring, as there are no active sessions. This means that the MIMM can request an enhanced scanning mode to rapidly detect and monitor different modes of operation.

Modes are identified by assisted scanning and the available mode information (held in the CMM) is updated. This request is authorised by first utilising the AM to authenticate the source, and the permissions for the source entity held in the CMM.

The MNSM is triggered to select the most appropriate mode. Information regarding the terminal capabilities in the current mode, together with the performance estimates and resources available in alternative modes are requested from the CMM, which in turn forwards the requests to the corresponding terminal agents. This includes an estimate of the time taken and resources consumed during the reconfiguration process.

Detailed information about the terminal capabilities and the ability to support configurations corresponding to the alternative modes are obtained by forwarding requests to terminal agents responsible for different aspects of the configuration required to support the alternative modes of operation. For example, alternative modes can require reconfiguration to take place from the physical layer right up to the network layer or even higher,
which is likely to fall into different domains of responsibility under the responsibility of different terminal agents.

Service and network negotiations take place utilising the user preferences and the currently running application QoS requirements (obtained from the QM), to enable the decision on the most appropriate mode to be made.

The request to change mode is made to the CMM. The request is forwarded to the appropriate terminal agent. The reconfiguration can then take place in the correct sequence from physical layer upwards.

The mode change may require software to be downloaded with a request being made to the SDM. Successful completion of the download is notified to the CMM.

Finally, the terminal is in the new mode of operation and the relevant entities (MIMM and MNSM) are informed.

**Network Centric Deployment**

In a network centric deployment, a proxy reconfiguration manager deployed in the network performs the majority of the negotiation functions on behalf of the terminal. This enables the terminal functionality to be simplified at the expense of more complexity within the network.

Taking the second user scenario to illustrate the operation of this deployment example. This time:

The user inserts direct media containing software for many different modes of operation. The CMM is informed (via the MNSM) that the new software modules are available.

MIMM is triggered with the terminal having no active session and so the modes are identified by enhanced assisted scanning and the available mode information (held in the CMM) is updated.

The MNSM is triggered to select the most appropriate mode. As this is a network-centric deployment (and there is no current mode of operation), the user must manually select the required mode from a list of available modes.

The request to change mode is made to the CMM. The request is forwarded to the appropriate terminal agent. The reconfiguration can then take place in the correct sequence from physical layer upwards.

The terminal is in selected mode of operation and the relevant entities are informed (MIMM and MNSM). Now the information contained within the proxy reconfiguration manager must also be updated.

When an incoming call arrives it first triggers the selection of most suitable mode in the proxy request module within the PRM.

Information regarding the terminal capabilities in the current mode, together with the performance estimates and resources available in alternative modes are requested from the proxy CMM within the PRM. This includes an estimate of the time taken and resources consumed during the reconfiguration process.

Detailed information about the terminal capabilities and the ability to support configurations corresponding to the alternative modes is obtained.

Service and network negotiations take place utilising the user preferences to enable the decision on the most appropriate mode to be made.

The request to change mode is made to the proxy CMM within the PRM. The request is forwarded to the actual CMM using the current mode of the terminal, which then passes on the request to the appropriate terminal agents. The reconfiguration can then take place in the correct sequence from physical layer upwards.

The terminal is in selected mode of operation and the relevant entities are informed (MIMM and MNSM). Now the information contained within the PRM must also be updated.
4.2 Mobivas: System Aspects

4.2.1 Service provision requirements and management

The forthcoming abundance of 3G services that will be typically developed by many co-operating entities and will need to be deployed on multiple types of networks significantly complicates service deployment and provisioning and demands a higher level of intelligence and flexibility from the underlying network. Therefore, a need for intelligent and flexible service provision platforms that will mediate between service developers, network operators and end-users is emerging. Challenging issues that need to be addressed by such platforms include the following:

