Question(s):	3	**Meeting, date:**	Melbourne, 28th Feb – 4th Mar 2005
Study Group:	16 **Working Party**: 2	**Intended type of document (R-C-D-TD)**:	D
Source:	Editor		
Title:	**Draft H.Sup.BHCoA: Decomposed Gateways – Control Load Quantum – Busy Hour Context Attempts (BHC_OA) and Busy Hour Session Attempts (BHSA).** (Ed. 0.2)		

Contact:	ALBRECHT SCHWARZ ALCATEL Germany	Tel: Fax: Email:	+49-711-821-41425 +49-711-821-40247 Albrecht.Schwarz@alcatel.de
Contact:			

Please don't change the structure of this table, just insert the necessary information.

AVD-2699: Draft *H-series Supplement* **H.Sup.BHCoA**

Summary:

This is the output document from the November 2004 Geneva meeting. There are no input changes from the editor at this meeting.

Document was mapped on an ITU-T Rec. template.

Draft ITU-T Recommendation H.Sup.BHCoA

Decomposed Gateways:

Control Load Quantum Recommendations for Two-Party Communication Services – Busy Hour Context Attempts (BHC_OA) and Busy Hour Session Attempts (BHSA)

CONTENTS

Page

Summary

<Mandatory material>

Keywords

Performance, NGN, Load Control, Traffic Model

Introduction

<Optional – This clause should appear only if it contains information different from Scope and Summary>

1 Paradigm Shift – Motivation

The successful control load quantum in traditional circuit-switched networks (CSN): *Busy Hour Call Attempts* (BHCA), for a time unit 'hour,' respectively denoted as *Call Attempts Per Second* (CAPS), for a time unit 'second', -, as well as the corresponding control performance quantum *Busy Hour Call Completions* (BHCC), respectively denoted as *Call Completions Per Second* (CCPS), are misleading in H.248 systems.

NOTE – "Traditional" refers to the call definition and control load understanding according ITU-T Q.543 [5], the control performance framework for digital switching systems.

An H.248 based packet-switched network (PSN) is (1) architecturally different in comparison to legacy SCNs, particularly in the following three principal directions:

- *decomposed control structure* into H.248 MGC and H.248 MG, whereby the main vertical control processing portion is part of the 'controller',
- *server* approach, by centralising the distributed control of many legacy switching systems into a few number of session control servers, and
- the typical *1:N relation* with regard the MGC:MG ratio.

It is obvious that any re-use of legacy terminology requires a careful handling and common understanding.

NOTE – The reuse of 'BHCA', 'CAPS', etc. is principally possible in H.248 environments, particularly in the scope of PSTN/N-ISDN service emulation. But it isn't recommended, particularly due to potential misunderstandings, and the PSTN/ISDN extending scope of H.248.

Additionally the architectural motivation for the network is based on a technical incentive which requires a "BHCA mapping" on H.248 systems: implying that a prerequisite knowledge of (2) load control and overload protection mechanisms is a basis for understanding the underlying control load quantum. For example, the H.248.11 *Overload Control Package* defines a tight co-operation principle between a MGC and associated MGs; H.248.11 applies the same principles to load quantification.

(3) A third aspect concerns relating the pure Packet-to-Packet (Pa2Pa) MG application with session control protocols at a MGC level, i.e., without the presence of a direct *call* relationship (e.g., 3GPP IP Multimedia Subsystem).

1.1 Purpose

The aim of this document is to introduce BHC_OA (Busy Hour **Context** Attempts) as a baseline for control load metrics for H.248 systems, and also to introduce a definition of a control load quantum based on a "Basic H.248 Context".

The main purposes of this Recommendation are:

- Definition of performance engineering parameters relevant for control processing in H.248 systems,
- Definition of performance design objectives relevant for H.248 systems,
- Exemplary processing capacity calculations,

- Others.

1.2 Scope and Initial Objectives

The objectives of the current edition are:

- Identification of the need for an extended performance engineering framework in the context of decomposed control platforms,
- Introduction of a new terminology (such as BHC$_o$A, BHSA, effective multiplication factor),
- Initial proposal of a control processing model,
- Initial proposal of H.248 Context-based performance classes, and
- Basic relations of load and performance parameters according the defined performance framework.

The initial scope is to achieve consensus on a qualitative basis, the natural next step would be then to commence quantitative performance investigations.

1.3 Linearity Assumption

Linearity is assumed. Also first-order traffic engineering calculations frequently use linearization approximations, particularly in the context of control load estimations (like BHC$_a$A)[1].

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation

2.1 Normative References

[1] *TERMINOLOGY – Proposal for further definitions to Draft H.248 Overload Control Package* (Draft H.248.11). ITU-T SG16, Delayed Contribution T01-SG16-021015-**D-0263**, October 2002.

[2] *Signalling Requirements for the Support of Narrowband Services over Broadband Transport Technologies – Capability Set 2 (CS-2),* ITU-T Technical Report **TRQ.2141.0** (= **Q.Sup31** Supplement 31 to ITU-T Q-series Recommendations), December 2000.

---

[1] E.g., [8]: The assumption of a *linear relationship* between processor *occupancy* and *offered load* (BHCA) holds well in *steady-state*, *fault-free* conditions with a *constant call-type distribution*, up to the designed occupancy level for overload capacity.

[3] *ITU-T Vocabulary: SANCHO Data Base* (ITU-T **S**ector **A**bbreviations and defi**N**itions for a tele**C**ommunications t**H**esaurus **O**riented database). http://www7.itu.int/sancho/

[4] *Terms and Definitions of Traffic Engineering*, ITU-T **E.600**, March 1993.

[5] *Digital Exchange Performance Design Objectives*, ITU-T **Q.543**, March 1993.

[6] *Call Processing Performance for Voice Service in Hybrid IP Networks*. ITU-T **Y.1530**, February 2004.

[7] *Methodology for the Calculation of IMT-2000 Terrestrial Spectrum Requirements*, ITU-R **M.1390**, January 1999.

2.2 Bibliography

[8] *Traffic Calculations in SPC Systems.* VILLAR, J. E.; 8th ITC, November 1976.

2.2.1 ITU-T E-Series

[9] *Traffic Reference Period.* ITU-T **E.492**, 1996.

[10] *Traffic Intensity Measurement Principles.* ITU-T **E.500**, November 1998.

[11] Estimations of Traffic Offered in the Network. ITU-T **E.501**, May 1997.

[12] *Traffic Measurement Requirements for Digital Telecommunication Exchanges.* ITU-T **E.502** (rev. 1), 1992.

[13] *Traffic Measurement Data Analysis.* ITU-T **E.503** (rev. 1), 1992.

[14] *Forecasting New Telecommunication Services.* ITU-T **E.508**, October 1992.

[15] *Network Dimensioning using End-to-End GoS Objectives.* ITU-T **E.529**, May 1997.

[16] *User Demand Modelling.* ITU-T **E.711**, October 1992.

2.2.2 Telcordia

[17] *Generic Requirements for Voice over Packet End-to-End Performance.* **GR-3059-CORE**, March 2000, Telcordia Technologies.

[18] *Switching System Overload Control Generic Requirements.* **TR-NWT-001358**, September 1993, Telcordia Technologies.

[19] *LSSGR: Traffic Capacity and Environment.* **GR-517-CORE**, December 1998, Telcordia Technologies.

3 Terminology and Definitions

3.1 Session versus Call

The telecommunication network specific term “call” is often translated to the term “*session*” for packet-switched connectionless networks (e.g., Internet). The notion of a *session* is also fundamental to IP-based NGN architectures. A **session** extends the traditional notion of a **call** in telecommunication networks. An "H.248 session/call" and the associate creation of an "H.248 Context" is typically triggered by a specific *Call Control Protocol* (e.g., SS7 TUP, SS7 ISUP, BICC, DSS1, H.225/H.245, etc.), or a *Session Control Protocol* (e.g., SIP, SIP-T, NGN-SCP) events. The differentiation between a "call" and a "session" is transparent and is actually not too relevant from an H.248 perspective. Both may be used interchangeable from the Gateway Control Protocol point of view. The key control association is fundamentally the H.248 Context.

NOTE – ITU-T Recommendation E.600 [4] defines the individual terms "*call*", "*call attempt*", and "*busy hour*", primarily in the context of BHC_aA (*Busy Hour Call Attempts*).

In order to avoid confusion with the legacy BHCA definition, it is recommended that the terms BHSA and BHC_oA are used in the context of H.248 systems. That's why the term 'session' is continuously used in this document, too.

3.2 General Definitions

Session rsp. Call	'Session' or 'Call' means a generic term related to the creation, modification and deletion of a **H.248 Context** (in a MG). Normally a qualifier is necessary to make clear the aspect being considered, e.g. session attempt. This definition is motivated by alignment with ITU-T E.600 [4].
Session/Call Attempt	'Session/Call Attempt' means an attempt to achieve the creation of one or more new H.248 Context(s) in the MG. This definition is motivated by alignment with ITU-T E.600 [4].
Load	'Load' means the total number of "… attempts" presented to a MGC respectively to a MG during a given interval of time (i.e. offered load). This definition is motivated by alignment with performance objectives of ITU-T Q.543 [5].
Session Load	→ *see Figure 2*
Context Load	→ *MG Context Load, see Figure 2*

These definitions are illustrated in Figure 2.

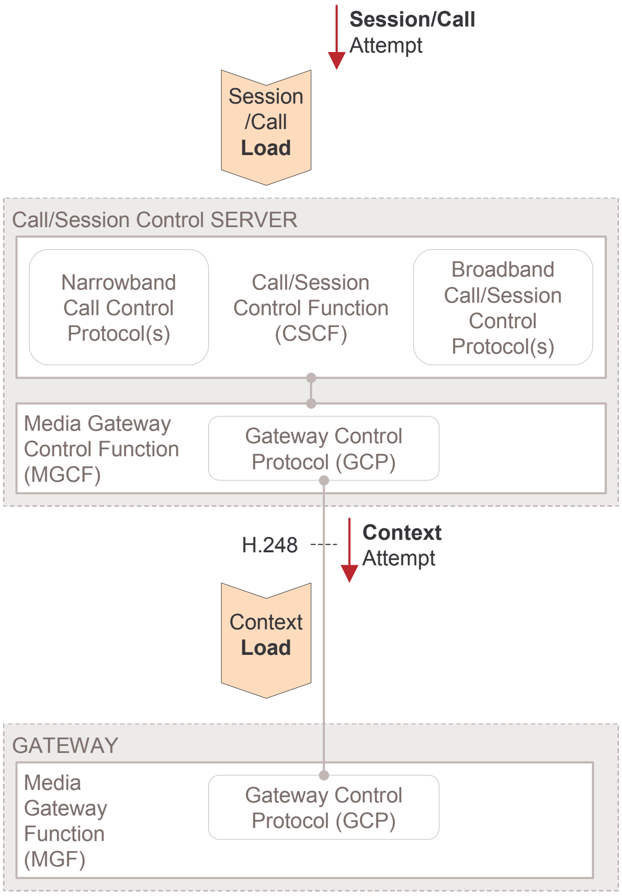

Figure 1: "Context Attempts" and generated "Context Load"

3.3 BHxA related Definitions

The following table provides a list of generic, BHxA-related load parameters, and corresponding technology-specific example parameters.

