



D B1.4 – Platform Assessment

Ian Cooper, Les Humphrey, Dave Thorne (BT)
BT Exact Technologies, Adastral Park, Martlesham Heath, Ipswich IP5 3RE, UK
ian.r2.cooper@bt.com, les.humphrey@bt.com, david.j.thorne@bt.com

Luis Cebrián Badesa, Ruth Vázquez Cerro, Carlos Vaquero Avilés-Casco (TID)
Telefónica Investigación y Desarrollo, S.A., Parque Tecnológico Walqa, Ctra Zaragoza
N330-A, Km 566, 22197 Cuarte (Huesca) - SPAIN
luiscb@tid.es, rv@tid.es, vaquero@tid.es

Identifier:	Deliverable D B1.4
Class:	Report
Version:	0
Version Date:	09/01/2006
Distribution:	Public
Responsible Partner:	BT, TID
Filename:	WPB1_0046_V00_final.doc

DOCUMENT INFORMATION

<i>Project ref. No.</i>	IST-6thFP-507295
<i>Project acronym</i>	MUSE
<i>Project full title</i>	Multi-Service Access Everywhere
<i>Security (distribution level)</i>	Public
<i>Contractual delivery date</i>	M23
<i>Actual delivery date</i>	M24
<i>Deliverable number</i>	D B1.4
<i>Deliverable name</i>	Platform Assessment
<i>Type</i>	Report
<i>Status & version</i>	Final – v00
<i>Number of pages</i>	33
<i>WP / TF contributing</i>	WP B1
<i>WP / TF responsible</i>	WP B1
<i>Main contributors</i>	BT, TID
<i>Editor(s)</i>	Ian Cooper (BT)
<i>EU Project Officer</i>	Pertti Jauhiainen
<i>Keywords</i>	QFD, QoS, MPLS, VPN, IP, Ethernet, ATM, SPVC
<i>Abstract (for dissemination)</i>	This deliverable describes a methodology for performing platform assessment based upon QFD techniques.

DOCUMENT HISTORY

Version	Date	Comments and actions	Status
V00	09-01-2006	Public version – methodology only	Final version

TABLE OF CONTENTS

DOCUMENT INFORMATION	2
DOCUMENT HISTORY	3
TABLE OF CONTENTS	4
LIST OF FIGURES	4
1 WHAT DOES PLATFORM ASSESSMENT HOPE TO ACHIEVE?	9
1.1 Introduction.....	9
1.2 The Evolving Access Network	9
1.2.1 <i>Required Network functionalities</i>	9
1.2.2 <i>Scenarios</i>	10
2 METHODOLOGY	11
2.1 Platform Assessment Methodology	11
2.1.1 <i>Introduction</i>	11
2.1.2 <i>QFD</i>	11
2.1.3 <i>Filling in the cells</i>	14
2.1.4 <i>Formulating the Scores</i>	15
2.1.5 <i>Analysing the Results</i>	15
2.2 Applying The Methodology to MUSE	15
2.2.1 <i>Modified QFD Methodology for MUSE</i>	15
2.2.2 <i>Definitions</i>	16
2.2.3 <i>Additional MUSE Analysis Stage</i>	17
2.2.4 <i>Service Enabler Set</i>	17
2.2.5 <i>Platform Options Set</i>	17
2.2.6 <i>Creating The Capability Matrix</i>	18
2.2.7 <i>Completing the Matrixes</i>	19
2.2.8 <i>Combining the Relevance and Goodness Factor Matrixes</i>	19
2.3 Worked Example QFD Score Calculation	24
3 DISCUSSION.....	25
3.1 Introduction.....	25
4 CONCLUSIONS.....	26
APPENDIX I SERVICE ENABLERS	28
APPENDIX II NOTES FOR COMPLETING QFD CELL WEIGHTINGS	30
APPENDIX III EXAMPLE RELEVANCE FACTOR MATRIX	32
APPENDIX IV EXAMPLE GOODNESS FACTOR MATRIX	33

LIST OF FIGURES

Figure 1 Initial framework for applying QFD to Use Case driven Platform Assessment Methodology.....	13
Figure 2 Extending QFD to Prioritise Platform Options and Service Check.....	14
Figure 3 Goodness & Relevance Factor Combination - Method A	21

Figure 4 Goodness & Relevance Factor Combination - Method B22
Figure 5 Goodness & Relevance Factor Combination - Method C23
Figure 6 Method 'C' Look-Up-Table Resultant24
Figure 7 Capability Matrix for Example Platform25

Abbreviations

AAA	Authentication, Authorisation & Accounting
AEN	Access Edge Node
AN	Access Node
ARP	Address Resolution Protocol
ATM	Asynchronous Transfer Mode
BRAS	Broadband Remote Access Server
BT	British Telecom plc
CAPEX	Capital Expenditure
COPS	Common Open Policy Services
CPE	Customer Premises Equipment
CPN	Customer Premises Network
DHCP	Dynamic Host Configuration Protocol
DSLAM	Digital Subscriber Line Access Multiplexer
IP	Internet Protocol
ISDN	Integrated Service Digital Network
ISP	Internet Service Provider
LSP	Label Switching Path
LSR	Label Switching Router
L2TP	Layer 2 Tunnelling Protocol
MAC	Medium Access Control
MPLS	Multi-Protocol Layer Switching
MUSE	Multi-User Service Everywhere
NAP	Network Access Provider
NSP	Network Service Provider
OAM	Operation and Management
OPEX	Operational Expenditure
OSS	Operational Support Systems
PON	Passive Optical Network
PPP	Point-to-Point Protocol
PPPoA	PPP over ATM
PPPoE	PPP over Ethernet
PSTN	Public Switched Telephone Network
QFD	Quality Function Deployment
QoS	Quality of Service
RADIUS	Remote Authentication Dial In User Service
RTP	Real Time Protocol
SE	Service Enabler
SIP	Session Initiation Protocol
SPVC	Soft Permanent Virtual Circuits
SS7	Switching System #7
TDM	Time Division Multiplexing
TID	Telefonica Investigación y Desarrollo
ToIP	Telephony over IP
VLAN	Virtual LAN
VoIP	Voice over IP
VPN	Virtual Private Network

References

- [1] MUSE Deliverable DA2.2, "Network architecture and functional specifications for the multi-service access and edge", January 2005
- [2] MUSE Deliverable DA2.3, "Network architecture migration mechanisms – consolidation", June 2005
- [3] MUSE Deliverable DTF1.1 "Reference Models for a European Multi-Service Access Network"
- [4] MUSE Deliverable DTF1.4 "IPv4/IPv6 forwarding access and edge platform and opportunities for IPv6 in access", July 2005
- [5] Karl E. Wieggers *Software Testing & Quality Engineering*, January/February 2000
- [6] Richard Denney, "Succeeding with Use Cases – Working Smart To Deliver Quality", Addison-Wesley, May 2005 ISBN 0-321-31643-6
- [7] MUSE Milestone MA2.3, "Network Architecture: high-level description for individual architectural issues", June 2004

Executive Summary

The purpose of this deliverable was to elaborate a methodology for the assessment for multi service access platforms.

Performing an assessment of telecom access platforms is a very complex multi-dimensional task. This is due at least in part to the subjective nature of many of the metrics used to perform the assessment that makes it difficult to accurately judge the overall performance of a particular access platform. To attempt to overcome this problem, a methodology has been devised based upon Quality Function Deployment (QFD) in order to perform Platform Assessment on the various Muse network models. It is the ability of the QFD methodology to combine diverse subjective judgements into a meaningful result that makes it seem attractive in this context.

It has been found difficult to implement an assessment methodology that is network operator independent. Originally it was envisaged that a tool could be created that would enable any network operator to be able to easily judge the MUSE platforms and consider which variant would best serve their particular requirements. However, due to the large amount of operator specific input data that is required in order to perform the assessment process, it is necessary for each operator to undergo a lengthy data inputting stage in order for them to obtain the most useful output from the assessment methodology.