- Network reconfiguration for optimal service provision. The multitude of available services, with highly diverse requirements from the network, creates the need for a dynamic and intelligent way of managing network reconfigurability to enable optimal service provision.
- Rapid deployment and efficient management of disparate services. Dynamic service introduction by their developers is a key aspect for future mobile systems, thus reducing time-to-market and increasing the range of services available to end-users. Moreover, post-deployment service management functions like usage monitoring, charging and billing are particularly critical for the commercial exploitation of a service. All these operations should be automated to the highest degree possible.
- Creation of a service one-stop-shop for end-users, where the discovery and optimal provision of a huge number of unrelated services is performed from a single user interface, and is customized to terminal characteristics and user preferences. This, to a large extent, will enable seamless and user-friendly service discovery tailored to user’s needs.

More specifically, some basic issues to be covered by flexible service provision and reconfiguration management platforms for 3G and beyond are:

- **Support of advanced business models**
- Dynamic service deployment, registration and update by VASPs
- **Dynamic user registration**
- Advanced re-configurability control (network/terminal etc.)
- **Support suitable Security schemes**
- Advanced Service discovery - provision of service listings for available VASs based on various parameters, e.g.:
  - Terminal classification
  - User profile
  - Service specific data
  - Location
- **Support of advanced Charging/Accounting/Billing schemes:**
  - Support of flexible charging/accounting models
  - User friendly Billing procedures, e.g., provision of single bill to end user for all consumed VASs

4.2.2 The need for network/mobile systems reconfigurability management

In 2G networks, services provided to mobile users were either rigidly integrated in network equipment or developed with proprietary tools by mobile operators or equipment manufacturers. This situation led to the availability of a limited number of services that were tightly coupled with the type of network and the vendor of equipment they were running on. However, in 3G and beyond, an open marketplace is expected to emerge, where a huge number of diverse services will be developed by Independent Software Vendors (ISVs) and typically will not be targeted solely to mobile networks. This situation creates the need for more flexible networks that can be adapted dynamically to the requirements of the multitude of services that are provided over them. Thus, network reconfigurability becomes a critical issue to the successful development of the 3G market according to the expectations of the market players that have invested in this technology as well as the end-users. Network and mobile systems reconfigurability will provide the glue for integrating diverse mobile technologies, services offered by numerous “players” (service providers, operators etc.) and mobile users with various profiles and requirements.

Network reconfiguration to accommodate a service may occur either during service deployment to a network or during service activation and execution. Some examples of the types of reconfiguration actions that would be useful in a mobile network are the following:

- **Quality of Service (QoS) provisioning.** Network equipment (e.g., routers) re-configuration may be necessary for e.g., the identification of the transport flows corresponding to the usage of a particular
service and support of the required QoS. The desired QoS for a service access session by a particular
user may be specific to the service (e.g., certain services may require a minimum QoS level to be
accessed) and may also depend on the identity and preferences of the user.

**Charging and billing.** The system used to gather service usage data, process it and calculate the
 corresponding charges for the end-user, should be dynamically reconfigurable. This is the only way it
can take into consideration the various charging-related events occurring in the network (tariff
updates, tariffing policy changes) and subsequently produce an accurate user bill.

**Dynamic software/protocol downloading.** The optimal provision of a service may necessitate certain
software elements to be installed dynamically to the terminal or some place in the network during
service deployment or activation. For example, due to limited bandwidth available at the radio
communications link, certain service content (e.g., images, audio) should be drastically compressed,
so that it can be transmitted in real-time to the user terminal. To do that, an appropriate codec could
be downloaded to a node at the edge of the mobile operator’s network as well as the terminal.

The reconfiguration actions have impact in various levels of mobile systems architecture and introduce
high complexity that has to be handled by some reconfigurability management intelligence (distributed or
not).

