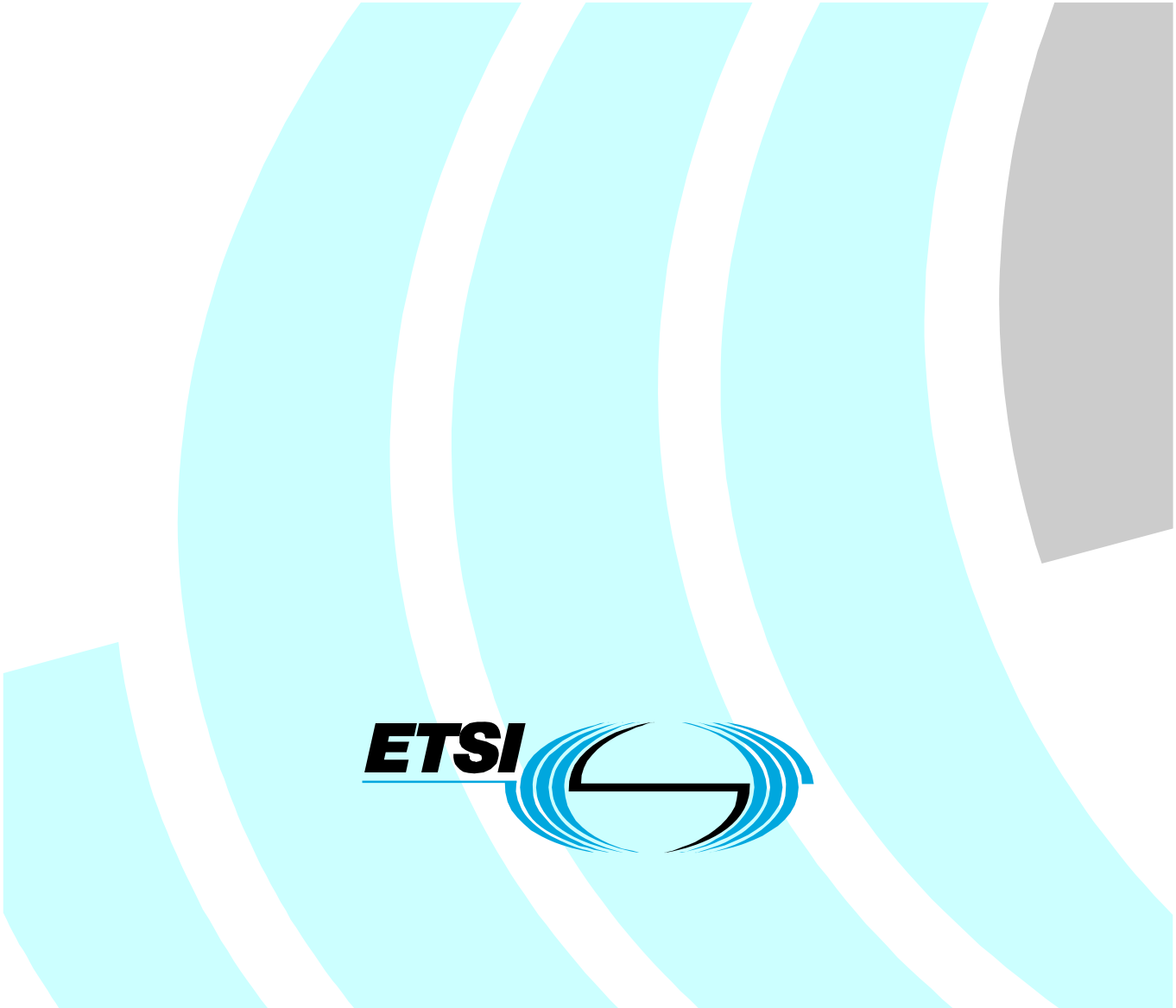
ETSI TR 102 807 V1.1.1 (2010-03)

Technical Report

Speech and multimedia Transmission Quality (STQ); Process description for the transaction view model



Reference
DTR/STQ-00161m

Keywords
mobile, QoS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Report (TR) has been produced by ETSI Technical Committee Speech and multimedia Transmission Quality (STQ).