BHC_aA $BHC_{Q.543}A$ (briefly BHCA)	Busy Hour **Call** Attempts Note: 'call' = PSTN or N-ISDN call according ITU-T Q.543
BHC_bA $BHC_{Q.19XX}A$	Busy Hour **Bearer Connection** Attempts Note: 'bearer connection' = connection controlled by ITU-T **Q.19XX BICC CS1, CS2, CS3 Bearer Control Function (BCF)**
BHC_oA $BHC_{H.248}A$	Busy Hour **Context** Attempts Note: 'context' = ITU-T **H.248 Context**

BHC_oA_{MG}	Busy Hour **Context** Attempts on *Media Gateway* level Note: 'context' = **Media Gateway Context** for either one of following **H.248-based** MG types: • IETF RFC 3525/ITU-T H.248.1 Media Gateway (MG), • ITU-T Q.1950 Bearer Interworking Function (BIWF) or Media Gateway Unit (MGU)[2], • 3GPP 29.232 Circuit-Switched Media Gateway Function (CS-MGW), • 3GPP 29.332 IP Multimedia Media Gateway Function (IM-MGW), • ITU-T "SG11" Packet Gateway Function (PGF), • ITU-T J.171 Annex B Media Gateway (MG)[3]
BHC_oA_{MGC}	Busy Hour **Context** Attempts on *Media Gateway Controller* level Note: 'context' = **Media Gateway Controller Context** for either one of following H.248-based MGC types: • IETF RFC 3525/ITU-T H.248.1 Media Gateway Controller (MGC), • ITU-T Q.1950 Call Service Function (CSF), • 3GPP 29.232 Mobile Switching Center Server (MSC Server)[4], • 3GPP 29.332 Media Gateway Control Function (MGCF), • ITU-T "SG11" Packet Gateway Control Function (PGCF) • ITU-T J.171 Annex B Media Gateway Controller (MGC)
BHSA	Busy Hour **Session** Attempts
$BHS_{SIP}A$ $BHSA_{RFC3261,SIP}$	Busy Hour **Session** Attempts Note: 'session' = according IETF RFC 3261 Session Initiation Protocol
$BHS_{SCP}A$ $BHSA_{NGN\text{-}SCP}$	Busy Hour **Session** Attempts Note: 'session' = according Draft ITU-T TRQ.NCAPX NGN Session Control Protocol
$BHS_{SIP}A$ $BHSA_{3GPP,SIP}$	Busy Hour **Session** Attempts Note: 'session' = according 3GPP 24.229 IP Multimedia Call Control Protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP)

NOTE – The difference between Busy Hour Context Attempts on MG level BHC_oA_{MG} and MGC level BHC_oA_{MGC} is illustrated in *Figure 10: Control Processing Model – Load/Performance Chaining*.

The corresponding performance, BHxC-related definitions are appropriate.

Finally, a technical, BHxA-related load parameter is provided which is useful for performance considerations on the MG level:

---

[2] see TRQ.2141.0 [2] Annex C

[3] Reference: ITU-T J.171 Annex B, *IPcablecom Trunking Gateway Control Protocol (TGCP); TGCP Profile 2*, May 2003. The "TGCP Profile 2" is based on H.248, and entitled "TGCP_H248".

[4] e.g., Serving MSC Server, Gateway MSC Server

BHC$_{h,DSP}$A	Busy Hour **Channel** Attempts Note: 'channel' = general resource component type "media conversion unit" (MCU) within a MG; a technical realization for a MCU is a "DSP channel"[5] → a "DSP Channel" is the intra-system segment of a user plane connection (e.g., bearer channel) related with a DSP component

NOTE – The term "mean value" is understood to be the expected value in the probabilistic sense.

4 Abbreviations

ALN	Analog Line (H.248 Termination physical type)
BHC$_a$A	Busy Hour **Call** Attempts
BHC$_b$A	Busy Hour Bearer **Connection** Attempts
BHC$_h$A	Busy Hour **Channel** Attempts → e.g., DSP Channel
BHC$_o$A	Busy Hour **Context** Attempts
BHC$_o$A$_{MGC}$	Busy Hour **Context** Attempts (H.248 Context on **MGC** level)
BHC$_o$A$_{MG}$	Busy Hour **Context** Attempts (H.248 Context on **MG** level)
~~BHCA$_{es}$~~	~~Busy Hour Call Attempts establish traffic (ITU-R M.1390 [7])~~
BHSA	Busy Hour **Session** Attempts
BHSC	Busy Hour Session Completions
BICC	Bearer Independent Call Control
C	H.248 Context
C$_a$HT, CHT	Call Holding Time
C$_o$HT	Context Holding Time
C$_a$APS	Call Attempts Per Second
C$_o$APS	Context Attempts Per Second
C$_a$CPS, CCPS	Call Completions Per Second
C$_o$CPS	Context Completions Per Second
CP	Context Processor (H.248) Control Path (System)
CSCF	Call/Session Control Function
CSN	Circuit-Switched Network (H.246, H.332, Y.1001)
DSP	Digital Signal Processor *(general)* Digital Speech Processor *(North America specific)*
e	FIXTHIS
GCP	Gateway Control Protocol
MCU	Media Conversion Unit

[5] Channel in that sense is the basic "capacity unit" for a digital signal processor in H.248 MG systems.

MEGACOP	Media Gateway Control Protocol (= H.248)
MG	Media Gateway
MGC	Media Gateway Controller
MGF	Media Gateway Function
MGCG	Media Gateway Control Function
MSC	Mobile Switching Center
NGN	Next-Generation Network
Pa2Pa P2P	Packet-to-Packet
Pe2Pe	Peer-to-Peer
PSN	Packet-Switched Network
r	FIXTHIS
SAPS	Session Attempts Per Second
SCN	Switched-Circuit Network (H.247) Switched Communication Network (G.177) Signalling Communication Network (Y.1703) NOTE – SCN and CSN denoting the same thing in context of H.248 systems. Due to above ambiguousness of SCN shall be solely abbreviation 'CSN' used in this document.
SCP	Session Control Protocol
SCPS	Session Completions Per Second
SHT	Session Holding Time
SIP	Session Initiation Protocol
SP	Session Processor
STM	Synchronous Transfer Mode
TDM	Time Division Multiplexing → H.248 Termination for *Synchronous Transfer Mode* (STM) interfaces, i.e., TDM is used to abbreviated *Synchronous Time Division Multiplexing* (STDM) [but not *Asynchronous TDM* (ATDM)].

4.1 Mathematical Symbols

λ	Arrival rate	$[s^{-1}]$	Mean arrival rate of service requests[6]
λ_{CoAPS}	MGC "Context Attempt" rate	$[s^{-1}]$	Mean "Context attempt" rate generated by a MGC for a MG
μ	Service rate	$[s^{-1}]$	Mean service rate of the processing entity[7]

6 E.g., Control plane events: for instance, session initiation messages, call Setup messages, H.248 ADD requests, etc.; User plane events: any type of packet arrivals (e.g., IP packet, MAC frame, ATM cell, AAL2 CPS-Packet, FR frame)

$\mu_{Context}$	Context Service rate	$[s^{-1}]$	Mean service rate per H.248 Context
ρ	Utilization		Mean occupancy of a processing entity
ρ_{CcC}	Utilization factor		Mean occupancy of a processing entity by completing H.248 Contexts
ρ_{CcR}	Utilization factor		Mean occupancy of a processing entity by rejecting H.248 Contexts
ϕ	Throughput rate	$[s^{-1}]$	Mean throughput rate of served requests
$\phi_{Context}$	Throughput rate	$[s^{-1}]$	Mean effective H.248 Context throughput rate
ϕ_{CoBPS}, ϕ_{CoB}	Context Blocking rate	$[s^{-1}]$	Mean rate of blocked H.248 Contexts
ϕ_{CoCPS}, ϕ_{CoC}	Context Completion rate	$[s^{-1}]$	Mean rate of completed H.248 Contexts
ϕ_{CoRPS}, ϕ_{CoR}	Context Rejection rate	$[s^{-1}]$	Mean rate of rejected H.248 Contexts
h_{Co}, $h_{Context}$	Service time	[s]	Mean service time per H.248 Context
h_{CoC}	Service time	[s]	Mean service time per completed H.248 Context
h_{CoR}	Service time	[s]	Mean service time per rejected H.248 Context
A	Offered load	[Erl]	
A_{CP}	Offered load	[Erl]	Mean offered load per Context Processor
B	Blocking probability		
Y	Carried traffic	[Erl]	
Y_{CP}	Carried traffic	[Erl]	Mean carried traffic per Context Processor
Ω	Queue occupancy		Message buffers, etc.
τ	Delay	[s]	Mean delay of a message

4.1.1 Indices

$\ldots_{Co}$ $\ldots_{Context}$	Context	H.248 Context
$\ldots_{CP}$ $\ldots_{ContextProcessor}$	Context Processor	MGC or MG embedded Context Processor
$\ldots_{CoA}$	Context Attempts	Load
$\ldots_{CoC}$	Completed Contexts	Performance: "Goodput"
$\ldots_{CoR}$	Rejected Contexts	Performance: "Badput" (e.g., rejected,

7 Technical realizations: e.g., CPUs, DSPs, IP Forwarding Engine, ATM SAR device, Ethernet switch, etc.

		blocked, discarded Contexts)
...BL	Basic Load	Basic (or background) server load, i.e., the none-H.248 related load
...HL	High Load	
...NL	Nominal Load	Engineered capacity, recommended operating point for a considered resource
...OL	Overload	

NOTE – The attribute 'mean' in system/performance parameter notations characterizes the "time mean" (of the underlying stochastic process). But the purpose of this document is also to provide worst-case estimations for system/performance parameters. These specific requirements will be denoted by an additional *index*.