The methodology described, although based upon QFD, has been modified to fit into the MUSE context. A look-up-table approach has been used to create a non-linear relationship between 'relevance' and 'goodness' which are the factors used to judge the performance of service enablers within the confines of a particular platform option. The nature of the approach adopted is that a highly discriminating system has been developed that heavily penalises platforms that do not provide the required level of service enabler support whilst conversely, the methodology does not penalise poor performance if the particular aspect of a service enabler is considered to be irrelevant to the operator. Extreme results generated by the methodology have been found to be very interesting in that they often highlight misunderstandings with regards to the performance of particular objects under assessment.

1 WHAT DOES PLATFORM ASSESSMENT HOPE TO ACHIEVE?

1.1 Introduction

One aim of the Platform Assessment task is to perform a consistency check of a defined platform by answering the question, “can the platform actually deliver the services it intended to support”?

A second aim of the Platform Assessment task is to provide a methodology that will provide operators that are not involved in MUSE to be able to assess whether an Access Platform will be able to meet their service needs.

A platform can be defined from the point of view of a vendor or from the point of view of an operator. From the vendor point of view, a platform is an assembly of building blocks (e.g. access nodes, edge nodes, servers) that allow operators for flexibly build an access and edge network in line with their specific requirements. From an operator point of view, a platform is a single defined configuration of network elements that allow the operator to implement and deploy various services from the same platform.

The MUSE documents DA2.2[1], DA2.3[2], DTF1.1[3] and DTF1.4[4] describe a MUSE platform that provides a number of variants and options, in other words more from a vendor point of view. The present deliverable describes a methodology to assess a specific defined configuration, in other words from an operator point of view.

The methodology was used in MUSE to assess some specific platform configurations. The results of this assessment are confidential because the interpretation requires the necessary MUSE background. The authors however decided that the methodology itself is valuable to share in the present public deliverable. The different options of the MUSE access platforms are however included for illustration purposes to describe the methodology.

1.2 The Evolving Access Network

One of the main priorities when considering ‘next-generation’ networks is to consider how operators are going to migrate from their existing platforms. It is also likely that regulatory issues have been, and will still need to be, taken into account. Therefore it is important to remember that none of the MUSE platforms is ever likely to be a greenfield network and considerable effort may be required by the operator in order to migrate from their existing platforms, which of course includes the necessary process and systems migration.

1.2.1 Required Network functionalities

A new network has to provide a wide variety of functionality; some of this is present in existing networks, but may need enhancement, some is entirely new. A non-exhaustive list of this functionality is given below:

Auto-Configuration: Several new alternatives are being discussed, including PPP, DHCP, RADIUS, DIAMETER, ARP, COPS. Depending upon the choice made, the Access Node will be involved in implementing this technology as it involves Layer-2.

QoS: Quality of Service is of paramount importance if next generation networks are going to be implemented. Applications such as videoconferencing and Telephony over IP (ToIP) would make little commercial sense without a minimum quality being assured. Presently there is no clear indication as to what protocols will be used to implement QoS. IntServ and DiffServ are two options (both with advantages and disadvantages) and both have new requirements for network elements (Access Node & Edge Node).

Multicast: Multimedia services will undoubtedly increase e.g. IP- Tv Internet Broadcast etc. and this traffic will impact the network together with QoS issues. Optimisation for broadcast-like traffic should be performed and the use of IP multicast could be a valid solution.

VPNs: Several VPN technologies presently exist e.g. IPSec and MPLS thus the access node (and edge node or access network switches) may need to become IP or MPLS aware.

OAM: Operation and Management is a definite requirement especially in a multi-provider scenario. End users will be affected by network failures and an SP must be able to identify the point of failure in an efficient way in order to minimize the service down-time. In a multi-provider scenario, another provider may own the point of failure; therefore this provider must be informed to facilitate the repair. A common OAM infrastructure must be therefore deployed. Such an OAM infrastructure may involve all network elements, and will need to include other considerations such as provider network privacy and security.

Security: New network capabilities imply new security risks. For example, auto-configuration processes and new QoS implementations may become susceptible to attack.

IPv6: The next generation IP protocol must be considered in the evolved network.

1.2.2 Scenarios

1.2.2.1 Present Scenarios

Present day scenarios are based on ATM access using PPP sessions (from CPE to BRAS) for authenticating and connecting users (PPPoE or PPPoA). A pre-provisioned PVC is defined between the DSLAM and the CPE in order to create Layer 2 connectivity (e.g. VPI/VCI=8/32), and PPP is used to establish CPE-BRAS connectivity. DHCP is frequently used on the client side in order to acquire an IP address. This scenario is connection oriented and is substantially different from the IP/Ethernet scenario that is connectionless.

Also, multi-provider scenarios already exist (e.g. in Spain) where different service providers provide BRAS and DSLAMs. This must be also considered when migration scenarios are defined.

1.2.2.2 Future Scenarios

Ethernet may become the future access network technology, but implementing a pure connectionless access network technology as an evolution from a connection oriented network is not a straightforward task.

Connectionless evolution tends to avoid the PPP protocol (using DHCP and 802.1x as alternates for authenticating and configuring CPE). However several problems arise from the broadcast-like behaviour of Ethernet networks, for example, security risks such as ARP poisoning. Also, DSLAMs must be capable of protecting access network (and the end-users attached to it) from malicious users. Anti-spoofing filters, Proxy ARP and VLAN support are useful tools for use against this.

Given these types of problem, other scenarios are being discussed, which maintain connection-oriented behaviour for access network by means of MPLS and/or (stacked) VLANs.

2 METHODOLOGY

2.1 Platform Assessment Methodology

2.1.1 Introduction

Defining a platform assessment methodology within the context of MUSE is not a straightforward task. The methodology has to input many diverse and relatively subjective input requirements with a wide variety of access platform options together with more stringent service requirements and be able to locate the best platform option from within this wide search-space for the delivery of the service requirements.

2.1.2 QFD

One methodology, which can be adopted to perform platform assessment, is known as QFD – Quality Function Deployment. QFD can be briefly described as a family of methodologies for providing a systematic evaluation of multiple actions in terms of their detailed contribution to multiple objectives. What this basically means is that QFD can input a diverse range of subjective input data and produce a “hard” design decision. It inputs requirements and identifies how these requirements can be met from a given set of potential solutions and it is recognised as a powerful tool for deciding upon the best solution from a range of options.

Prof. Yoji Akao and Prof. Shigeno Mizuno first developed QFD in the late 1960s in Japan as a quality control system aimed at efficiently delivering products and services. However, it wasn't until 1972 when QFD was used in oil tanker design at the Kobe shipyards of Mitsubishi Heavy Industry, that the matrix notion for the methodology was developed. QFD was later extensively used in the Japanese, and later, North American automobile industries.

QFD is not concerned with quality in the traditional sense (although it was originally developed as a technique to provide shared responsibility with regards to quality assurance metrics) and it is now associated with all the qualities of a system, including performance, reliability etc. QFD has more recently become used within the software industry where it is involved in the prioritisation of Use Cases. These Use Cases may then be linked with business drivers and it becomes possible to identify the Use Cases that are best aligned to the business drivers. It is then possible to extend QFD to identify alternative design approaches from within a set of prioritised Use Cases. Used in this way, QFD serves as a tool to identify those aspects of (say) product design best aligned to the Use Cases that are in turn aligned to the business drivers.

QFD analyses a system by applying (in this order):

- a) Business drivers
- b) User requirements
- c) System requirements

These simplified requirements are basically Karl Wiegers' levels of requirements types [5].

The business drivers are the reasons for a product being developed and what value the product will ultimately provide. User requirements reflect the point of view of the user and are described in terms of tasks or goals to be accomplished. The system requirements represents the product from the point of view of the system itself and what is required of the hardware, firmware, software to support the user requirements.

Therefore QFD could provide the process for achieving an assessment methodology with regards to a telecoms access and aggregation network – providing, that is, we can obtain the necessary business drivers and requirement models. QFD has traditionally been used within a manufacturing context, but more recently it has begun to be involved in areas of software design and areas that are more Use Case driven [6]. In particular, it is in the latter Use Case driven approach where there exists the potential to be able to perform assessment of telecoms networks.