Introduction

The present document describes, in a process-oriented fashion, how QoS parameters for a particular service can be defined. It treats the relationship between transaction modelling and the point-of-observation view. Finally, it gives an example on how a standard trigger event list can be derived from a QoS parameter definition.

This document accompanies a major undertaking aimed at a more formal definition of QoS parameters. Some early groundwork was laid with the "generic transaction model" which introduced the idea of a hierarchical description of use cases of particular services. In this model, the concept for what was later called "point of observation" had also been drafted. The discussion about the required degree of seamlessness event flows related to QoS parameter was also part of this general evolution process.

The following list describes the elements and development of the whole concept:

- Find a generic structure of service usage descriptions.
- Describe a single case of service usage as the basic transaction for a particular service.
- Describe transactions as consisting of phases.
- Propose a hierarchical model of transaction description, starting with the "user perception" which is starting point and justification for any subsequent, eventually more refined description.
- Recognize that there are different points of observation (PCOs) on which events take place. Require that trigger events used for a particular QoS parameter should, unless there is a grave reason to do otherwise, come from the same point of observation.

In the end, the complete model has the following components

- For each service type, a formally clean definition of the basic transaction for that service, complete with a definition of possible outcomes (results).
- For each transaction type, a clean, hierarchical description which links technical trigger events to relevant user-perception events and QoS parameters description aspects of quality for that service.
- A pool of formally cleanly identified trigger events, and a definition of each event-based QoS parameter as a function of such trigger points.

For the sake of easy and efficient implementation, it is desirable to have technical definitions with a structure as clear as possible; most preferably, this structure should even have the form of a "formal language" allowing for automated creation of technical implementations (e.g. as ASN.1 or XML).

On the other hand, the "primary directive" for any QoS parameter is that it should represent "real user" perception of an aspect of service quality. In other words, every QoS parameter should justify its existence by a clear relationship to such user perception.

Unfortunately, formally strict systems have a tendency to become quite unreadable, or lose their easiness and elegance by requiring a big overhead of rules to enable correct usage. .Therefore, it is a real challenge is to reach both goals of formal strictness and clear relationship to user perception simultaneously. Purpose of this article is to show a way how this can be done.

One should be aware, however, that there are also some primarily non-technical aspects which have to be considered:

- The underlying formal structure needs to work for all existing services, and will always be challenged with the emergence of new services and their QoS parameters. It should be expected that therefore the methodology itself will also evolve with time.
- Existing definitions are deeply entrenched, in processes and products and will produce inertia at best, and most probably resistance, when it comes to changing "external" properties such as names or technical definitions. Also, there is no immediate benefit of changes made here. Therefore, it is likely that there will always be "old layers" of definitions and specifications.

1 Scope

The present document describes underlying concepts of formal description of QoS parameters for particular services, and the way used by a task force within the STQ MOBILE working group to evolve the standards by a more formal trigger point and QoS parameter definition.

The present document presents a process-oriented method to:

- Build a full framework of transaction definition, trigger event definition, and QoS parameter definition for a (fictive) new service or to evolve a description for an existing service to a more formal shape
- Create trigger event lists from existing QoS parameter definitions, with Web Radio service as an example

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific.

- For a specific reference, subsequent revisions do not apply.
- Non-specific reference may be made only to a complete document or a part thereof and only in the following cases:
  - if it is accepted that it will be possible to use all future changes of the referenced document for the purposes of the referring document;
  - for informative references.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

The following referenced documents are indispensable for the application of the present document. For dated references, only the edition cited applies. For non-specific references, the latest edition of the referenced document (including any amendments) applies.

Not applicable.

2.2 Informative references

The following referenced documents are not essential to the use of the present document but they assist the user with regard to a particular subject area**.** For non-specific references, the latest version of the referenced document (including any amendments) applies.

[i.1] ETSI TS 102 250-7: "Speech and multimedia Transmission Quality (STQ); QoS aspects for popular services in GSM and 3G networks; Part 7: Network based Quality of Service measurements".

[i.2] ITU-T Recommendation X.290: "OSI conformance testing methodology and framework for protocol Recommendations for ITU-T applications - General concepts".