...min	Minimum	Minimum requirement with regard to worst case assumptions
...max	Maximum	Maximum requirement with regard to worst case assumptions

5 Basic Model for 2-Party Communication Services

The control load quantum shall be based on a basic teleservice, a conversational communication between two session parties.

NOTE – The same principle was applied in PSTN/N-ISDN by using speech telephony service between two calling parties (caller & callee) for "basic call" definitions.

5.1 Network Model

The 2-Party property leads to H.248 Contexts types with two H.248 Terminations. H.248 Context processing is done on an MGC and MG level. The scope of this recommendation is beyond the H.248 Context level, and shall comprise session processing as well. The two technical network elements shall be denoted as *session control server*, and *gateway*. Figure 2 shows that simplified architectural network model.

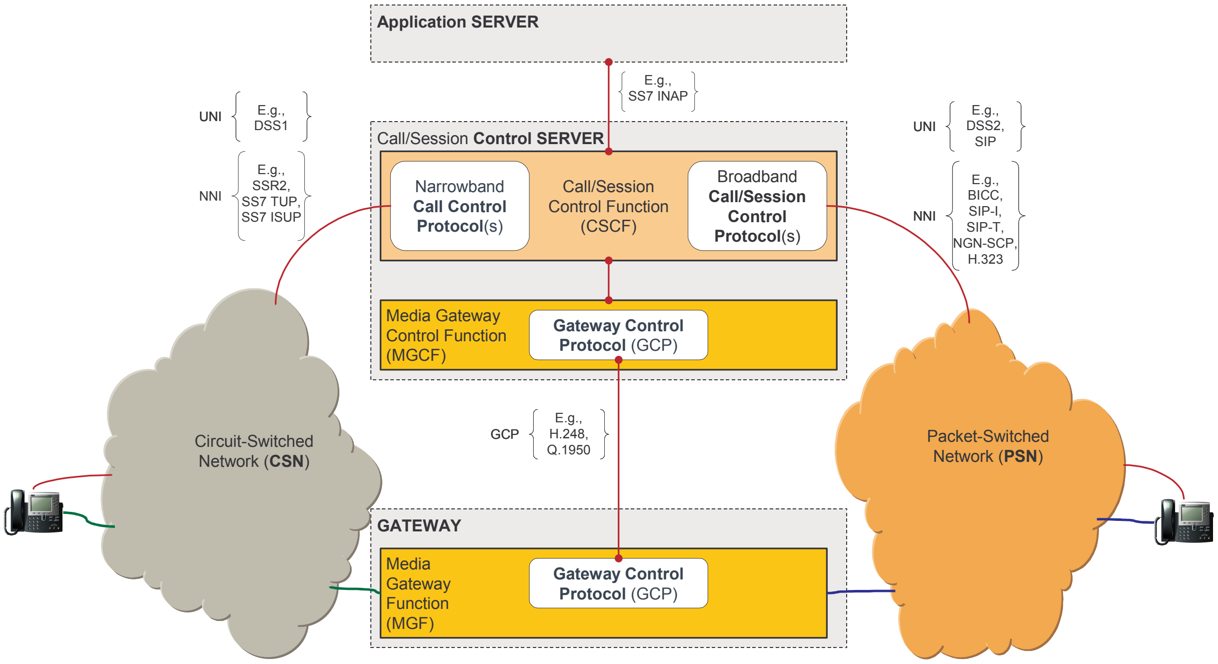

Figure 2: NGN Domains Transport, Control & Application

The dashed boxes are physical network elements (gateway, session control server, application server). The rectangles representing functional entities:

- Media Gateway Function (MGF),
- Media Gateway Control Function (MGCF),
- Call/Session Control Function (CSCF).

NOTE – These functional entities are apparently the most common ones used in various ITU-T, 3GPP, ETSI, etc. NGN models.

The rounded rectangles pointout the three major considered generic control protocols: Gateway Control Protocol (GCP), and call/session control protocols for circuit- and packet-switched networks. The double braces show exemplary control technologies for the various signalling interfaces. Of course, the specific GCP is H.248, and all other H.248-based control interfaces such as Q.1950, 3GPP 29.232, 3GPP 29.332, etc.

NOTE – Other GCP types like IPDC, MGCP, ITU-T J.171 are out of scope.

Out of scope of this performance recommendation is the specific network level (e.g., customer premises equipment domain, access network domain, or core network domain) where the specific H.248 MG may be deployed. Thus, dedicated performance aspects of residential MGs, access MGs, trunking MGs, etc. won’t be considered.

Also out of scope are potential differences between mobile or fixed NGNs.

5.2 Session Variants

5.2.1 Overview

H.248 distinguishes between two basic Termination types: physical (PHY) and ephemeral (EPH). Figure 3 summarizes the three resulting Context types for two party communication services.

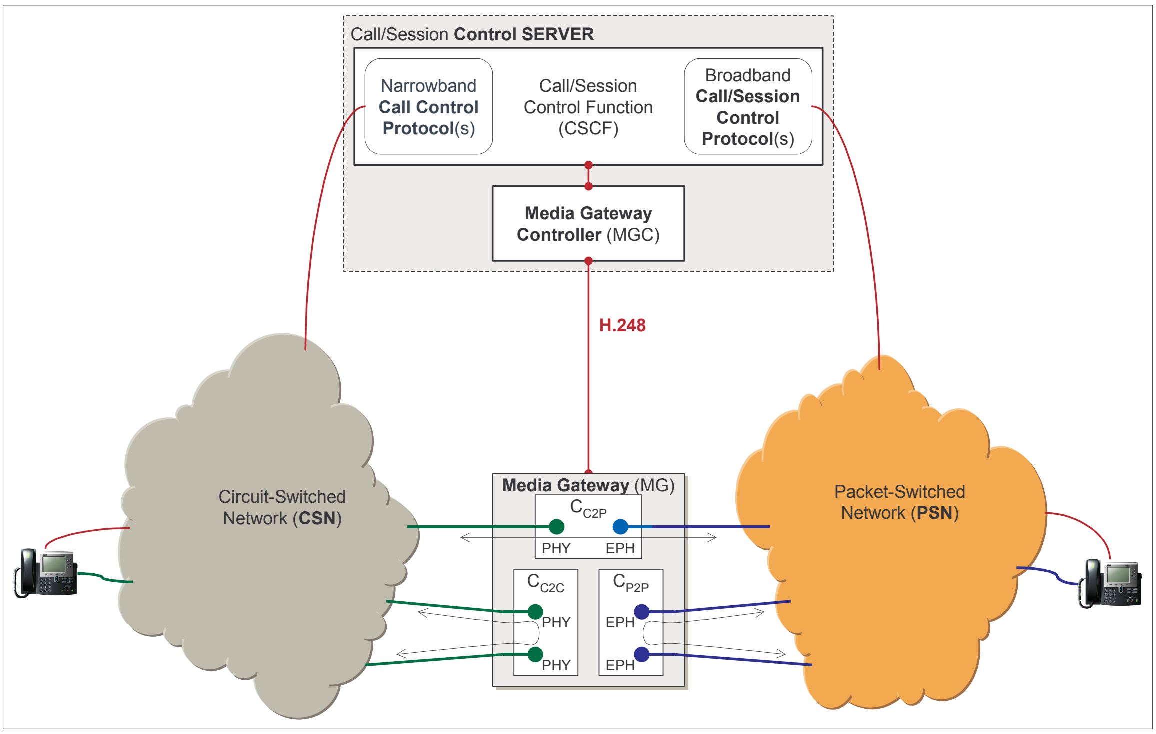

Figure 3: Session Categories – **Overview**

All three principal Context types represent valid interworking scenarios.

5.2.2 Circuit-to-Packet Interworking

The circuit-to-packet (C2P) interworking scenario (e.g., voice over Internet Protocol) is the most common one for fixed NGNs. This C2P session type is outlined in Figure 4.

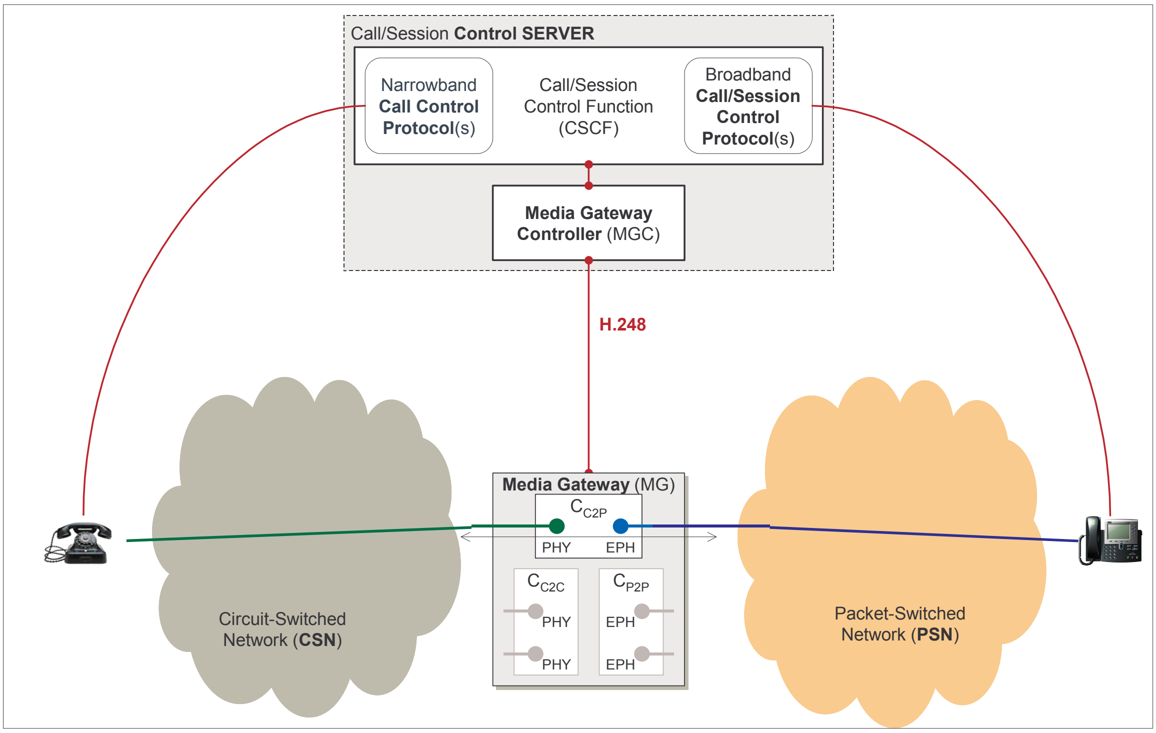

Figure 4: Session Type (1) – Circuit-to-Packet Interworking (C2P)

NOTE – The specific H.248 physical Termination type, e.g., TDM for synchronous time division multiplexed interfaces, or ALN for analog lines, is out of scope.