The set of business drivers considered important to telecoms operators could be described loosely as 'customer needs'. These business drivers are actually the assessment criteria used throughout in the QFD analysis and these are listed below:

- a) Scalability – allows for growth in the network as and when required whilst minimising up-front CAPEX costs, and the ability to scale to carrier volumes.
- b) CAPEX – controlling capital expenditure is essential to ensure profitability.
- c) OPEX – this is usually viewed as being even more important to control than CAPEX. Recurring costs need to be kept as low as possible.
- d) Migration cost – the cost of migrating from an existing platform needs to be given consideration.
- e) OSS cost – operational support systems are generally very complex and involve a high level of maintenance and support. Adding a single new feature or service can impinge upon several such support systems.

- f) Configuration cost – this is the one-off cost when new systems are commissioned or new services implemented
- g) Timescale (for implementation) – any new services need to roll-out when required – this is extremely important in a competitive environment
- h) Regulatory impact – it is very important to follow regulatory guidelines.
- i) Security – it is imperative that telecoms networks are highly secure and this issue becomes even more important as networks migrate from connected to connectionless architectures.

A typical QFD analysis would begin by inputting the business drivers of a project into a set of pre-defined Use Cases (for example services such as Best Effort Internet Access) and getting as an output, a list of Use Cases having undergone prioritisation (or ranking). These Use Cases would then be input into a second matrix and used to prioritise other instances such as (say) pre-defined common SEs (where SEs are not services themselves but are building blocks that allow new services to be introduced). This is shown in Fig. 1 below.

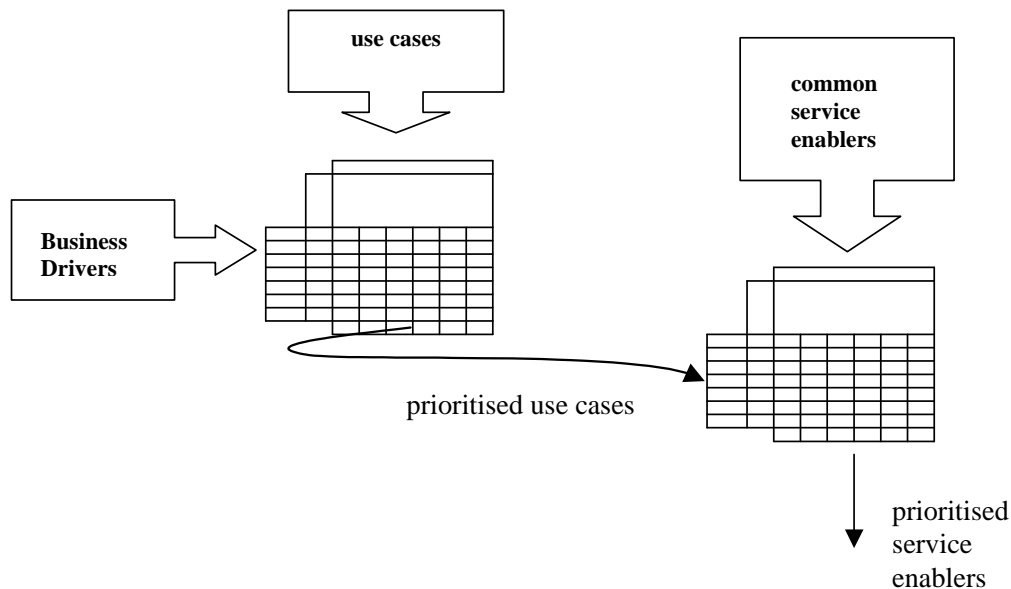


Figure 1 Initial framework for applying QFD to Use Case driven Platform Assessment Methodology

What the QFD process attempts to achieve is to convert ambiguous requirements into technical non-ambiguous requirements. This is achieved by establishing links between various instances and by prioritising these links. The need to prioritise is important since, in the case of platform evaluation, one platform design may not be the optimum for ALL the services requirements that are likely to be placed upon that platform.

Figure 1 above shows how once the prioritised Use Cases have been ordered then these can be fed into a second sequence of matrixes and used to prioritise a set of common SEs.

Figure 2 below then shows how the prioritised SEs can be used to develop a list of prioritised platform options. At this point, providing the previous stages have been properly conducted, then we should now have a list of prioritised platform options i.e. this should have chosen the access/aggregation platform option that is best suited to delivering all the required service options (as defined by the SEs) and dependent upon the specific business driver inputs as shown in Figure 1. Ideally the initial business drivers would be ‘manually’ prioritised and this feature enables the effect of different Use Case priorities to be viewed with regards to a different business driver priorities. This also allows different operators to input their own preference of business driver priorities.

Once the platform options have been prioritised, we can also provide a matrix view on the effectiveness of the platform selection technique. We simply need to check the box for each service if that platform option can provide that service. What we hope to see is a matrix with correspondingly fewer checks as we progress down the list of ranked platform options.

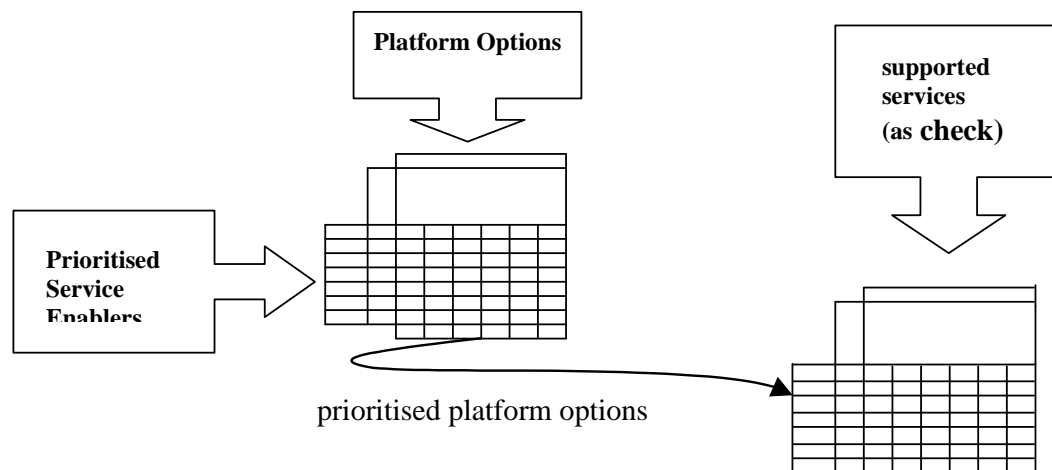


Figure 2 Extending QFD to Prioritise Platform Options and Service Check

2.1.3 Filling in the cells

Although it would be possible to fill in each cell value with almost any value – a standard guideline technique, which is known to aid the prioritisation, is to use the following common scale of values (or relative weights):

- 9 – very relevant
- 3 – relevant
- 1 – a little relevant
- 0 (blank) not relevant

Also as a guideline, depending on the total number of Use Cases being considered, on a row-wise basis it is considered best not to have more than 20% of each row given the figure '9'. The reasoning for this is based on the assumption that 20% of Use Cases delivers 80% of the greatest impact or results for your money.

However, it is always possible (especially when a general relationship between say business drivers and Use Cases is being sought – as opposed to simply finding the best choice of Use Case for a set of business drivers) that every cell could receive a 'high' figure (say 9). Basically, if one is attempting to find the 'best choice' then a restrictive use of high scores forces the issue whereas if one is simply attempting to identify non-applicable cases – then in theory all could receive a high score.

It should also be noted that when performing a QFD analysis it is more usual to have a 'team' (consensus) effort rather than have an individual filling in the relative weight scores. An important part of the QFD process is the discovery and brainstorming that occurs whilst a team is considering the correlation between various 'drivers' and Use Cases.

2.1.4 Formulating the Scores

Once the relative weights have been inputted to the table then the output is calculated by performing a sum of products calculation on a per column basis between the priority measure and each individual column. Finally these 'raw' scores are usually converted into a relative weight value and this also makes it easier to view the results and hence reorder the results since these prioritised Use Case results usually become the 'row drivers' in the proceeding QFD analysis.