[i.3] ETSI TS 102 250-2: "Speech and multimedia Transmission Quality (STQ); QoS aspects for popular services in GSM and 3G networks; Part 2: Definition of Quality of Service parameters and their computation".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

phase: single component within a transaction having a clearly defined start and success criterion

transaction: complete sequence of phases which makes up a meaningful single activity from the customer's point of view (example: speech call, ftp download)

trigger event: event used for definition of QoS parameters

EXAMPLE: Typical trigger events can be the reception of a protocol message in a protocol layer, or starting an action done by a user or a machine.

trigger point: point in time when a trigger event occurs

NOTE: This may contain additional information.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

EPG	Electronic Program Guide
PCO	Point of Control and Observation
QoS	Quality of Service
SAP	Service Access Point
TCP/IP	Transmission Control Protocol/Internet Protocol
TP	Trigger Point

4 Deriving standardized triggers from existing QoS parameter definitions

For preparation, a list structure is prepared having the following columns:

Field	Description
TP ID	Trigger point ID.
PCO	PCO where the trigger event can be observed.
Scenario	This identifies the event flow this QoS parameter belongs to (relevant for QoS parameter sets which belong to different scenarios; the case of PoC is a good example where multiple scenarios exist). Not required if all QoS parameter belong to the same scenario.
Phase	Identifies the phase the respective QoS parameter belongs to. See the GTM model description. For "old" QoS parameter where phase has not been directly identified, this field can be treated as optional.
Type	Purpose of this field is to control automatic creation of QoS parameter by selection a pre-defined pattern of trigger point processing. Currently three different types are identified: **Ratio** (tbd if further differentiation into Success Ratio and Failure Ratio is useful) Basic processing: Start Trigger is "Try" element; Stop trigger is "Success" element). **Time**: Basic processing: Time for each transaction is difference between timestamps of stop and start trigger provided both are valid (i.e. time is valid only for successful transaction). **Time Window**: Start and Stop triggers denote the time window in which a third data entity should be collected (e.g. sample MOS for audio speech quality).
TP def, customer view	This links the technical trigger point to a meaningful event on the customer-view plane.
TP def, technical view	Technical definition of the trigger point in terms of the respective PCO.
Ref: Ratio	This fields should contain the clause numbers in the source document (e.g. TS 102 250-2 [i.3]. NOTE 1: Document version identifier is also required in case that clause numbers vary within document versions). NOTE 2: This set of fields is meant to provide the easiest possible way to access the original QoS parameter definition for editing and reference purposes.
Ref: Time	
Ref.: Time Window	

This list is used to receive all the information taken from the specification document.

The associated process is:

1) For each QoS parameter:

   1.1 Create at least two list rows with respective information from the parameter definition (user perception and technical definition of start and end (success) trigger).

   1.2 Identify, for each row, the scenario to which the respective parameter belongs.

   1.3 Identify the point of observation associated to the technical trigger.

2) After completion of Step 1 for all QoS parameters, post-process the list created by identifying duplicate trigger points. Create trigger point IDs for every unique trigger point.

3) Create a list of all trigger points derived in Step 2 (this can e.g. be done implicitly by appropriate automatic processing of the list).

4) (prospective) Create a new version of the QoS parameter definitions where the parameter is described in terms of trigger point IDs).

5 Defining a new set of QoS parameters and trigger events for a service

5.1 General Rules

In order to decompose the use case(s) of a service into an appropriate hierarchy of transactions, finding suitable QoS parameters and the underlying events in a systematic way, a **top-down** approach has to be applied. Such a systematic top-down process description is the main topic of the "transaction view model".

On the other hand, hierarchies of *definitions* are by nature always **bottom-up**. E.g. in order to define a QoS parameter in terms of certain events, these events have to be defined beforehand.

Thus, the whole process of creating appropriate definitions for (event-driven) QoS metrics of a new service may be summarized as first identifying a transaction hierarchy in a top-down decomposition and then ascending the *same* hierarchy, creating the desired definitions bottom-up.

At the bottom of the definition hierarchy are the *events*, which therefore require the most careful identification and (formal as well as comprehensive) description. For any particular event, we should distinguish between:

1) Its ***motivation*** which means its description *from the user perspective*, or at least how it is *related* to the user perspective. This description is typically given in terms of **human interactions** ("press button" etc.).