5.2.3 Packet-to-Packet Interworking

Figure 5 shows the session variant with two ephemeral H.248 Terminations. This interworking case is abbreviated as packet-to-packet (Pa2Pa) [8].

[8] Abbreviation 'P2P' might be confused with Peer-to-Peer (→ shall be thus abbreviated with Pe2Pe).

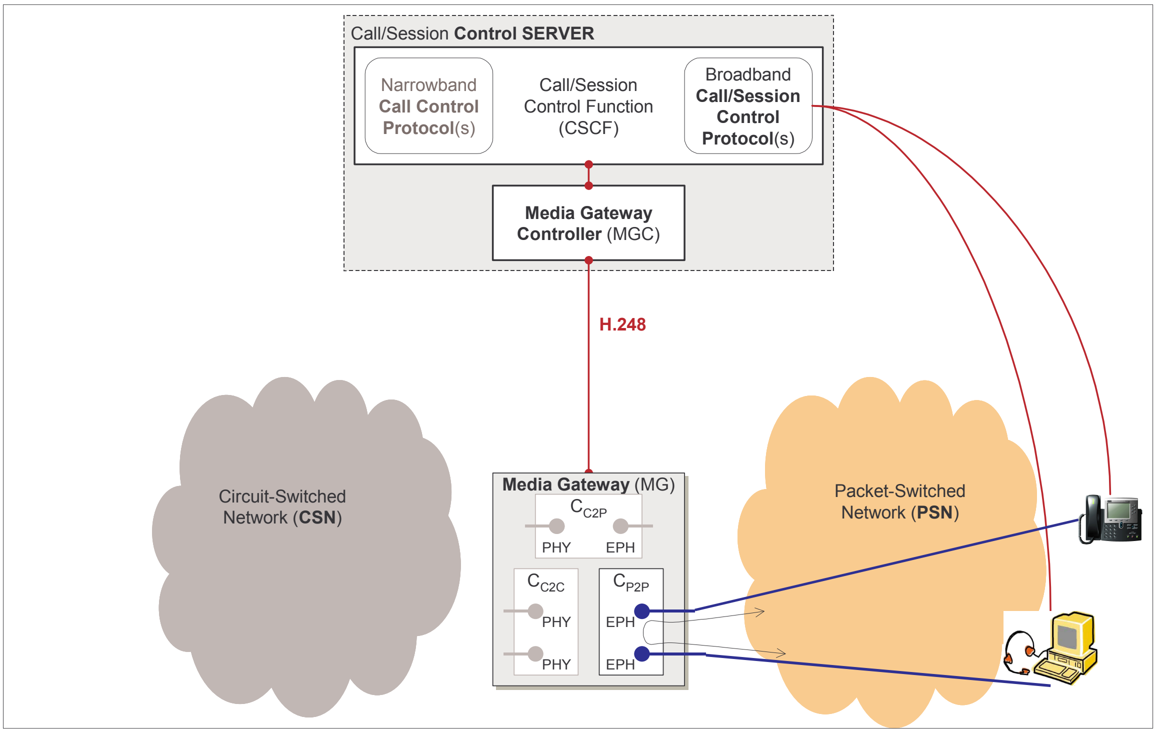

Figure 5: Session Type (2) – Packet-to-Packet Interworking (Pa2Pa)

5.2.4 Circuit-to-Circuit Interworking

The third session variant is circuit-to-circuit interworking (C2C). C2C type sessions are typically needed in order to implement an *internal traffic* type of interworking[9].

---

9 **Internal traffic** is "Traffic originating and terminating within the network considered" (ITU-T E.600). Internal traffic typically exists at local and transit exchanges. Any "CSN exchange" emulation/simulation scenario using H.248 MGs resulting in C2C type Contexts. Internal Traffic is emulated/simulated by C2C sessions (e.g., TDM-to-TDM, ALN-to-TDM, ALN-to-ALN) in NGNs.
Internal traffic corresponds to **Intrasystem Calls** (see GR-517-CORE [19], Figure 6-1).

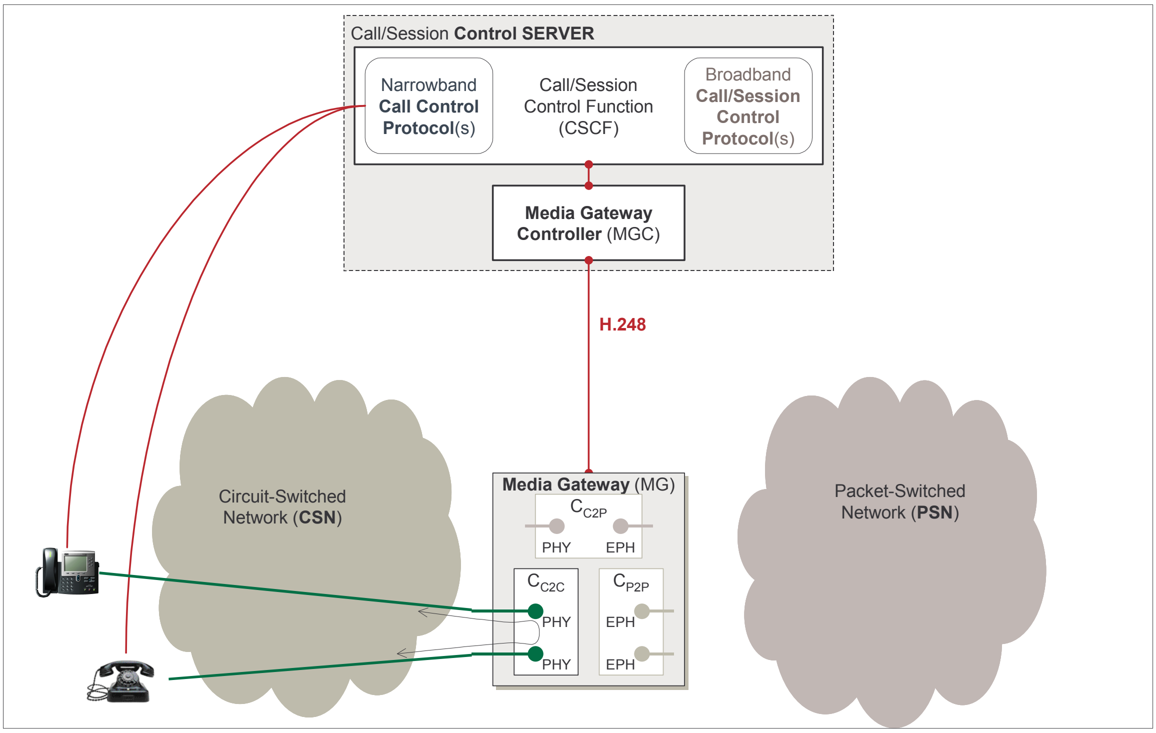

Figure 6: Session Type (3) – Circuit-to-Circuit Interworking (C2C)

5.3 Basic H.248 Context

This document proposes that the performance framework for control load metrics should be built on H.248 Contexts comprising of two H.248 Terminations. Such a Context shall be denoted as a **Basic H.248 Context**, similar to the *basic call* definitions for legacy General Switched Telephone Networks (GSTN), or Intelligent Networks (IN).

NOTE – Exemplary ITU-T *basic call* definitions:
Q.1290: A call between two users that consists of communication only, and does not include additional features.
Q.1300: A call involving exactly two communication entities.

First-order performance evaluations for basic H.248 Contexts must not take into account:

- Session type,
- H.248 Termination type,
- specific physical respectively ephemeral transport technologies,
- others.

More explanations about Basic H.248 Context are in subsection 6.4.

6 Processing Performance

Consider the vertical hierarchy of control interfaces in Figure 2, where there are multiple chained instances with different control processing performance requirements. A simplified architecture is proposed in the following.

NOTE – A more detailed view is for instance outlined in ITU-T TRQ.2141.1 Figure 2, showing an object reference model for BICC CS2 Call Bearer Control.

6.1 Idealized Model

The *monolithic control* of existing TDM switching systems was decomposed by the transition towards NGN architecture approaches. The major considered control entities are:

- the **Session Control Processor** (briefly Session processor), located in the control path of network element "session control server", and
- the **Context Control Processor** (briefly Context processor), located in the control path of network element "gateway".

Figure 7 shows that simplified *two-level control* hierarchy, as an evolution of monolithic controls. This model may be more detailed in future, e.g., by differentiating the CSCF and MGCF control parts within the session control server.

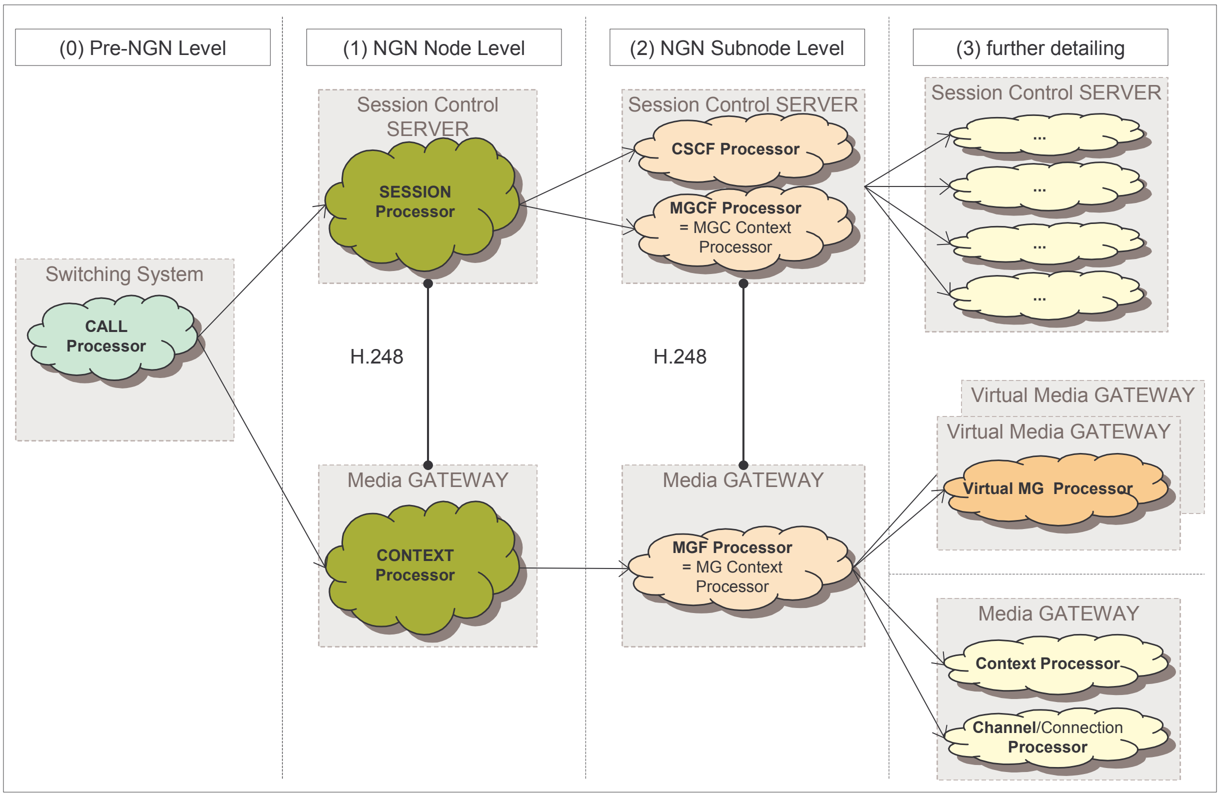

Figure 7: Control Processing Model – Potential Levels of Detailing

The scope of this recommendation is thus the (1) *NGN Node Level* of detailing as indicated in Figure 7. Other potential levels are for further study.