2.1.5 Analysing the Results

The final step in the QFD analysis would be to input the prioritised Platform Options and correlating these with the Service Requirements. Ideally, the best Platform Option would provide a good match to every service in the table.

2.2 Applying The Methodology to MUSE

2.2.1 Modified QFD Methodology for MUSE

What we have from MUSE, is a set of defined Service Requirements and SEs together with a list of Platform Options. They were defined concurrently in order to make timely progress in the project. Therefore what we must endeavour to perform is the matching of the SEs to the Platform Options (assuming that these SEs are sufficient for the task). The aim is therefore to locate the Platform Option that best matches the SEs and hence identify the platform that will deliver the widest service bundle (from within the Service Requirements outlined within MUSE). Unfortunately, having to redefine the methodology in such a fashion is not without significant problems – the main one being the definition of exactly how this match can be performed in consideration of all the metrics that have to be considered.

Throughout this analysis we have to remember that, for the MUSE case, the QFD business drivers are effectively the assessment criteria (or Goodness Factors).

2.2.2 Definitions

SEs (Service Enablers) these are not services themselves but are building blocks that allow new services to be introduced with regards to NGN platforms (e.g. QOS support). With reference to QFD, each SE is given a priority rating of value 0,1,3 or 9 by the operator that indicates the perceived importance of this particular SE, given that operator's service aspirations.

Relevance Factors – these are operator specific values that stress the relevance of each Goodness Factor with regards to particular operators' perspective i.e. certain aspects of a particular SE could be irrelevant to an operator's network implementation objectives and therefore it would be given a correspondingly low Relevance Factor weighting.

Goodness Factors – these are analogous to Business Drivers within QFD analysis and the actual factors used to perform platform assessment are listed in section 2.1.2 above. The rationale behind the Goodness Factors is to identify salient points that are most relevant when attempting to evaluate the performance of a telecom operator network. These factors include functionality aspects, CAPEX/OPEX considerations and also timescale information. The judgement of many of these factors is essentially subjective in nature and it is the role of the QFD analysis to convert this subjective information into quantified results.

The methodology adopted is described below:

1. Prioritise the SEs as a function of importance with regards to particular operators' perspective. This is achieved by allocating a priority value of 0,1,3 or 9 to each SE. This priority value is used in the final sum-of-products calculation using the data in the 'capability matrix'.
2. Produce a 'Relevance Factor' for each Business Driver (Goodness Factor) with regards to each SE – note that this is platform independent. Details of how the relevance factor matrix is filled-in can be found in Appendix II.
3. Create a table of Business Drivers (Goodness factors) as a function of SE for each platform option. Details of how the Goodness Factor matrix is filled in can also be found in Appendix II
4. Produce the 'capability matrix' using the relevance factors and the goodness factors vs each Platform Option.
5. Perform a sum-of-products calculation column-wise for each platform option between each SE and the corresponding SE priority value that was set in point 1 above. This will prioritise the Platform Options with regards to their ability to support the SEs and hence produce a result.

2.2.3 Additional MUSE Analysis Stage

The traditional method within a QFD analysis is to perform a sum of products calculation for each of the cells in the capability matrix (in our case also including the relevance factor matrix) to form an overall weighting for each cell.

A major problem with the methodology adopted is that we have both a “relevance factor - r” matrix and a matrix of “goodness factors - g” and we need to determine what is the best method to combine these two matrixes (‘r’ and ‘g’).

If we were to perform a simple multiplication of r.g then we would not differentiate between a high goodness factor with a low relevance and a low goodness factor with a high relevance. In practice the latter should receive a much lower weighting than the former since if we have a low relevance factor then we don’t care whether the goodness factor is high or low – but if the relevance factor is high then a low goodness factor implies a deleterious effect upon that particular SE.

More detail of how the capability matrix is created is given in Section 2.2.7.

2.2.4 Service Enabler Set

The list of SEs defined within MUSE is available from document MA2.3 [7]. Appendix I lists the SEs that have been used for this analysis complete with a brief description. This list is based upon those described in MA2.3. Note that it is very important that each SE is well defined since any ambiguity at this point may have a dramatic effect upon the result of the Platform Assessment results.

2.2.5 Platform Options Set

This section attempts to define a set of Platform Options and uses DTF1.1 (Reference Models for a European Multi-service Access Network) [3] as a basis. It also considers DTF1.4[4] for the IP reference model in more detail and DA2.2[1] for the corresponding Ethernet reference model.

The multiple deployment scenarios represented mean that there are also multiple variants of the MUSE access platform in addition there are additional variations arising from multiple architectural choices.

From DTF1.1 we obtain the following list of platform options:

2.2.5.1 Ethernet Network Model

1. Ethernet Intelligent Bridging – *connectivity in the AN is based upon MAC addressing (as in a standard Ethernet switch) however the AN has added intelligence for security, traffic management and accounting.*

2. Ethernet C-VLAN Cross-Connecting (residential) – *connectivity at the AN is based upon allocating a VLAN –ID to every end user. The AN behaves as a cross-connect, switching ports via the VLAN IDs for connectivity downstream and switching upstream traffic to the uplink.*
3. Ethernet S-VLAN Tagging (business) – *Business users generate 802.1Q-tagged Ethernet frames and expect them to be transported transparently across the network to one or multiple business locations (L2 VPN). In the AN the upstream frames are transparently sent and tagged with the corresponding S-VLAN (Based on the line), whilst the downstream frames are transparently transmitted (after stripping the S-VLAN tag) to the line corresponding with the S=VID. This approach can be combined with the bridged mode and cross-connect model.*

2.2.5.2 IP Network Model

4. IPoPPPoE Transparent L2 Switching – *keeps the PPP termination at the same place as for the Ethernet model. The AN therefore switches traffic transparently at L2 towards the AEN where it either terminated (IP Wholesale) or tunnelled towards the NSP (PPP Wholesale via L2TP).*
5. IP PPP Relaying – *relaying the PPP traffic in the AN which allows terminating L2 for PPPoE traffic before forwarding the PPPoE payload to a PPP server.*
6. IP PPP Processing – *Fully process the PPP sessions at the IP forwarder either by L2TP tunnelling towards an EN or by terminating and aggregating the PPP sessions. There is full separation at L2 between users and the aggregation network for this traffic.*
7. IP Forwarding – *NAP provides the IP transport service. IP sessions are characterised by the MAC and associated IP address in the CPN and optionally by C-LAN when service multiplexing is used.*
8. IP Routing (For Application Wholesale) – *here the NAP provides routed IP service for application wholesale. The NAP and NSP provide an IP service to end customers. In an access network with routed IP there is no end-to-end L2 connectivity between the CPG and the AEN. The CPG is therefore connected via L2 to the nearest IP forwarder in the network*
9. IP Routing for IP Wholesale – *similar to (8) except that IPoE traffic for retail and wholesale users must be supported.*

2.2.6 Creating The Capability Matrix

In order to allow the impact that each SE has on each platform option we need to construct the capability matrix that correlates SE with platform option. In order to achieve this we have to also consider the 'Goodness Measures' (which in QFD terminology could be considered analogous to the Business Drivers).

2.2.7 Completing the Matrixes

The first task is to prioritise the SE from an operators' perspective. This is manually achieved (with scores of 0,1,3,or 9) and, like all QFD process discussions, as much debate as possible should be performed at arriving at these priority values.

Then it is required to complete the 'Relevance Factor' matrix as described in section 2.1.3.

Appendix III shows an example 'Relevance Factor' matrix together with the manually assigned priority values for each SE.

The third (and more onerous) task is to create the capability matrix where, for each SE and platform option, the Goodness Factors have to be created. This means that in this case we have to consider 9 goodness measures for 9 platform options and for 19 SEs making 1539 entries in total. The capability matrix is computed from the Goodness and Relevance Factor matrixes plus the prioritised SE values are used to calculate the final weighted outputs.

Appendix II lists some of the questions that an operator should ask in order to successfully complete the Goodness Factor matrix.