2) Its ***abstract definition***. It describes the event at roughly the same abstraction level as the *motivation*. However, instead of referring to human interactions, appropriate technical terms are used, e.g. the names of standardized procedures which either emulate the user behaviour as closely as possible, or are directly triggered by the relevant user interaction. Yet, this definition is *abstract* in the sense that it does ***not*** employ any protocol messages or service primitives used in the implementation of such a procedure at a specific interface or reference point.

3) Its ***specific realizations*** at different PCOs. These realizations are usually described by specifying which particular protocol message or service primitive is sent/received at the respective PCO. The PCO itself is identified by a standardized *interface* or *reference point* **and** a particular *SAP* at that interface / reference point (see [i.2]).

All three elements are required for a complete event definition, the third one being understood to be extensible by adding different PCOs.

NOTE: In the context of the present document, the term *control* in PCO should not be understood as implying a strict requirement that sending of a message or service primitive has to be performed *actively* by the measurement system at that PCO (although this might be preferable in modelling user behaviour). For the purpose of defining a specific event realization it does not matter whether this event realization is actively triggered or just observed.

The characterization of an event by its abstract definition is required to be *unique* except for point 3. Thus, a unique, symbolic *ID* is associated with the abstract definition. This ID is used to refer to the event higher up the definition hierarchy, e.g. in QoS parameter definitions. In this sense, these definitions also become "abstract definitions". Along with a definition that uses a particular event it should be specified which of its specific realizations are permissible in that context.

Assigning a unique identifier to each identified PCO as well, the specific realizations of an event can easily be labelled by concatenating the event ID and the PCO ID, separated by a dot.

Initially, the PCO is chosen to be:

- as close as possible to the user perception;
- the same for begin and end of a (sub-)transaction (phase);
- the same or at least equivalent (with respect to user perception) for *all* phases of a use case.

Deviations from these rules have to be explained explicitly.

Later on, other PCOs may be added for different purposes (e.g. network based measurements, see [i.1]).

5.2 Detailed Process Description

The process of defining a new set of QoS parameters and trigger events for a service contains the following steps:

1) Define the event flow and the possible results of a single transaction of the target service from customer's point of view.

2) Define and name the phases of the transaction and their success criteria. A phase name should describe the target or desired result of that phase; this allows QoS parameter name creation by providing a clear reference (e.g. "call establishment"…"call establishment success ratio" Create clarity about the degree of seamlessness.

Ideally, the TA is described by phases seamlessly, meaning that the "success" condition for a phase is equal to the initial "try" condition for the next phase.

3) Identify trigger events defining the "borders" (begin/end) of the sub-transactions (phases) and find appropriate, unique names and *abstract definitions* (see clause 5.1) for these events. These definitions should resemble the user perception of the respective phase border as closely as possible.
Make sure that in the case of two adjacent ("seamless") phases, only *one* event is defined for the border between them. To each identified, unique event also assign a unique, symbolic ID.
In case a phase or trigger does not have an equivalent in the customer perception plane, explain why this phase is a desirable one nevertheless, and make a clear reference to the related customer perception.

4) Define the QoS parameters for the phases identified in Step 3 Name them, using the phase names (e.g. "<phase name> success ratio")

5) Specify PCOs where the events defined in Step 3 are to be triggered or observed, by identifying an appropriate SAP at a standardized interface or reference point. Assign a unique, symbolic ID to each PCO (see clause 5.1 for rules for choosing appropriate PCOs).

6) Define the specific realizations of the trigger events defined in Step 3 at the respective PCOs specified in Step 5, usually by referring to the sending / reception of a particular standardized protocol message or service primitive. The unique symbolic ID for such a "specific event" is constructed as described in clause 5.1.

7) (prospective) Create a new version of the QoS parameter definitions where the parameter is described in terms of trigger event IDs.

6 Example: Using the process for Web Radio QoS parameters

The following example illustrates the process for an existing service according to the definition in clause 4.

Step 1:

Create a table containing all trigger points currently used for defining the QoS parameters for a service.

Remarks:

- The "Ref. Ratio", "Ref. Time" and Ref. Time Window" columns are provided for easy reference to the clauses of the source standard.
- The entry "TCP/IP" in the "PCO" column is to be understood as "SAP between application and transport layer at the *R* reference point".
- For the sake of simplicity, the column "TP def, technical view" comprises both *abstract definition* and *specific realization* (at the indicated PCO) of the respective event.

Result: (placeholder column for TP-ID inserted).