NOTE 1 – The term ‘processor’ shall denote the logical entity responsible for all control processing work. The technical realization may be very different, from a single CPU to multi-processor systems, in any form of cluster organization (e.g., distributed, hierarchical, load and/or functional sharing modes, etc.).

NOTE 2 – Potential concepts for refined detailing of MG controls are indicated in Figure 7. Technical motivation behind might be (a) high-capacity MGs, (b) Virtual MG support, and/or (c) MG-embedded Bearer Control Units [e.g., the so-called BIWN (Bearer Interworking Node) case in ITU-T TRQ.2141.0, see [2] Figure C.2].

Out of scope are so-called “*combined gateways*” because of (a) the existing monolithic style of the control processor, and (b) the non-existence of an H.248 interface. Combined gateways are for instance: H.323 gateways, BICC CS1 interworking nodes, 3GPP Release 3 MSCs, or SIP gateways[10] with integrated user and control plane endpoints.

6.2 Session Processing Performance

For further study. The initial scope shall be the gateway node.

6.3 Context Processing Performance

The major *performance parameter* is the *effective throughput* figure of merit (sometimes called *goodput*)[11]. Scope is the Media Gateway embedded *Context Processor*. Average service time $h_{Context,Basic}$ for processing *elementary H.248 Contexts* shall be abbreviated by:

$$h_{Context,Basic} \quad [\mathrm{s}]$$

Equation (1): Average **service time** per **basic H.248 Context** $h_{Context,Basic}$

NOTE – A high-level definition for *Basic H.248 Contexts* was introduced in subsection 5.3. Further discussion is provided in subsection 6.4.

Ideal Context processor **capacity** (see subsection 7.1 for explanation):

$$\mu_{Context,Basic} = \frac{1}{h_{Context,Basic}} \quad [\mathrm{s}^{-1}]$$

Equation (2): Context Processor – Maximum **service rate** $\mu_{Context,Basic}$

Ideal **throughput** under ideal conditions:

---

[10] Example: a SIP gateway housing RTP endpoints together with SIP user agent functionality, as well as for instance CSN circuits together with CSN call control.

[11] The complementary figure, the *ineffective throughput* is often denoted as *badput*. This noneffective throughtput is generating *blind load* in the control processor.

$$\phi_{Context,Basic} = \mu_{Context,Basic} \quad [s^{-1}]$$

Equation (3): Context Processor – Effective Context throughput $\phi_{Context,Basic}$ under ideal conditions

Equation (3) means that the stationary throughput is equal to the service rate of the control processor.

6.3.1 Completion Rate C_OCPS

Effective throughput for a real context processor under ideal conditions, i.e., every context attempt may be successfully processed:

$$\phi_{CoCPS} = \phi_{Context,Basic} \quad [s^{-1}]$$

Equation (4): Context Processor – Context Completions Per Second ϕ_{CoCPS}

NOTE – 'Ideal' means that every H.248 Context may be successfully served. There aren't any unsuccessful sessions, error situations, rejected Context requests, inadequately handled Contexts[12], or other cases.

6.3.2 Completion Rate BHC_OC

The context completion rate in time unit 'hour^{-1}':

$$\phi_{BHCoC} = \phi_{CoCPS} \cdot 3600 \quad [h^{-1}]$$

Equation (5): Context Processor – Busy Hour Context Completions ϕ_{BHCoC}

6.4 H.248 Performance Classes

Any meaningful NGN service requires, from H.248 point of view, at least a single H.248 Context. A two-party communication service demands a Context with two H.248 Terminations at a minimum. Such a generic Context shall be denoted as a 'Basic Context' (see also subsection 5.3). The necessary control processing performance during the whole lifetime of a Basic H.248 Context shall be associated with a performance class (= Class 1 in Figure 8).

---

12 "Inadequately handled H.248 Context attempts" may be defined according ITU-T Q.543: […] are attempts which are blocked (as defined in the E.600-Series Recommendations) or are excessively delayed within the MG (or MGC). "Excessive delays" are those that are greater than three times the "0.95 probability of not exceeding" values recommended in ….

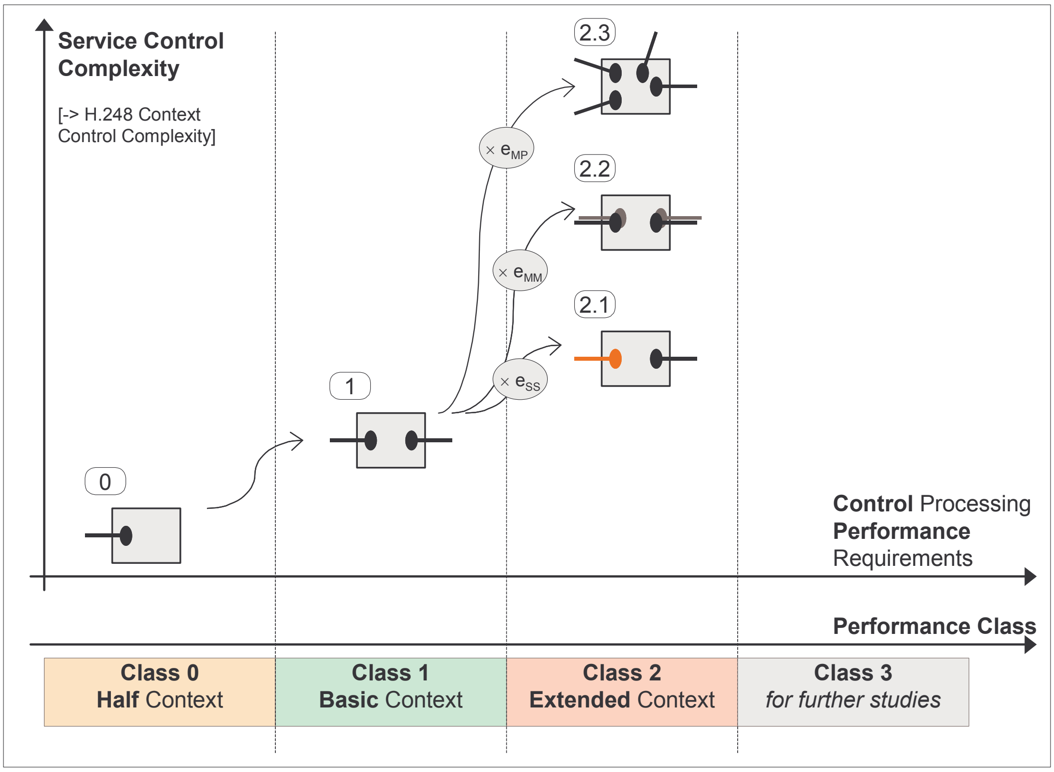

Figure 8: Performance Classes – Qualitative Categorization

The principle of differentiating a *basic service* from *extended services*, as, for example, supplementary services, is well known in telecommunication networks. This rule is also applied in performance engineering as a first classification principle, for separating basic load requirements and basic performance requirements, and additional demands associated with extended services.

NOTE – An "extended service" may be for instance (ITU-T Q.1741.1): a service which modifies or supplements a basic (telecommunication) service. Consequently, it cannot be offered to a user as a standalone service. It must be offered together with or in association with a basic (telecommunication) service. The same supplementary service may be common to a number of basic (telecommunication) services.

The same principle may be applied in defining separated categories for *Basic H.248 Contexts* and *Extended Contexts*. Figure 8 is illustrating such an abstraction concept by various performance classes. From a performance engineering point of view the Extended Context types will be linked with Basic Context by so called **extension factors** $e_{(+)}$. Exemplary Extended Context types will be introduced in the following clause, quantitative dependencies are discussed in subsection 10.1.

NOTE – Performance considerations related to operations on H.248 root terminations (e.g., specific audits) are for further studies.

6.4.1 Reduced Performance Necessity

There are processing requirements below the Basic Context level. This is indicated by the "Half Context" case in **Figure 8** (Class 0). A control load quantum below the basic level may make sense to cover, for instance, in the following cases:

- abandoned session during establishment phase,
- test signal sequences (e.g., some selected H.248.17 scenarios),

- channel associated signaling (with later Context change),
- digit collection (with later Context change),
- delivery of PSTN supplementary services in on-hook state, or
- others.

NOTE – Whether the solely H.248 Termination of the "half context" belongs to the H.248 Null Context or not, shall not be distinguished.

6.4.2 Potential Extension Areas

Table 1 provides three initial categories for potential extension areas. The resulting Extended Contexts have extended performance requirements.

Table 1/H.248.BHCoA – Exemplary Extended Contexts

Class 'Extended'	Extension Factor $e_{(+)}$	Class Labelling
2.1	e_{SS}	Superset Services (SS) => extension from basic services towards additional services *per H.248 Termination* => e.g., like inband signalling, channel associated signalling, Subscriber Line Protocol based PSTN supplementary services, overload protection, etc.
2.2	e_{MM}	Multimedia (MM) => extension from monomedia towards multimedia sessions => e.g., single media stream per H.248 Termination, i.e., multiple Terminations per session party; or multiplexed cases: multiplexed media streams, cascaded multiplexing Terminations, etc.
2.3	e_{MP}	Multiparty (MP) => extension from 2-Party (2PY) towards 3-Party (3PY), and general Multiparty session configurations
2.4		for further studies

NOTE – This initial categorization scheme of Table 1 might be too coarse for specific performance engineering cases. But a more detailed classification, e.g., by separating e_{SS} in for instance $e_{SS,CAS}$, $e_{SS,CLIP}$, or $e_{SS,Test}$ within class 2.1, is for further studies.