Appendix IV shows an example Goodness Factor matrix for a single platform option.

2.2.8 Combining the Relevance and Goodness Factor Matrixes

Having created the goodness factor (g) and relevance (r) factor matrixes the next problem to be overcome is how are these two matrixes best combined. One obvious method would be to create a 'goodness' vector by calculating the magnitude of the resultant vector when the 'g' and 'r' vectors are combined i.e.

$$v = \text{SQRT} ((g1.r1)^2 + (g2.r2)^2 + \dots\dots\dots(g9.r9)^2)) \quad \text{Method A}$$

An evident problem with such a calculation is that one would obtain an equal result for a high goodness factor with a low relevance factor as that obtained with a low goodness factor with a high relevance. This is intuitively incorrect since one would expect a high relevance and a low goodness factor to have a lower overall score than a low relevance and a high goodness factor; in the first case the particular SE has a bad score against something that is very relevant and in the second case the SE has a good score – but this doesn't matter since the result is irrelevant.

An alternative approach would be to consider using a look-up-table to create a non-linear relationship between the 'g' and 'r' factors e.g.:

g	9	3	1	0	9	3	1	0	9	3	1	0	9	3	1	0
r	9	9	9	9	3	3	3	3	1	1	1	1	0	0	0	0
o/p	9	3	1	0	4	2	1	0	2	1	0	0	0	0	0	0

The resultant vector can still be created in the same manner:

$$v = \text{SQRT} (o/p1^2 + o/p2^2 + \dots\dots o/p9^2) \quad \text{Method B}$$

Using uniform goodness factor matrixes (of values 0,1,3 and 9) plus a random goodness factor distribution and creating two capability matrixes we can see the difference between these two techniques.

Figure 3 below shows the results for Method A and Figure 4 shows Method B. It can be seen that in both cases, although the magnitude of the outputs are very different (except in the 0 case), the form of both graphs is very similar. In both cases the methodology is capable of discriminating between the uniform goodness factor inputs and also produce a similar relative magnitude output in the random distribution case.

An alternative approach would be to introduce a negative weighting system to attempt to provide strong discrimination against zero values in the Goodness Factor matrix (i.e. heavily penalise poorly supported SEs) and this can be implemented, for example, using the following look-up-table to create a non-linear relationship between the goodness and relevance factors:

g	9	3	1	0	9	3	1	0	9	3	1	0	9	3	1	0
r	9	9	9	9	3	3	3	3	1	1	1	1	0	0	0	0
o/p	9	3	-3	-9	4	2	-2	-4	2	1	-1	-2	0	0	0	0

The purpose of introducing a negative resultant is to increase the discrimination between poorly performing aspects of platform analysis. For example, using Methods A or B the lowest value that can be achieved for a single entry in the Capability Matrix is zero. However, if a platform consistently performs poorly at aspects considered to be very relevant by an operator, then logically, such aspects should be allowed to accumulate and should not be capped at zero.

The resultant can no longer be a vector so a simple summation is performed to produce an output scalar.

$$s = o/p1+ o/p2+ \dots\dots o/p9 \quad \text{Method C}$$

Figure 5 shows the result using this methodology to combine the relevance and goodness factors. It can be seen that the output is now symmetrical (which the look-up-table values would suggest) with the 0 and 1 uniform goodness factors yielding an overall negative result whilst the 3 and 9 planes yield a positive result. The random goodness factor distribution yields a result that is approximately zero.

Both method B and C are able to discriminate between low goodness factors and high relevance compared to high goodness factors and low relevance. During initial experiments with 'real' platform data it was found that Methods A & B do not discriminate sufficiently against 'zero' scores in the Goodness Factor Matrix and therefore it was decided to proceed with Method C as the chosen methodology for this platform assessment task.

2.2.8.1 Method A

service enablers	priority	0s	1s	3s	9s	random
point-to-POP QOS flow	9.00	0.0000	25.6320	76.8960	230.6881	121.7497
peer to peer	1.00	0.0000	19.0000	57.0000	171.0000	50.1199
charging	3.00	0.0000	25.6320	76.8960	230.6881	183.3439
service performance monitoring	3.00	0.0000	24.0208	72.0625	216.1874	118.7182
service testing	9.00	0.0000	19.0000	57.0000	171.0000	0.0000
security	9.00	0.0000	25.6320	76.8960	230.6881	145.9555
multicasting	1.00	0.0000	8.0623	24.1868	72.5603	48.8365
content caching/network storage	1.00	0.0000	7.4833	22.4499	67.3498	47.0638
labelling	9.00	0.0000	21.0000	63.0000	189.0000	98.2700
network resource provisioning	1.00	0.0000	23.8537	71.5612	214.6835	148.2194
media transcoding	1.00	0.0000	7.5498	22.6495	67.9485	0.0000
encrypted QOS stream support	3.00	0.0000	11.3578	34.0735	102.2204	31.6228
conferencing	3.00	0.0000	16.7631	50.2892	150.8675	0.0000
legal intercept	3.00	0.0000	20.8087	62.4260	187.2779	145.4029
signaling awareness	3.00	0.0000	11.7047	35.1141	105.3423	85.4927
stream oriented firewalling	1.00	0.0000	14.4568	43.3705	130.1115	48.5592
emergency call support	9.00	0.0000	22.4722	67.4166	202.2498	0.0000
raw score		0.0000	1434.8932	4304.6796	12914.0387	5330.3174
relative weight		0.00%	5.98%	17.95%	53.84%	22.22%

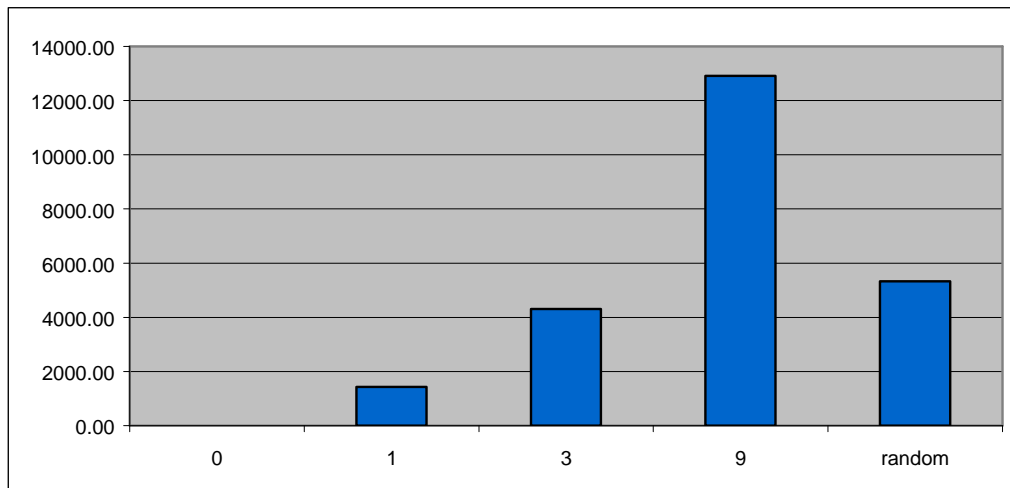


Figure 3 Goodness & Relevance Factor Combination - Method A

2.2.8.2 Method B

service enablers	priority	0s	1s	3s	9s	random
point-to-POP QOS flow	9.00	0.0000	3.0000	8.7178	25.7682	16.1555
peer to peer	1.00	0.0000	2.8284	7.2801	19.7990	9.6437
charging	3.00	0.0000	3.0000	8.7178	25.7682	15.9687
service performance monitoring	3.00	0.0000	2.8284	8.2462	24.2281	16.2481
service testing	9.00	0.0000	2.8284	7.2801	19.7990	10.7703
security	9.00	0.0000	3.0000	8.7178	25.7682	6.8557
multicasting	1.00	0.0000	2.6458	5.4772	10.9545	6.4807
content caching/network storage	1.00	0.0000	2.4495	5.0990	10.1980	3.0000
labelling	9.00	0.0000	3.0000	7.8102	21.6564	6.3246
network resource provisioning	1.00	0.0000	2.6458	8.0623	23.9792	7.2111
media transcoding	1.00	0.0000	2.4495	5.1962	10.3923	4.4721
encrypted QOS stream support	3.00	0.0000	2.4495	5.6569	13.1529	1.4142
conferencing	3.00	0.0000	2.6458	6.7082	17.7482	10.3923
legal intercept	3.00	0.0000	2.8284	7.6158	21.3776	16.2173
signaling awareness	3.00	0.0000	2.6458	5.9161	13.6015	4.8990
stream oriented firewalling	1.00	0.0000	2.6458	6.3246	15.8114	5.1962
emergency call support	9.00	0.0000	2.8284	7.9373	22.8473	13.4536
raw score		0.0000	196.7699	530.1910	1481.3158	713.4594
relative weight		0.00%	5.39%	14.53%	40.60%	19.55%