TP ID	PCO	Scenario	Phase	Type	TP def, customer view	TP def, technical view	Ref: Ratio	Ref: Time	Ref: Time Window
	TCP/IP	1	EPG Retrieval	Start	Customer accesses Web Radio EPG.	HTTP GET on EPG URL	6.9.7	6.9.8	
	TCP/IP	1	EPG Retrieval	Success	EPG content successfully received	Successful reception of EPG content (HTTP 200 OK, eventually followed by additional blocks).	6.9.7	6.9.8	
	TCP/IP	1	Tune-in	Start	Attempt to retrieve tune-in information	Obtain tune-in information via a HTTP GET to a location obtained from EPG.	6.9.9	6.9.10	
	TCP/IP	1	Tune-in	Success	Receive tune-in information	Successful reception of tune-in information (HTTP 200 OK, eventually followed by additional blocks).	6.9.9	6.9.10	
	TCP/IP	1	Reproduction set-up	Start	Attempt to retrieve audio stream	Attempt to retrieve audio content from stream server listed in tune-in information (HTTP GET).	6.9.11	6.9.12	
	TCP/IP	1	Reproduction set-up	Success	Indication that player starts buffering (may not be visible in all players)	Reception of first block of content (audio data).	6.9.11	6.9.12	
	TCP/IP	1	Reproduction	Start	Attempt to retrieve audio stream	Attempt to retrieve audio content from stream server listed in tune-in information (HTTP GET).	6.9.13		

TP ID	PCO	Scenario	Phase	Type	TP def, customer view	TP def, technical view	Ref: Ratio	Ref: Time	Ref: Time Window
	TCP/IP	1	Reproduction	Success	Reach the end of intended stream playback time without break in IP connection	Reach the end of intended stream playback time without break in IP connection	6.9.13		
	Audio output	1	Audio Quality	Start of time window	Attempt to retrieve audio stream				6.9.14
	Audio output	1	Audio Quality	end of time window	Reach the end of intended stream playback time without break in IP connection				6.9.14

Step 2: Identify unique trigger event IDs

Step 3: Create a list of all trigger events derived in Step 2

Result:

TP ID	PCO	Scenario	Phase	Type	TP def, customer view	TP def, technical view
WR1	TCP/IP	1	EPG Retrieval	Start	Customer accesses Web Radio EPG	HTTP GET on EPG URL
WR2	TCP/IP	1	EPG Retrieval	Success	EPG content successfully received	Successful reception of EPG content (HTTP 200 OK, eventually followed by additional blocks)
WR3	TCP/IP	1	Tune-in	Start	Attempt to retrieve tune-in information	Obtain tune-in information via a HTTP GET to a location obtained from EPG
WR4	TCP/IP	1	Tune-in	Success	Receive tune-in information	Successful reception of tune-in information (HTTP 200 OK, eventually followed by additional blocks)
WR5	TCP/IP	1	Reproduction set-up	Start	Attempt to retrieve audio stream	Attempt to retrieve audio content from stream server listed in tune-in information (HTTP GET)
WR6	TCP/IP	1	Reproduction set-up	Success	Indication that player starts buffering (may not be visible in all players)	Reception of first block of content (audio data)
WR7	TCP/IP	1	Reproduction	Success	Reach the end of intended stream playback time without break in IP connection	Reach the end of intended stream playback time without break in IP connection

With the trigger event IDs inserted, initial table becomes:

TP ID	PCO	Scenario	Phase	Type	TP def, customer view	TP def, technical view	Ref: Ratio	Ref: Time	Ref: Time Window
WR1	TCP/IP	1	EPG Retrieval	Start	Customer accesses Web Radio EPG	HTTP GET on EPG URL	6.9.7	6.9.8	
WR2	TCP/IP	1	EPG Retrieval	Success	EPG content successfully received	Successful reception of EPG content (HTTP 200 OK, eventually followed by additional blocks)	6.9.7	6.9.8	
WR3	TCP/IP	1	Tune-in	Start	Attempt to retrieve tune-in information	Obtain tune-in information via a HTTP GET to a location obtained from EPG	6.9.9	6.9.10	
WR4	TCP/IP	1	Tune-in	Success	Receive tune-in information	Successful reception of tune-in information (HTTP 200 OK, eventually followed by additional blocks)	6.9.9	6.9.10	
WR5	TCP/IP	1	Reproduction set-up	Start	Attempt to retrieve audio stream	Attempt to retrieve audio content from stream server listed in tune-in information (HTTP GET)	6.9.11	6.9.12	
WR6	TCP/IP	1	Reproduction set-up	Success	Indication that player starts buffering (may not be visible in all players)	Reception of first block of content (audio data)	6.9.11	6.9.12	
WR5	TCP/IP	1	Reproduction	Start	Attempt to retrieve audio stream	Attempt to retrieve audio content from stream server listed in tune-in information (HTTP GET)	6.9.13	6.9.14	