### 4.2.3 Introduction of open APIs for enabling flexible service provision and systems
reconfiguration management

Currently the applications and services being delivered by network operators are primarily voice based
and deployed almost entirely within the confines of each operator’s network. The situation is changing
quickly in several dimensions:

- Applications are moving beyond voice network services, becoming part of larger, non-telephony
  applications.
- Applications and services that may provide added value to service offerings will be developed by
  independent VAS providers.
- Applications will increasingly be hosted outside the network operator’s domain, in entities such as
  corporations and application service providers.
- Such business models presume the enabling of third-party service/application providers to offer services
  through various mobile operator networks systems. The network functionality may be required to provide
  services like call control, addressing, location, billing, and notification. Third-party applications, which
  reside on application and World Wide Web servers in application provider networks, corporate networks,
  and the Internet, and on the network operator’s network, should be able to access these network services
  in order to fulfill service provision. Business arrangements between the network operator and application
  providers dictate how these network services are accessed by the applications. A large number of
  applications should be quickly introduced by a large number of developers using commercially available
tools.
- In order to enable support of advanced business models, several industry initiatives emerging from
  consortia and standards bodies are defining architectures and interfaces, in the form of either application
  programming interfaces (APIs) or protocols, for enabling introduction of services provided by
  independent Service Providers and mobile operator networks. OSA, Parlay, JAIN are some of the efforts
  that have been elaborated to this end.

**A breakthrough to flexible service provision and reconfiguration management will be the introduction
of open APIs for service requirements registration for network/system reconfiguration, service
deployment and update, QoS requirements and reconfiguration actions, charging schemes and policy
reconfigurations, protocol/software download requirements and other aspects.**

To support this more complicated value chain, some important issues to be addressed are:

- Introduction of advanced mechanisms to enable third-party applications to access and possibly
  reconfigure network services provided by the underlying network.
- Introduction of a framework enabling network operators or third trusted parties to set up and administer
  complex, flexible, service provision and charging arrangements between network providers, end
  users and the third-party application providers.
- Enforcement of advanced security mechanisms.
- Introduction of a framework enabling dynamic user registration to services, intelligent service discovery,
  seamless service provision and reconfiguration control
Framework for a Service Provision and Reconfiguration Control Manager

Generic architecture

In order to support and manage the high complexity introduced by the requirements for flexible service provision, reconfiguration control, introduction of open APIs in various layers to support service deployment, execution and user profile driven reconfiguration actions, the introduction of a generic framework for flexible service provision and reconfiguration control is necessary. Figure 1 illustrates such a framework for:

Integrating third-party services/applications with a set of network services (for example, call control, messaging, location, and billing) available in today’s wired, wireless, and data networks
Delivering services seamlessly across these networks.

As shown in Figure 1, the Reconfiguration Control/Service Provision Manager (RCSPM) offers and supports several different open APIs to service providers enabling authorised 3rd party applications to perform reconfiguration actions on the underlying network (e.g., QoS, billing policy). Each API gives applications access to network services and provide protocol translation and network security based on policies. Although Figure 1 does not include an exhaustive list of APIs, the set of APIs presented (such as Parlay, OSA, and JAIN) is expected to grow over time. As standards evolve, the RCSPM will adaptable to support additional APIs.

In more detail the RCSPM consists of the following modules:

The Reconfigurability Manager. This module will support the reconfiguration actions that are to be performed to the underlying network infrastructure. Reconfiguration actions typically are triggered in two ways: 1) By the VASPs during the service deployment (registration) procedure. The necessary reconfiguration actions will be identified from e.g., XML-encoded service definitions fed by the VASP to the Service Deployment Manager. The latter forwards the appropriate requests to the Reconfigurability Manager. 2) By the services themselves during service execution. Through the reconfigurability extensions to open APIs, as shown in Figure 1, network reconfiguration functionality is directly accessible to authorised 3rd party applications. This functionality will be implemented via the Reconfigurability Manager. The latter maps actions to interfaces of system elements (e.g., routers, billing system).