The "Extended Class" is one case where there are increased performance requirements *per session*. It shall be noted that another case may be the Session-to-Context ratio (see chapter 9 'Session-to-Context Relation').

6.4.3 Classification Tools

6.4.3.1 Signalling Scenario

Signalling scenarios (also known as *Message Sequence Charts*) are often used as first-order qualifiers for the indication of underlying service control complexity. Additionally second-order qualifiers might be for example the respective *signalling message types*. Particular signalling message *information elements* may act as 3rd-order qualifiers.

Similar proceeding may be applied for H.248 signalling as well, by considering for instance, the mean number of H.248 Commands per session, Context manipulation functions, Termination modifications, etc. An H.248 Context control complexity indicator might be then "derived" from a signalling complexity.

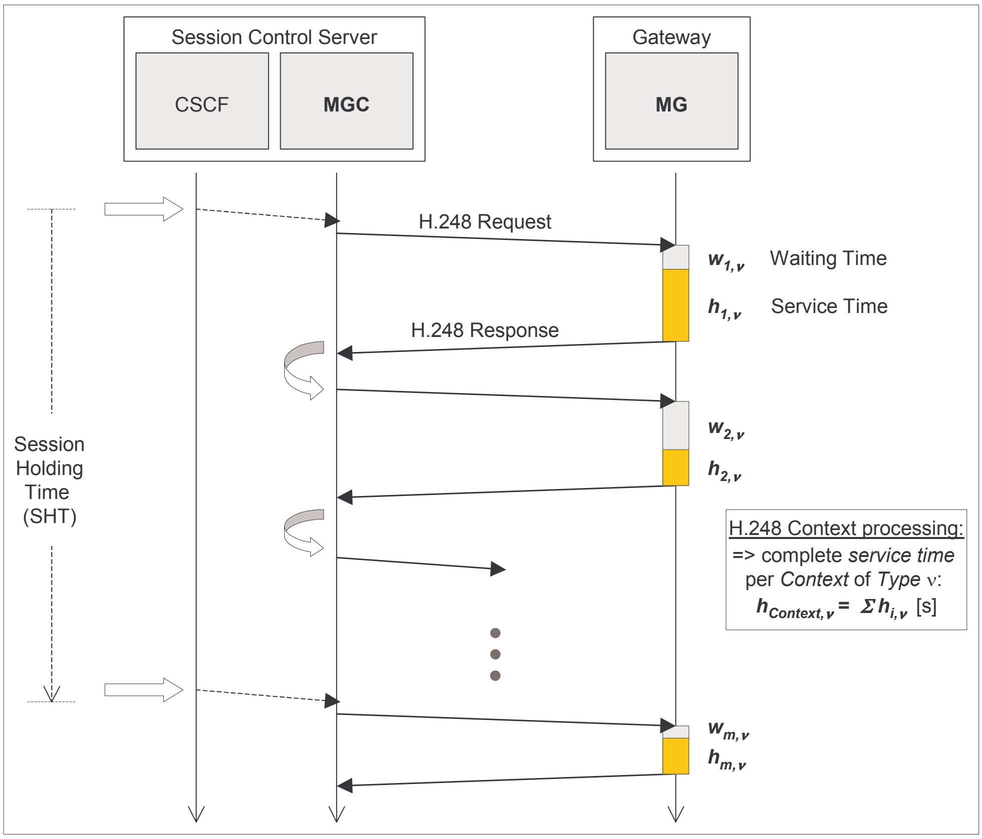

Figure 9: Generic H.248 Signalling Scenario

Figure 9 illustrates a generic H.248 signalling scenario. The usage of H.248 signalling scenarios for the derivation of H.248 performance metrics is for further study.

6.4.3.2 Session/Context State Machine Models

Refined BHC_aA models were often based on the consideration of advanced finite state machines for call modeling. The same principle may be applied for H.248 Context modeling. A Context state machine model approach is for further study

NOTE – A simple state machine for modelling an H.248 Context lifetime might be: There are two Context states, either 'idle' or 'active'. The active state is reached by Context creation, and left, for example by final Termination SUBtract. There may be two further types of state transitions defined for characterizing active-to-active state transitions, (a) MODification events (triggered by MGC), and (b) NOTification events (triggered by MG local events). Corresponding traffic parameters, e.g., modification rate, notification rate, may be defined for profiling service and thus, qualifying H.248 performance classes.

6.4.3.3 Code Count Method

The Code Count method is a traditional instrument for first-order estimations of performance requirements. This reverse engineering approach is based on the analysis of control software. In the meantime modern source code analyser tools[13] allow the automatic generation of a variety of software metrics. Some of these metrics might be used for performance classification., e.g., the specific volume metric “number of lines containing source code”.

NOTE – Of course, an absolute classification isn’t possible due to the implementation specific character of software (e.g., programming language, architecture). But a relative classification with regard to quantitative categorization of performance classes, as well as the separation of subclasses within a dedicated class, is straightforwardly possible.

7 Capacity

Performance is always limited in every technical system by its inherent available capacity. The control processor *capacity* figure is consequently an important link between *performance* (previous chapter 6), and *load* (subsequent chapter 8). These principles still apply in the case of H.248 systems. The main purpose of this chapter is thus to recall the two major capacity terms.

7.1 Theoretical Capacity

The theoretical control processing capacity is the maximum service rate, i.e., the maximum session completion rate, respectively the maximum H.248 Context completion rate. See for instance $\mu_{Context,Basic}$ (Equation (2)) for Basic H.248 Contexts processed by the Context processor.

7.2 Engineered Capacity

The engineered capacity is always below the theoretical processor capacity. If a session/Context based definition is required in future, then an Q.543-based adaptation is recommended.

NOTE – ITU-T Q.543 “Engineered Capacity”: The mean offered load at which the exchange just meets all the **grade of service requirements** used by the Administration to engineer the exchange.

8 Reference Control Load

The purpose of this chapter is to focus on the *load parameters* related to Context processing. Further *Performance* objectives (beyond chapter 6 basis) are scoped in the subsequent chapters. Figure 10 shows the principle dependencies between the several load factors and corresponding performance types. The control processing model is based on the “*NGN subnode level*” according Figure 7.

---

[13] For instance: www.scitools.com, …

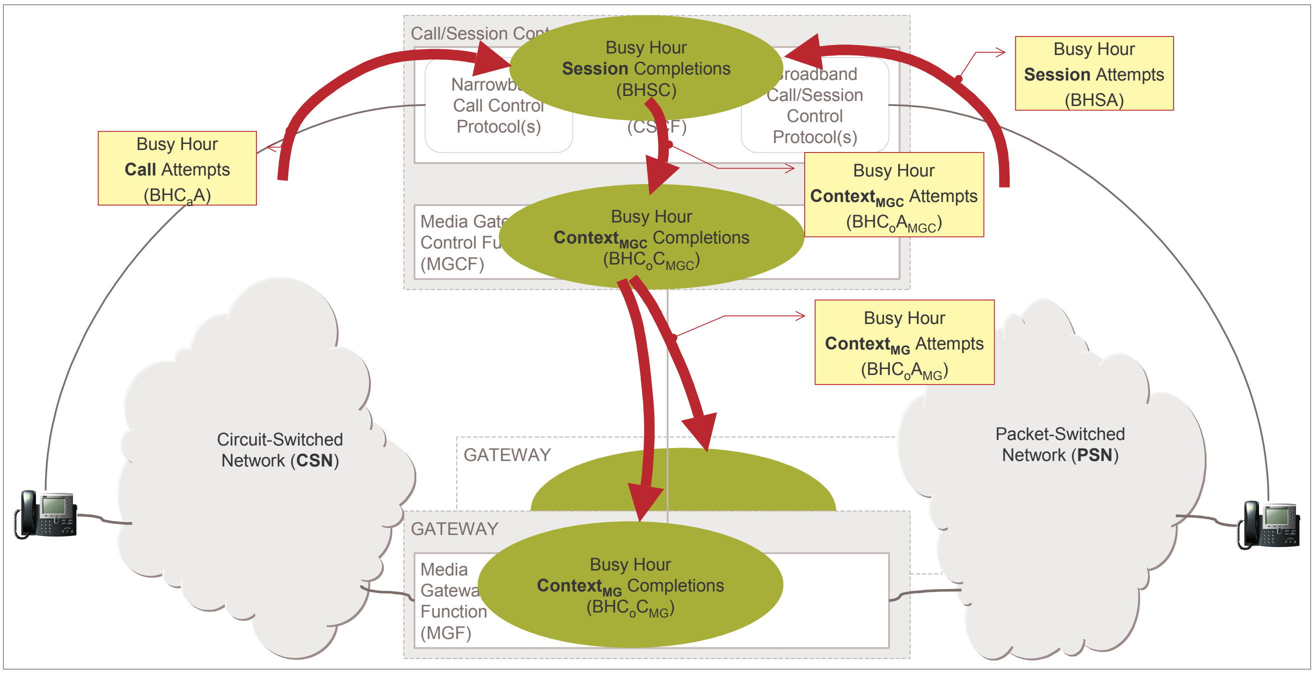

Figure 10: Control Processing Model – Load/Performance Chaining

NOTE 1 – Sometimes load is put on a level with performance. This isn't definitely the general case for technical systems like H.248 nodes. Of course, under certain conditions, performance equals the load. For instance, in stationary *low load* situations the BHC_oC may be approximated with the BHC_oA figure.

NOTE 2 – A simple model describing the principal load-performance behaviour may be the *Lost Context Model*, see subsection 12.1, and 12.2.3.

The two-processor model continues to be assumed in the sections which follow (as illustrated in Figure 7 *NGN Node Level*).

8.1 Session Processor Load Parameters

The arrival rate of session attempts may be defined in time unit levels of a 'second' and an 'hour'.

8.1.1 Arrival Rate SAPS

The rate of session attempts per second λ_{SAPS}:

$$\lambda_{SAPS} \quad [s^{-1}]$$

Equation (6): Session Processor – **Session Attempts Per Second** λ_{SAPS}

8.1.2 Arrival Rate BHSA

The session attempt rate in time unit 'hour^{-1}':

$$\lambda_{BHSA} = \lambda_{SAPS} \cdot 3600 \quad [h^{-1}]$$

Equation (7): Session Processor – **Busy Hour Session Attempts** λ_{BHSA}

8.2 Context Processor Load Parameters

The arrival rate of H.248 Context attempts may be defined in time unit levels of a 'second' and an 'hour'.