TP ID	PCO	Scenario	Phase	Type	TP def, customer view	TP def, technical view	Ref: Ratio	Ref: Time	Ref: Time Window
WR7	TCP/IP	1	Reproduction	Success	Reach the end of intended stream playback time without break in IP connection	Reach the end of intended stream playback time without break in IP connection	6.9.13	6.9.14 (check)	
WR5	Audio output	1	Audio Quality	Start of time window	Attempt to retrieve audio stream				6.9.14
WR7	Audio output	1	Audio Quality	end of time window	Reach the end of intended stream playback time without break in IP connection				6.9.14

Step 4:

Express the QoS parameters using the trigger event IDs.

Please note the clause numbers given in the "Ref. Ratio", "Ref. Time" and Ref. Time Window" columns of above table.

Old definition:

6.9.7 Web Radio EPG Retrieval Failure Ratio [%]

6.9.7.1 Abstract Definition

This parameter denotes the probability that a subscriber cannot access the Web Radio EPG successfully.

6.9.7.2 Abstract Equation

$$\text{Web Radio EPG Retrieval Failure Ratio}[\%] = \frac{\text{unsuccessful attempts to access the EPG}}{\text{all attempts to access the EPG}} \times 100$$

6.9.7.3 Trigger Points

Event from abstract equation	Trigger point from customer's point of view	Technical description/protocol part
EPG retrieval attempt	Start: Customer accesses Web Radio EPG.	Start: HTTP GET on EPG URL.
Successful attempt	Stop: EPG content successfully received.	Stop: Successful reception of EPG content (HTTP 200 OK, eventually followed by additional blocks).
Unsuccessful attempt	Stop trigger point not reached.	

6.9.8 Web Radio EPG Retrieval Time [s]

6.9.8.1 Abstract Definition

This parameter describes the time period needed to access the Web Radio EPG successfully.

6.9.8.2 Abstract Equation

$$\text{Web Radio EPG Retrieval Time}[s] = \left(t_{Stop\_ER} - t_{Start\_ER}\right)[s]$$

6.9.8.3 Trigger Points

Event from abstract equation	Trigger point from customer's point of view	Technical description/protocol part
t_{Start_ER}: Time of EPG retrieval attempt	Start: Customer accesses Web Radio EPG.	Start: Time of sending the HTTP GET on EPG URL.
t_{Stop_ER}: Time of successful EPG retrieval attempt	Stop: EPG content successfully received.	Stop: Time of successful reception of EPG content (HTTP 200 OK, eventually followed by additional blocks).

New definition

6.9.7 Web Radio EPG Retrieval Failure Ratio [%]

6.9.7.1 Abstract Definition

This parameter denotes the probability that a subscriber cannot access the Web Radio EPG successfully.

6.9.7.2 Abstract Equation

$$\text{Web Radio EPG Retrieval Failure Ratio}[\%] = \frac{\text{count(WR2)} - \text{count(WR1)}}{\text{count(WR1)}} \times 100$$

6.9.7.3 Trigger Points

Replaced by a reference to the global or service-wise trigger event table.

6.9.8 Web Radio EPG Retrieval Time [s]

6.9.8.1 Abstract Definition

This parameter describes the time period needed to access the Web Radio EPG successfully.

6.9.8.2 Abstract Equation

$$\text{Web Radio EPG Retrieval Time}[s] = (t_{WR2} - t_{WR1})[s]$$

Valid for transactions where WR2 was reached.

6.9.8.3 Trigger Points

Replaced by a reference to the global or service-wise trigger event table.

History

Document history		
V1.1.1	March 2010	Publication