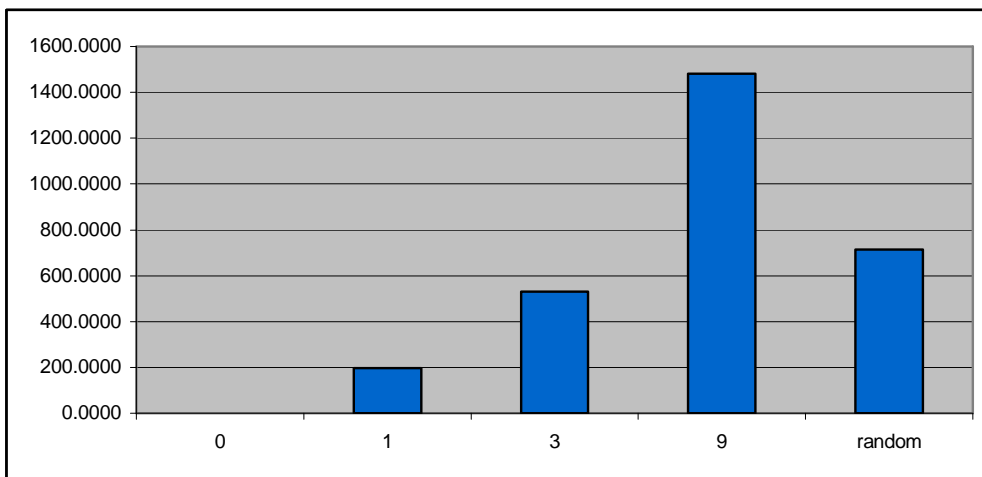


Figure 4 Goodness & Relevance Factor Combination - Method B

2.2.8.3 Method C

service enablers	priority	Platform Options				
		0s	1s	3s	9s	random
point-to-POP QOS flow	9.00	-76.0000	-26.0000	26.0000	76.0000	-16.0000
peer to peer	3.00	-54.0000	-21.0000	21.0000	54.0000	-2.0000
charging	9.00	-76.0000	-26.0000	26.0000	76.0000	4.0000
service performance monitoring	9.00	-69.0000	-24.0000	24.0000	69.0000	18.0000
service testing	9.00	-54.0000	-21.0000	21.0000	54.0000	-11.0000
security	9.00	-76.0000	-26.0000	26.0000	76.0000	32.0000
multicasting	1.00	-32.0000	-16.0000	16.0000	32.0000	11.0000
content caching/network storage	3.00	-28.0000	-14.0000	14.0000	28.0000	-14.0000
labelling	9.00	-61.0000	-23.0000	23.0000	61.0000	7.0000
network resource provisioning	1.00	-67.0000	-23.0000	23.0000	67.0000	-24.0000
media transcoding	3.00	-30.0000	-15.0000	15.0000	30.0000	6.0000
encrypted QOS stream support	3.00	-35.0000	-16.0000	16.0000	35.0000	8.0000
conferencing	9.00	-47.0000	-19.0000	19.0000	47.0000	-8.0000
legal intercept	9.00	-59.0000	-22.0000	22.0000	59.0000	-14.0000
signaling awareness	3.00	-37.0000	-17.0000	17.0000	37.0000	-11.0000
stream oriented firewalling	1.00	-42.0000	-18.0000	18.0000	42.0000	2.0000
emergency call support	9.00	-64.0000	-23.0000	23.0000	64.0000	-4.0000
raw score		-5931.0000	-2196.0000	2196.0000	5931.0000	22.0000

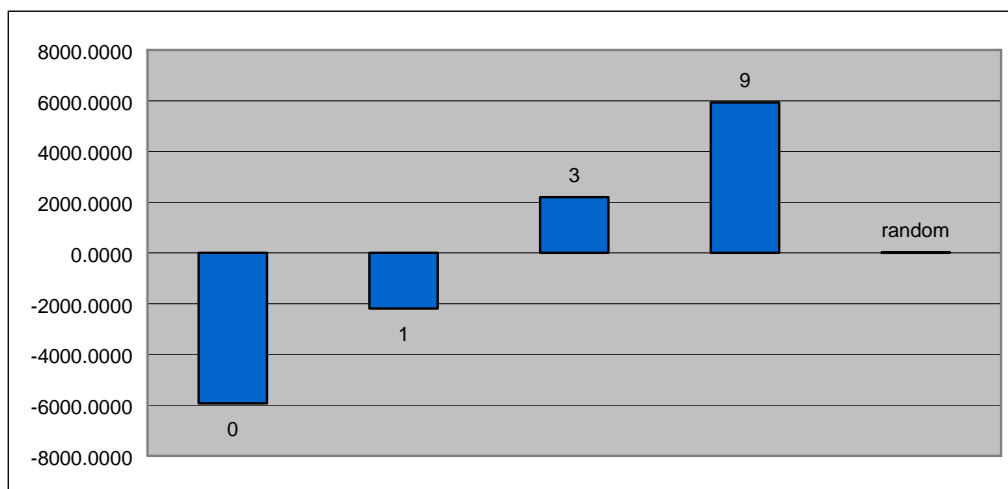


Figure 5 Goodness & Relevance Factor Combination - Method C

Figure 3, Figure 4 and Figure 5 above shows the resultant capability matrix when test data representing platform options is subjected to the goodness & relevance combination Methods A, B & C respectively. In each case the 'BT' relevance matrix has been used but in place of a platform specific goodness factors, uniform goodness factors have been used with values of 0, 1, 3 and 9 – where the 9 case would represent the 'perfect' platform and '0' would represent the worst possible case. In addition, a random goodness factor distribution has also been introduced for comparison. Note also that Methods B&C both require an intermediary look-up-table stage required in the calculation that has not been shown but this is included in the worked example shown below.

2.3 Worked Example QFD Score Calculation

Using the example relevance and goodness factor matrixes (as given in Appendixes III and IV respectively) we can calculate the QFD score for this particular example platform option using 'Method C'.

First the goodness and relevance factors need to be combined using the look-up-table described in section 2.2.8 (Method 'C') and the result of this operation can be seen in Figure 6 below:

	SCALABILITY	CAPEX	OPEX	Migration	OSS	Config.	Timescale	Regulation	Security
point-to-POP QOS flow	3.00	1.00	3.00	-3.00	-3.00	3.00	3.00	1.00	-2.00
peer to peer	2.00	9.00	4.00	1.00	2.00	0.00	4.00	2.00	-1.00
charging	3.00	3.00	3.00	-2.00	3.00	3.00	3.00	2.00	-9.00
service performance monitoring	-2.00	2.00	3.00	2.00	2.00	-3.00	3.00	2.00	-3.00
service testing	-2.00	2.00	2.00	1.00	3.00	0.00	3.00	2.00	-3.00
security	-3.00	2.00	-3.00	-3.00	3.00	-3.00	3.00	-3.00	-3.00
multicasting	-2.00	2.00	-2.00	1.00	2.00	2.00	1.00	-1.00	1.00
content caching/network storage	2.00	2.00	2.00	2.00	2.00	1.00	2.00	0.00	-2.00
labelling	-1.00	1.00	-3.00	2.00	-3.00	-2.00	2.00	1.00	2.00
network resource provisioning	-3.00	2.00	2.00	3.00	3.00	3.00	3.00	1.00	-1.00
media transcoding	1.00	1.00	-1.00	1.00	1.00	1.00	1.00	1.00	-1.00
encrypted QOS stream support	-1.00	2.00	2.00	1.00	1.00	2.00	2.00	2.00	2.00
conferencing	1.00	2.00	3.00	1.00	2.00	3.00	2.00	1.00	3.00
legal intercept	-1.00	2.00	-2.00	2.00	2.00	-3.00	-3.00	-3.00	-3.00
signaling awareness	3.00	-2.00	2.00	-1.00	2.00	0.00	-2.00	0.00	-2.00
stream oriented firewalling	2.00	2.00	-2.00	-1.00	-1.00	-3.00	2.00	1.00	3.00
emergency call support	4.00	2.00	-3.00	3.00	2.00	-1.00	-3.00	-2.00	3.00