The User Access Session Manager. A user should log-in with the RCSPM before he can access the services registered with the platform. State information is maintained during the lifetime of a user RCSPM access session.
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE SERVICE_DEFINITION "service_definition.dtd">
<Service_DEFINITION>
<Service>
  <SAPPublicKey>365636</SAPPublicKey>
  <ServiceID>1</ServiceID>
  <ServiceName>Soccer results</ServiceName>
  <ServiceVersion>2.2</ServiceVersion>
  <Description>Results of national soccer leagues</Description>
  <Language>English</Language>
  <Category>Sports/Soccer</Category>
  <Keywords>Soccer</Keywords>
  <Availability>Yes</Availability>
  <ServiceVersion>
    <ServiceVersionID>2</ServiceVersionID>
    <ServiceVersionName>Light Edition</ServiceVersionName>
    <VersionDescription>Plain text version</VersionDescription>
    <MexeClassmark>2</MexeClassmark>
    <SoftwareReq>
      <JVM>
        <Edition>Personal Java</Edition>
        <Version>1.2</Version>
      </JVM>
    </SoftwareReq>
    <TransProtocol>TCP</TransProtocol>
    <QoSIndicator>23</QoSIndicator>
    <URL>http://www.soccer.gr/results</URL>
    <IPAddr>195.138.67.173</IPAddr>
    <IPPort>8080</IPPort>
    <PricingModelNum>1</PricingModelNum>
    <TariffClassNumber>1</TariffClassNumber>
    <CostDescription></CostDescription>
    <DSERTimeout>12</DSERTimeout>
  </ServiceVersion>
</Service>
</SERVICE_DEFINITION>

Figure 16: An XML-encoded service definition

The Service Discovery Manager. This module will enable mobile users to quickly and efficiently discover the services registered with the RCSPM. Searches for services can be performed according to criteria such as category and keywords. The service listings produced contain all information that can be useful for a user to select a service (service description, indicative tariffs, etc.). These listings are tailored to the capabilities of the current terminal as well as the user preferences (e.g., language, favourite services). A terminal capability announcement mechanism and user profile information is used, respectively, for these purposes.

The User Profile Manager. The RCSPM will include user profiling logic, to enable service discovery and provision according to user preferences. The user profile contains information such as user identification data (e.g., name, IMSI, security keys), generic, service independent user preferences (e.g., language, default tariff class), user interface preferences (e.g., font size, preferred media type), as well as a list of user-specific favourite services. The system gives the user the ability to view and update user profile information at any time.

The Service Deployment Manager. Through this module the VASPs will be able to register their services with the RCSPM framework and thus make them available to mobile users. Service deployment includes insertion of the service information in the service database, as well as certain reconfiguration actions in the network (e.g., configuring network equipment to produce service usage monitoring information). The actions that should be performed during service deployment are determined by a service definition that is provided by the VASP. XML is a highly appropriate way of encoding the service definition. An example service definition in XML is shown in Figure 2. This definition obeys to an appropriate XML Document Type Definition (DTD).

The Service Data Manager. This module is an interface to a database of services that is maintained by the administrative entity operating the RCSPM (typically, a 3G mobile operator). This information is used for service discovery, and also for billing purposes. The service database may be dynamically accessed and updated by the VASPs.
Protocol Adapters

The network has a set of services such as call control, location, billing, messaging, and authentication that may span multiple network domains. To support these services, a set of protocol adapter components within the RCSPM delivers a translation capability. These protocol adapters hide the details of the network service protocol and allow services to be seamlessly delivered to multiple networks and end-user device types. Application requests made through an API are mapped or translated by API gateways and service capability components into requests to one or more protocol adapters. In turn, the protocol adapters communicate with network services using industry standard protocols—such as INAP, MAP, CAP, and SMPP—appropriate for each service. For example, an API request to deliver a short message from an application would be translated to the appropriate Short Message Service Center (SMS-C) protocol adapter of the targeted end user and to the appropriate protocol adapter of the billing service that records the billing record. The protocol adapters selected would support the class of SMS-Cs currently engaged for the targeted end user and the billing service. Likewise, requests from network services like call control are translated by protocol adapters and service capability components into a request to an application through the appropriate API gateway. Additional protocol adapter components can be added as new network service protocols are identified.