8.2.1 Arrival Rate C_OAPS

The rate of Context attempts per second λ_{CoAPS}:

$$\lambda_{CoAPS} \quad [s^{-1}]$$

Equation (8): Context Processor – **Context Attempts Per Second** λ_{CoAPS}

8.2.2 Arrival Rate BHC_OA

The Context attempt rate in time unit 'hour^{-1}':

$$\lambda_{BHCoA} = \lambda_{CoAPS} \cdot 3600 \quad [h^{-1}]$$

Equation (9): Context Processor – **Busy Hour Context Attempts** λ_{BHCoA}

8.2.3 Basic Context Control Load

The **offered load** $A_{ContextProcessor}$ (briefly A_{CP}) to the MG-embedded Context processor, generated by incoming attempts for basic H.248 Contexts, is:

$$A_{ContextProcessor} = \lambda_{CoAPS} \cdot h_{Context,Basic} \quad [Erl]$$

Equation (10): Offered load $A_{ContextProcessor}$ for basic H.248 Contexts

NOTE – The *offered load* A_{CP}, defined by Equation (10), corresponds to ITU-T E.500 [10] parameter *traffic intensity* A [Erl]. E.500 section 5.2 describes the "traffic intensity concept and stationarity". E.500 may be reused by replacing "job" with "H.248 Context", and "resource holding time" with "Context holding type (C_OHT)".

8.2.3.1 Normal Load

The definition of a "*Normal Basic Context Control Load*" parameter is for further study. An E.500 *Normal Load Traffic Intensity* based definition will be recommended (if required in future).

8.2.3.2 High Load

The definition of a "*High Basic Context Control Load*" parameter is for further study. An E.500 *High Load Traffic Intensity* based definition will be recommended (if required in future).

8.3 Reference Load Definitions

Reference load definitions, e.g., for performance class “Basic H.248 Context”, are for further study.

NOTE – Telcordia GR-517-CORE [19], or ITU-T Q.543 [5] are providing reference load definitions for digital exchanges. The reference loads are defined by using load parameter types “traffic intensity”, “arrival rate”, and/or “holding time”.

9 Session-to-Context Relation

9.1 Background

The H.248 decomposed gateway principle leads to the fact, that the correlation between a user plane connection (here H.248 Context), and a respective control plane association (here Session) disappears from a Media Gateway perspective. The knowledge about the session identifier and corresponding Context identifier(s) is located in the session control server (housing the MGC instance). The MG is missing that kind of information.

NOTE – The same situation applies for MG-embedded signalling gateways (SG), like IETF SIGTRAN SGs. For instance, in case of a SIGTRAN IUA SG, the MG doesn’t have the knowledge whether control plane connections (here Q.931/Q.921) are associated with user plane connections (here H.248 Context).

This means that the MG may not correlate session control load with Context control load.

NOTE – For instance in Figure 11, the MG doesn’t know that the (1st) H.248 Contexts $C_{i,j}$ belonging to Session S_i, and also that (2nd) the two consecutive H.248 Contexts $C_{i,1}$ and $C_{i,2}$ belong to the same Session S_i.

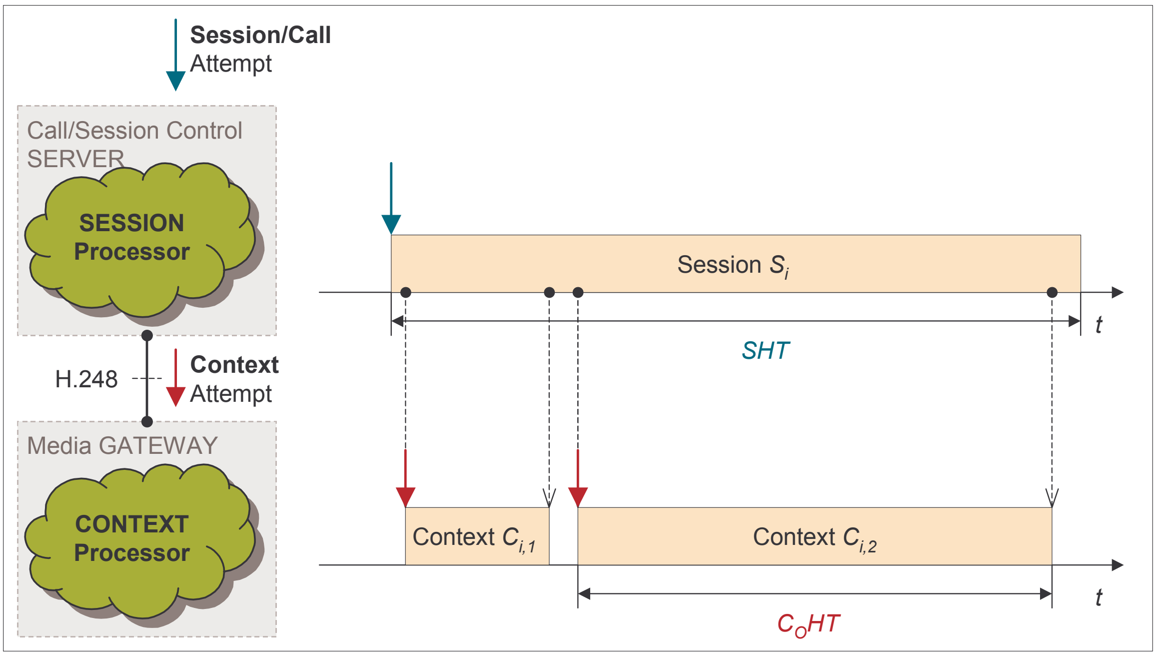

Figure 11: General Session-to-Context Relation

NOTE – The sketched holding times in **Figure 11** referring to the *mean Session Holding Time* (SHT) respectively *mean H.248 Context Holding Time* (C_OHT).

9.2 1:1-Relation

There is a 1:1 relation between a session and a corresponding H.248 Context for the majority of services. This means that a single H.248 Context C_i must be processed in a Media Gateway behind a single session S_i in the control server.

NOTE – It has to be noted that multiple MGs may be involved in the same session, and all these MGs are controlled by the same session control server. But this doesn't change the fact of the 1:1-relation from MG point of view.

9.2.1 Control Load – Session rsp. Context Arrival Rates

The resulting arrival rates on the session processor and Context processor level are identical.

$$\lambda_{CoAPS} = \lambda_{SAPS} \quad [\mathrm{s}^{-1}]$$

$$\lambda_{BHCoA} = \lambda_{BHSA} \quad [\mathrm{h}^{-1}]$$

Equation (11): Arrival rates for 1:1 relation

NOTE – Of course, identical arrival rates may not lead to identical load factors on the session processor and the Context processor . It's rather the usual case that $A_{ContextProcessor}$ differs from $A_{SessionProcessor}$ due to the *server* approach, i.e., typically is $A_{SessionProcessor} < A_{ContextProcessor}$.

9.3 1:N-Proportion

There are many services with a 1:N ratio of a single session and associated number of Contexts in a MG.

NOTE – Example: session triggered bearer connection tests before the end-to-end conversation phase. For instance, SS7 Continuity Checks for the call/session associated circuit. Such a test might be done via a first H.248 Contexts $C_{i,1}$, the consecutively following conversation is handled by second Context $C_{i,2}$. It shall be noted again that the MG may not correlate both Contexts $C_{i,1}$ and $C_{i,2}$. Other examples are given in subsection 6.4.1.

9.3.1 Rate Multiplication Factor N

The resulting Context attempt arrival rate is N times higher than the session arrival rate.

$$\lambda_{CoAPS} = N \cdot \lambda_{SAPS} \quad [\mathrm{s}^{-1}]$$

$$\lambda_{BHCoA} = N \cdot \lambda_{BHSA} \quad [\mathrm{h}^{-1}]$$

Equation (12): Arrival rates for 1:N relation

There is typically a mix of 1:1 and 1:N types of sessions in a real network, i.e., the average rate multiplication factor is between 1 and N. The crucial point is that the Context arrival rate is greater equal than the session arrival rate (e.g., $BHC_OA \geq BHSA$), qualitative coherences in Figure 12.

NOTE 1 – The Context arrival rate BHC_OA is often used as a load indicator (beside others) for the Context processor local overload protection mechanisms. If 1:N types exist in an H.248 network then the MG should be cautious in using the BHC_OA parameter in control loops for load regulation, or overload control, due to his lack of knowledge of the real multiplication factor.

NOTE 2 – The rate multiplication factor N is of type Integer, the average rate multiplication factor $\overline{N}$ is typically a non-Integer type.

NOTE 3 – The resulting average rate multiplication factor $\overline{N}$ is leading to a **virtual session attempt rate** (or **virtual call attempt rate**) of $\lambda'_{SAPS,MG} = \overline{N} \cdot \lambda_{SAPS}$ from H.248 Media Gateway perspective.

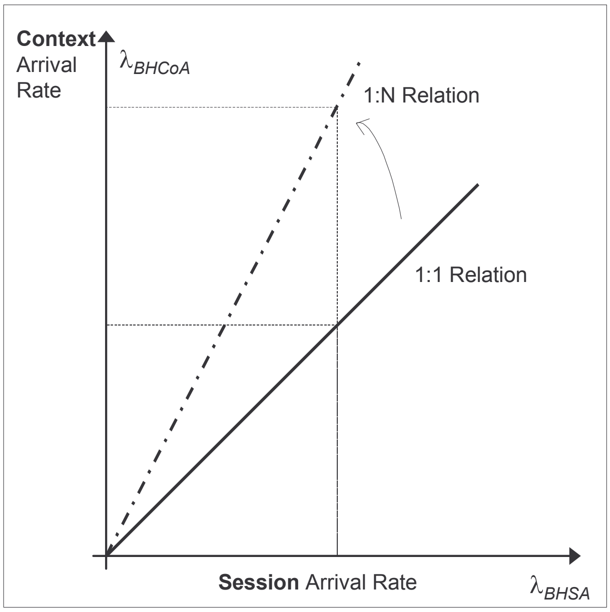

Figure 12: Session-to-Context Proportion – Multiplication Factor N between Arrival Rates

9.3.2 Effective Multiplication Factor κ

The individual Contexts $C_{i,j}$ may be of different complexity type (see 6.4 H.248 Performance Classes), resulting in different individual mean service times $h_{Context,Ci,j}$ from Context processor point of view. An *effective multiplication factor* κ characterizes the increased Context processing performance requirements behind a single session (in 1:N-relation scenarios):

$$\kappa = \frac{\sum_{j=1}^{N} h_{Context,Ci,j}}{h_{Context,Basic}}$$

Equation (13): **Effective multiplication factor κ** based on basic H.248 Context service time $h_{Context,Basic}$

NOTE - The effective multiplication factor κ is typically applied as a first-order performance estimation.