Figure 6 Method 'C' Look-Up-Table Resultant

Using 'Method C' the resultant is a simple scalar i.e. the sum of each row provides the corresponding value for each SE. The results of these summations for each SE can be seen in the capability matrix as shown in Figure 7.

For example the value entered into the capability matrix for SE 'point-to-POP QOS flow' is:

$$3+1+3-3-3+3+3+1-2 = 6$$

Performing a sum of product calculation between the scalars described above and the corresponding priority value (for each SE) produces the overall QFD score.

In this case the calculation is:

$$9*6+9*23+3*9+1*6+1*8+3*(-10)+3*4+3*11+9*(-1)+1*13+0*5+0*13+1*18+0*(-9)+3*0+3*3+9*5=393$$

It now be seen that in order to compare different platform options then each option has to have a goodness factor matrix that is subsequently used to compute the QFD score for that particular platform. By ranking the scores one has effectively ranked each platform with regards to how it can support the SEs and hence the required service support.

service enablers	Platform Options							
	priority	example platform 1	example platform 2	example platform 3	example platform 4	etc	etc	etc
point-to-POP QOS flow	9.00	6.0000	?	?	?	?	?	?
peer to peer	9.00	23.0000	?	?	?	?	?	?
charging	3.00	9.0000	?	?	?	?	?	?
service performance monitoring	1.00	6.0000	?	?	?	?	?	?
service testing	1.00	8.0000	?	?	?	?	?	?
security	3.00	-10.0000	?	?	?	?	?	?
multicasting	3.00	4.0000	?	?	?	?	?	?
content caching/network storage	3.00	11.0000	?	?	?	?	?	?
labelling	9.00	-1.0000	?	?	?	?	?	?
network resource provisioning	1.00	13.0000	?	?	?	?	?	?
media transcoding	0.00	5.0000	?	?	?	?	?	?
encrypted QOS stream support	0.00	13.0000	?	?	?	?	?	?
conferencing	1.00	18.0000	?	?	?	?	?	?
legal intercept	0.00	-9.0000	?	?	?	?	?	?
signaling awareness	3.00	0.0000	?	?	?	?	?	?
stream oriented firewalling	3.00	3.0000	?	?	?	?	?	?
emergency call support	9.00	5.0000	?	?	?	?	?	?
raw score		393.0000	?	?	?	?	?	?

Figure 7 Capability Matrix for Example Platform

3 DISCUSSION

3.1 Introduction

The Methodology introduced in this report is capable of performing sufficient discrimination (both positive and negative) with regards to the SEs used for the assessment of telecoms platforms (using Method 'C'). However, increasing the numbers of SEs used in the assessment would probably lower the overall discrimination ability of the technique.

As the list of goodness measures becomes more diverse then results would be expected to tend towards a normal distribution. This is due in part to the fact that the more capable a platform becomes then the more expensive this platform usually becomes from a CAPEX perspective and also it tends to score badly with regards to OPEX and configuration cost.

An alternative approach to using the QFD capability matrix results would be to concentrate upon the particular SEs, which a platform cannot support. An operator would base their platform decision purely upon functional capability. Once this has been performed then CAPEX and OPEX aspects can be considered separately i.e. to select from that subset of platforms that meets their functional requirements.

Initially, when the QFD based methodology was being considered, it was expected that it would be possible to generate a Goodness Factor matrix that was operator invariant. This would mean that in order for a 'new' operator to perform the platform assessment task then it would simply be a matter of them generating their own Relevance Factor matrix and providing a list of priority values for the SEs. Unfortunately, it soon became apparent whilst populating the Goodness Factor matrix that this task was far too operator specific for the 'common' Goodness Factor approach to be viable. It would however, be interesting to obtain and compare Goodness Factor matrixes from several different operators. It would also be interesting to use the modal value between several operators' Goodness Factors since this would approximate consensus with regards to QFD methodology.

Considering that QFD is based upon a consensus process, it is feasible that as further operators add their own input matrixes, then common patterns may be identified in the input data that would allow for a reduction in the overall input requirements of later operators. This question can only be answered if a significant number of operators complete the lengthy task of completing the "goodness factor" matrixes.

It should be noted that ideally, an operator should complete all the 'goodness factor' matrixes in a single sitting. This is due to it being found difficult to regain the orientation that was prevalent after taking a long break whilst completing the matrixes. It is expected that an operator can complete the 'goodness factor' matrices in approximately half a day – although QFD being a consensus driven methodology means that if there are many wildly differing views for a given cell then a time limit may have to be enforced.

Due to the particular look-up-table approach adopted in this study it has been found that the platform assessment methodology approach adopted is highly biased positively towards very good aspects and negatively towards bad aspects and it is reasonably invariant to plain average performance. As such the methodology is highly discriminating and although the goodness factor matrixes between different platforms can give similar scores when examined from an overall row and column score. When combined with the relevance factor matrix, small differences in the goodness factors can be amplified to produce a vastly different output result.

In order to use this methodology effectively it is important to use the full range of 0,1,3,9 scores when completing the relevance and goodness factor matrixes.

4 CONCLUSIONS

A methodology has been described that is based upon QFD to perform platform assessment for access/aggregation platforms. Given the subjective nature of the scores, some margin in the results should be taken into account and no absolute ranking can be given.

The initial aim of the methodology adopted was to create a tool that could be used by different operators to evaluate access platforms by inputting only a limited amount of operator-specific information. However, it was found that the whole platform assessment process is extremely operator specific and therefore any third party operator that wishes to use this technique to evaluate access platforms will have to undergo the lengthy process of creating a Goodness Factor Matrix for every platform being considered.

Extreme results generated by the methodology have been found to be very interesting in that they often highlight misunderstandings with regards to the performance of particular objects under assessment.

Remarks:

It should be noted that ideally, an operator should complete all the 'goodness factor' matrixes in a single sitting. This is due to it being found difficult to regain the orientation that was prevalent after taking a long break whilst completing the matrixes. It is expected that an operator can complete the 'goodness factor' matrices in approximately half a day – although QFD being a consensus driven methodology means that if there are many wildly differing views for a given cell then a time limit may have to be enforced.