The need for implementing such protocol adapters when developing the RCSPM, obviously requires such a reconfiguration mediator to be implemented.

The need for the introduction of enhanced open APIs to enable flexible service provision and reconfigurability management is based on similar aspects that lead to the development of existing open interfaces e.g. OSA. The main aspects are: to hide internal functionality and facilitate application development, introduction and reconfiguration. The enhanced open interfaces may be associated with physical/network entities involved to necessary reconfiguration actions. Figure 3 illustrates such aspects. Network entities may provide APIs for certain configuration actions on the underlying entity. By using the offered method calls authorized developers will be able to trigger respective reconfiguration actions. The required protocol adaptation for interacting with each element will take place on internal Service Capability Servers (SCS) of network entities.

Reconfiguration via open interfaces

In view of the introduction and support of open APIs for flexible service provision and reconfiguration management, the Reconfiguration Manager may act as a gateway between the services and the underlying network for managing reconfiguration actions based on service deployment and execution requirements. Figure 4 depicts such an interaction. Advanced security schemes are necessary to be adopted to sustain feasibility of such architectures.
4.2.4 Charging in an integrated service provision and reconfigurability management framework

In order to provide a re-configurable charging, billing and accounting service to independent service providers, the support of open generic APIs should be considered. Such APIs will enable service providers to make use of charging information, configuration of relevant to charging/metering network components’ functionality based on charging policy requirements. In order to enable flexible charging accounting and billing services for 3G networks, a functional entity responsible for the overall management for charging, billing and accounting issues should be introduced. The communication between this entity and underlying networks elements, as well as service providers should be supported by open generic APIs.

Integrated and discrete model for charging and billing

Charging and Billing Process can be seen as part of the service provisioning process (integrated) or as a separate service (discrete). The different views and their impact are described below.

In the first option, the charging process is seen as part of the provisioned service. This implies that the charging and billing process is tailored to the specific service and might collect charging information by directly interacting with some service specific entity (e.g. signaling information from a SIP server). The configuration is performed as part of the service equipment configuration and policies are defined as part of the service provisioning agreement.

Charging can also be seen as a separate or discrete service on its own. In this case it does not have to be coupled to a specific service. The charging, billing and accounting service can be provided by a generic system which is able to monitor, collect and charge traffic from different services. Discrete charging can be used for outsourcing the charging and billing process. This depends heavily upon the support of advanced business models and agreements.

A service provider that has outsourced the charging service has to request the charging information from the corresponding entity. The generated records are send from the charging entity to the service provider who may make modifications to the records before sending them to the customer.
The evolution of 3rd generation networks, which integrates the IP world with the telecommunication services, is leading to the creation of an open market, with a large number of independent Value Added Service Providers (VASPs), which will deploy their services through a limited number of network providers. In such an evolving competitive market, flexible charging systems and models accommodating both aforementioned options may have potential endowment.

Metering Devices (MDs)

For the support of advanced charging systems, a Metering Device (MD) should be placed in the edge of the core network in order to monitor and classify incoming or outgoing traffic. The MD could be a Layer 4 SmartSwitch-Router that processes traffic over the IP layer and could provide network flow reporting with the required details for the services usage (service type, application port, protocol etc.) An alternative solution could be the deployment of the IPmeter approach where all traffic is monitored by a device that is not involved in routing tasks.

Two types of metering devices can be distinguished, static and re-configurable. In case of static MDs all flows are measured with a fixed granularity, not distinguishing if a subsequent charging process needs the specific meter data or not. In most cases the huge amount of captured data makes a filtering stage after the metering necessary. In case of a re-configurable MD, it collects meter data only for flows specified by metering policies.

For re-configuration of the meter process the following issues must be addressed:

- **metering scope**: whether to meter all flows or only selected flows
- **metered flow attributes**: which attributes are to be collected for a specific flow
- **meter granularity**: measurement intervals

A breakthrough enabling advanced charging systems to evolve will be the definition of open APIs through which authorised entities will be able to reconfigure such devices dynamically based on service specific charging policies adopted.

Network elements for charging issues

In mobile network infrastructure elements, such as SGSN, GGSN collect charging information and through the CGF (Charging Gateway Functionality) provide this in a secure way to the operator’s billing system. In 3G networks, the architecture will be extended incorporating an IP multimedia subsystem (IMS), that will introduce new components related to charging (e.g. AAA server). The communication with the existing entities, as well as with the possibly introduced entities should be performed over open standard APIs. Moreover a need for specifying open APIs in order to configure network components’ functionality according to the policies adopted is emerging.

Charging Billing and Accounting adaptive to each VHE - VASP category

When the VASP is a Home Environment VASP (HE-VASP), the home network has the responsibility for the charging and billing process. Therefore, the home network’s components collect information not only for the usage of its resources but also for the service usage. The home network is also responsible for the accounting of the revenue that is due to the HE-VASP.

In case that the VASP is a privileged VASP (P-VASP), the home network permits the service to use particular capabilities but the full control of services remains to the P-VASPs. Thus, the service provider undertakes the charging and billing of the users for the service usage.

When the VASP is a non privileged VASP (NP-VASP), there is no privileged relationship between home network and service provider. Therefore, the NP-VASP should calculate the charge for its service usage and get directly revenue from the user.

The assumptions that have been made during the design of the proposed architecture streamline with the requirements for flexible Charging models. In the first case, the home network provides charging and billing as a service to 3rd parties (HE-VASP). The introduced entity realizing the charging, billing and accounting service collects charging records from the network’s components (i.e., CGF, MD) and calculates separately the transport and the service part. The home network bills the users and credits the HE-VASPs according to the agreements between them.

The service provision by a NP-VASP is possible over the proposed platform and the introduced entity calculates only the transport charge for the allocation and usage of its network resources. The service bill
is issued by the service provider, since the home network is not authorized to do that. In addition, the MD will not be configured to monitor the traffic to and from a NP-VASP, since there is no agreement between network operator and NP-VASP. Thus, it cannot provide charging information for such services usage.

In case of a P-VASP, the network operator has a business agreement with the P-VASP, so the MD could monitor the incoming and outgoing traffic related to these services and generate charging records. Although the charging and billing is not responsible for the service charge, these charging records might be used or alternatively discarded by the MD. The introduced entity for charging and billing calculates the transport charge, while the service provider should estimate the service charge. In order the P-VASP to be able to use flexible billing schemes for the service usage, metering devices such as L4+ systems should be used to monitor the outgoing traffic from the P-VASP network. Another possibility is the utilization of the charging information provided by the network’s components.

In such case, an open generic API between the introduced entity and P-VASPs should be defined in order to enable the P-VASPs to make use of charging information as well as to define the desired policies according to which the network components should be configured.

**Required processes of a reconfigurable manager for charging purposes**

In order to fulfill the arising requirements for efficient and flexible charging, the reconfigurable control manager should provide additional abilities to authorized entities such as:

- Accounting indication
- Pricing Policies & Tariffs Definition
- Location-based Charging
- Reconfiguration of networks elements (e.g. MDs)

![Figure 19: Reconfigurable manager for charging purposes](image)

Figure 19 presents how these services could be offered to authorized applications through open generic interfaces.

**CAB as a re-configurable manager for charging purposes**

The CAB system as has been designed and implemented in MOBIVAS project is suitable to play the role of the functional entity that will have the overall control for charging, billing and accounting issues.
Firstly, it provides a re-configurable charging, billing and accounting service. Moreover, the CAB is able to provide open generic API to independent service providers in order to make use of charging information as well as to configure network components’ functionality according to the desired policies. In addition, the communication between the CAB and network’s elements for the transport of charging records and configuration of these entities can take place over open generic APIs.

Figure 20 illustrates the CAB as a reconfiguration manager that provides additional abilities to the authorized entities.

5. References