APPENDIX I SERVICE ENABLERS

The MUSE SE list has been modified slightly to include those that have an influence upon platform assessment.

point-to-POP QoS flow: Setting up of a bi-directional flow between an end-user and a service edge node (and beyond into the service/application domain). Each flow may have different bandwidths and QoS requirements.

peer-to-peer: Setting up of a bi-directional flow between two end-users, without going via the POP.

charging (Usage Recorder): Records the usage of all services and network resources. Accounting units may be measured by unit time, traffic volume or number of invocations of a particular network resource.

service performance monitoring: Monitors the performance of network flows. Extracts performance measures directly from various network elements and also via the element manager of CPE and Service Domain network elements.

service testing: Provides a suite of test functionality both intrusive and non-intrusive with a means of correlating and interpreting the results. Test functionality includes: loop-backs (at various transport layers), service performance alarms, extraction of physical layer performance parameters, device logs per physical node and service specific tests.

security: Provides a set of functions that secures the network elements (including the MUSE domain) plus the CPE/service/application domains from malicious attacks and fraudulent use.

multicasting: Provides the near simultaneous distribution of content to multiple users. Multicasting may be performed at L3 (IP), L2 (ATM or Ethernet) and L1 when PONs are used as the physical layer.

content caching/network storage: Some services require that different end users have access to the same content but replayed at a different time. Such a service may be provided by content caching within the MUSE domain and could be provided as a wholesale service to a content provider.

labelling: There are three main labels that are required to manage the provision of services over a MUSE network namely the network-edge label, the user label and the service operator label. In addition there may be further subsidiary labels required to operate and track various services. The network edge label identifies the physical port that provides access to the network and the user label identifies a specific user who is using services across the network (in some cases these two labels may be bound together). The service operator label identifies the physical port that provides access to the service/application provider.

network resource provisioning: This SE manages the updating of the network topology and resource database when a new network element is added to the MUSE network domain. This includes the addition of customer premise domain interface devices and new service/application domain interface resources. This enabler may include auto-discovery, configuration and element testing. When a customer premise domain device is added this enabler becomes focussed into the more specific topic of CPE auto-configuration.

media transcoding: The real-time translation from codec-format A to codec-format B. If no common codec during call establishment then media transcoding will be necessary where the enabler translates the RTP payload sent by one subscriber into a codec-format understood by the recipient.

encrypted QoS stream support: This enabler provides QoS support for encrypted streams such as that required by video under DRM conditions.

conferencing: Provides audio/video mixing capability to provide a single output signal from multiple input signals. This allows all participants in a conferencing scenario to hear/view all other participants.

legal intercept: Historically, legal intercept of telephone PSTN conversations (wiretapping) has been a well-defined process. Such capability must also be provided for voice conversations from within a VoIP/packet network environment.

signalling awareness: Signalling protocols are used to establish real-time VoIP calls which requires the set-up and signalling of media streams (RTP traffic) over IP therefore the MUSE domain must be capable to parse/modify signalling packets. Four levels of signalling awareness are defined:

- no signalling awareness
- SIP firewalling
- basic signalling proxy functionality (signalling only)
- full proxy functionality

stream oriented firewalling: Basic network security is provided by the security enabler. In addition, stream oriented firewalling can provide a value add level of protection against intruders and DoS attacks. Protection is typically realised by a firewall operating upon different layers of the protocol stack. Stream oriented firewalling operates upon RTP traffic.

emergency call support: provides the same level of emergency call support as the PSTN. This should include:

- support for legacy emergency operator services interfaces e.g. MF and SS7
- support for lifeline support where this is a regulatory requirement
- provision of location information to enable the location of a caller's physical location to be determined

APPENDIX II NOTES FOR COMPLETING QFD CELL WEIGHTINGS

Relevance Factor

The 'relevance factor' matrix is designed to provide a level of operator specialisation within the QFD process. It is envisaged that this matrix will be need to be edited by each operator since it adds a relevance to each 'goodness factor' that will be specific to each operator. Inserting a weighting factor in each cell that outlines the operators' view on the importance (relevance) of each goodness factor for each SE completes this matrix. For example, a platform option may provide an efficient and ideal implementation of a particular SE aspect - and as such it receives a high weighting with regards to goodness factor score. However, with regards to an individual operator's perspective, this particular functionality of the SE could be irrelevant to their overall network implementation objectives and therefore a low weighting factor value is given so that the overall score for a particular platform is not skewed by irrelevant network options and features.

Goodness Factor

It was found to be quite difficult to actually assign a weight to each cell with regards to completing the goodness factor measures for each SE. The following descriptions approximate the thought processes/questions that had to be asked with regards to performing this task. Therefore on a platform-by-platform basis the following questions were considered:

- *Scalability*. Is this SE scalable on this platform? The better the scalability the higher the score.
- *CAPEX*. Would one need to buy additional hardware/software to implement this SE onto an existing platform? High score if no additional costs.
- *OPEX*. Are there be any additional running costs for this SE on top of what one would need to provide anyway? High score if no additional costs.
- *Migration cost*. Does implementing this SE allow a migration plan/path that we already intend to follow? If so give it a high score, if not then mark it down depending upon how troublesome it is perceived to be
- *OSS cost*. Would this SE going to require much OSS support on top of what is required anyway? If so give it a low score.
- *Configuration cost*. Does this SE require a lot or a little of configuration effort? If a lot give it a low score.
- *Timescale (for implementation)*. Can this SE be implemented and be useable within the timescales required? If longer than required then the lower the score.
- *Regulatory impact*. Will the regulator have any involvement with this particular SE within its perceived service lifetime? The more involvement with the regulator then the lower the score.
- *Security*. How inherently good is this platform at providing security for this SE? If good then give a high score.

For cases where a particular SE could not be supported by a particular platform option at all then this is identified with a zero score.

APPENDIX III EXAMPLE RELEVANCE FACTOR MATRIX

		Relevance Factor									
service enablers	priority	SCALABILITY	CAPEX	OPEX	Migration	OSS	Config.	Timescale	Regulation	Security	
		point-to-POP QOS flow	9.00	9	1	9	9	9	9	9	1
peer to peer	9.00	1	9	3	1	3	0	3	1	1	
charging	3.00	9	9	9	3	9	9	9	3	9	
service performance monitoring	1.00	3	3	9	3	3	9	9	3	9	
service testing	1.00	3	3	3	1	9	0	9	3	9	
security	3.00	9	3	9	9	9	9	9	9	9	
multicasting	3.00	3	3	3	1	3	3	1	1	1	
content caching/network storage	3.00	1	3	3	1	3	1	1	0	3	
labelling	9.00	1	1	9	3	9	3	3	1	3	
network resource provisioning	1.00	9	3	3	9	9	9	9	1	1	
media transcoding	0.00	1	1	1	1	1	1	1	1	1	
encrypted stream support	0.00	1	3	3	1	1	3	3	3	1	
conferencing	1.00	1	3	9	1	3	9	3	1	9	
legal intercept	0.00	1	3	3	3	3	9	9	9	9	
signaling awareness	3.00	9	3	3	1	3	0	3	0	3	
stream oriented firewalling	3.00	3	3	3	1	1	9	3	1	9	
emergency call support	9.00	3	3	9	9	3	1	9	3	9	

APPENDIX IV EXAMPLE GOODNESS FACTOR MATRIX

Platform Option

	SCALABILITY	CAPEX	OPEX	Migration	OSS	Config.	Timescale	Regulation	Security
point-to-POP QOS flow	3.00	3.00	3.00	1.00	1.00	3.00	3.00	3.00	1.00
peer to peer	9.00	9.00	9.00	3.00	3.00	9.00	9.00	9.00	1.00
charging	3.00	3.00	3.00	1.00	3.00	3.00	3.00	3.00	0.00
service performance monitoring	1.00	3.00	3.00	3.00	3.00	1.00	3.00	3.00	1.00
service testing	1.00	3.00	3.00	3.00	3.00	1.00	3.00	3.00	1.00
security	1.00	3.00	1.00	1.00	3.00	1.00	3.00	1.00	1.00
multicasting	1.00	3.00	1.00	3.00	3.00	3.00	3.00	1.00	3.00
content caching/network storage	9.00	3.00	3.00	9.00	3.00	3.00	9.00	3.00	1.00
labelling	1.00	3.00	1.00	3.00	1.00	1.00	3.00	3.00	3.00
network resource provisioning	1.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	1.00
media transcoding	3.00	3.00	1.00	3.00	3.00	3.00	3.00	3.00	1.00
encrypted QOS stream support	1.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	9.00
conferencing	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00
legal intercept	1.00	3.00	1.00	3.00	3.00	1.00	1.00	1.00	1.00
signaling awareness	3.00	1.00	3.00	1.00	3.00	3.00	1.00	3.00	1.00
stream oriented firewalling	3.00	3.00	1.00	1.00	1.00	1.00	3.00	3.00	3.00
emergency call support	9.00	3.00	1.00	3.00	3.00	1.00	1.00	1.00	3.00