Figure 13 illustrates how the increased Context processor load A_{CP} relates to the effective multiplication factor κ.

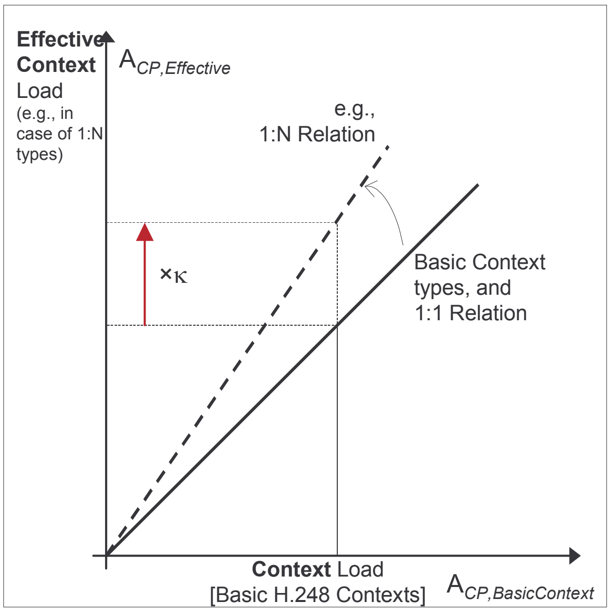

Figure 13: Context Processor Load A_{CP} – Effective Multiplication Factor κ

NOTE – For example , the effective Context processor load $A_{CP,Effective}$, applicable on a 1:N Session-to-Context types, may be related to basic Context processing, and estimated with

$$A_{CP,Effective} = \kappa \cdot A_{CP,BasicContext}.$$

10 Extensions for the Basic Control Load Quantum

The purpose of this chapter is to introduce the additional parameters, which are needed for handling the "Extended Context" performance class.

10.1 Extension Factors

The additionally needed average service time $h_{Context,(+)}$ extends the required context processor service time to:

$$h_{Context,Ext.} = h_{Context,Basic} + h_{Context,(+)} \quad \text{[s]}$$

Equation (14): Average **service time** per **extended H.248 Context** $h_{Context,ext.}$

NOTE – The '(+)' is a placeholder for one of the potential extension reasons mentioned in previous subsection 6.4.2.

A generic extension factor $e_{(+)}$ related to the basic Context service time is introduced:

$$e_{(+)} = \frac{h_{Context,Ext.}}{h_{Context,Basic}} = 1 + \frac{h_{Context,(+)}}{h_{Context,Basic}}$$

Equation (15): Generic **extension factor** $e_{(+)}$

Example for a specific extension factor, e.g., an average figure e_{SS} for class 2 superset services (e.g., PSTN supplementary services):

$$e_{SS} = 1 + \frac{h_{Context,SS}}{h_{Context,Basic}}$$

Equation (16): Exemplary specific **extension factor** e_{SS}

10.2 Throughput Reduction Factors

The increased service time requirements for extended H.248 Contexts lead to a reduction of the Context completion rate. The generic reduction factor $r_{(+)}$ is:

$$r_{(+)} = \frac{1}{e_{(+)}} = \frac{h_{Context,Basic}}{h_{Context,Basic} + h_{Context,(+)}}$$

Equation (17): Generic **reduction factor** $r_{(+)}$

10.3 Reduced Effective Throughput in case of Extended H.248 Context Processing

10.3.1 Completion Rate $BHC_{O,Ext}C$

The context completion rate is reduced in comparison to the basic Context completion rate:

$$\phi_{BHCo,extC} = r_{(+)} \cdot \phi_{BHCoC} \quad [h^{-1}]$$

Equation (18): Context Processor – Reduced **Busy Hour Context Completions** $\phi_{BHCo,extC}$ for extended Context processing

NOTE – It should be pointed out that rather than the Context Processor performance being reduced, it stays the same, e.g., in terms of program instructions per second performance unit.

11 Appendix I

Fundamental Relations

11.1 Relation between Effective Multiplication Factor κ and Extension Factor *e*

The two linear factors *Effective Multiplication Factor* κ and *Extension Factor e* are simply linked (according to Equation (13) and Equation (15)):

$$\kappa = \sum_{j=1}^{N} e_{(+),j}$$

Equation (19): **Effective multiplication factor κ** as sum of the individual **extension factors** $e_{(+),j}$

Equation (19) allows a quick first-order load/performance estimation in the case of the known individual class specific extension factors.

NOTE – The inclusion of class mixes, subclasses, weighting factors, etc. is for further study.

12 Appendix II

Basic Traffic Models for H.248 Systems

Some basic traffic models for H.248 systems are presented. Particularly for following performance evaluation areas:

- **Lost Context Model** (→ clause 12.1)
- **Overload Control Model** (→ clause 12.2)
- **Combined Control/User Plane Mode** (→ clause 12.3)
- **Control Performance versus Context Holding Time** (→ clause 12.4)

12.1 Lost Context Model

ITU-T **E.501** [11] Annex B "*Equivalent Traffic Offered*" describes the basic **load-performance dependency** in the case of a loss model. The model represents a conservation law. This E.501 "lost call model" can be mapped on to the MG level Context Processor:

In the *lost Context model* the equivalent traffic offered corresponds to the traffic which produces the observed carried traffic in accordance with the following relationship:

$$Y_{CP} = A_{CP} \cdot (1 - B_{CP}) \text{ [Erl]}$$

Equation (20): Lost Context Model for H.248 MG Context Processor

where:

Y: is the carried traffic (= *completed Contexts*);

A: is the equivalent traffic offered (*see Equation (10)*);

B: is the *Context* congestion through the part of the network (= *MG*) considered.

NOTE 1 – This is a purely mathematical concept. Physically, it is only possible to detect "offered traffic" whose effect on occupancies tells whether these attempts give rise to very brief seizures or to calls.

NOTE 2 – The equivalent traffic offered, which is greater than the traffic carried and therefore, greater than the effective traffic, is greater than the traffic offered when the subscriber is very persistent.

NOTE 3 – *B* is evaluated on a purely mathematical basis so that it is possible to establish a direct relationship between the traffic carried and call congestion *B* and to dispense with the role of the equivalent traffic offered *A*.

12.2 Overload Control Model

There is a H.248 Context Control Processor on MGC and MG level (see Figure 7, page 10). H.248.11 is describing an overload control framework, comprising both Context Processors on MGC and MG level. Purpose of H.248.11 is a co-operation principle between MGC and associated MGs, realized by a distributed control loop. Purpose of this clause is a basic model for local overload controls. Local means that the scope of the control loop is spatially limited on the network node, or geographically limited on network node locations.

12.2.1 Theoretical Throughput Model

Figure 14 shows a single server model for an H.248 Context Processor. The server has two phases. The server is either in idle state, or in phase ‘C’ in case of successful Context processing, or in phase ‘R’ in case of rejecting Context attempts.

NOTE 1 – Target of Context rejection phase is a protocol conform feedback to the “served user” instance. This is either an call/session control server internal application on top of the MGC in case of an “MGC Context Processor”, or the MGC itself in case of an “MG level Context Processor”. Protocol conform reaction shall prevent “repeated Context attempts”.

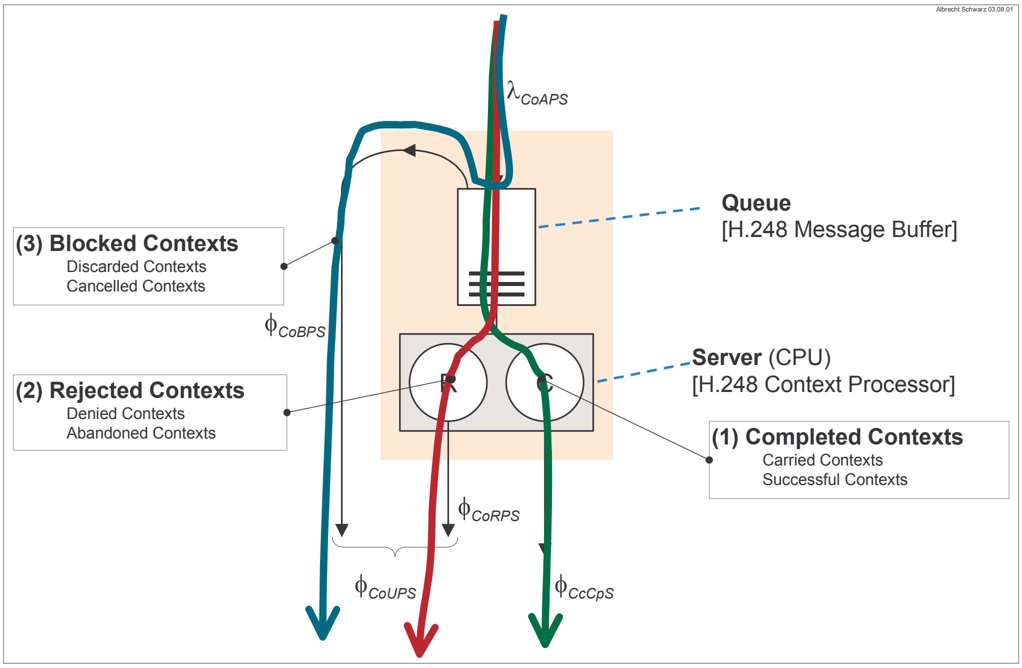

Figure 14: Traffic Model for Ideal Throughput Considerations

The H.248 message buffer has a limited size. Fully filled buffers may lead to H.248 traffic loss. The resulting traffic rate shall be denoted as blocked Contexts, in comparison to the rejection rate.

NOTE 2 – The difference between ‘blocking’ and ‘rejection’ is the fact, that blocking doesn’t need any server processing time.

12.2.2 Traffic Model for Real Systems

The queue blocking effect shall be not considered anymore. A real Context Processor is only aware of an H.248 protocol message, if the message is identified as such. Such a protocol analysis is always coupled with processing time. The resulting traffic model is illustrated in Figure 15.

Every Context attempts is either successfully handled as completed Context, or rejected.

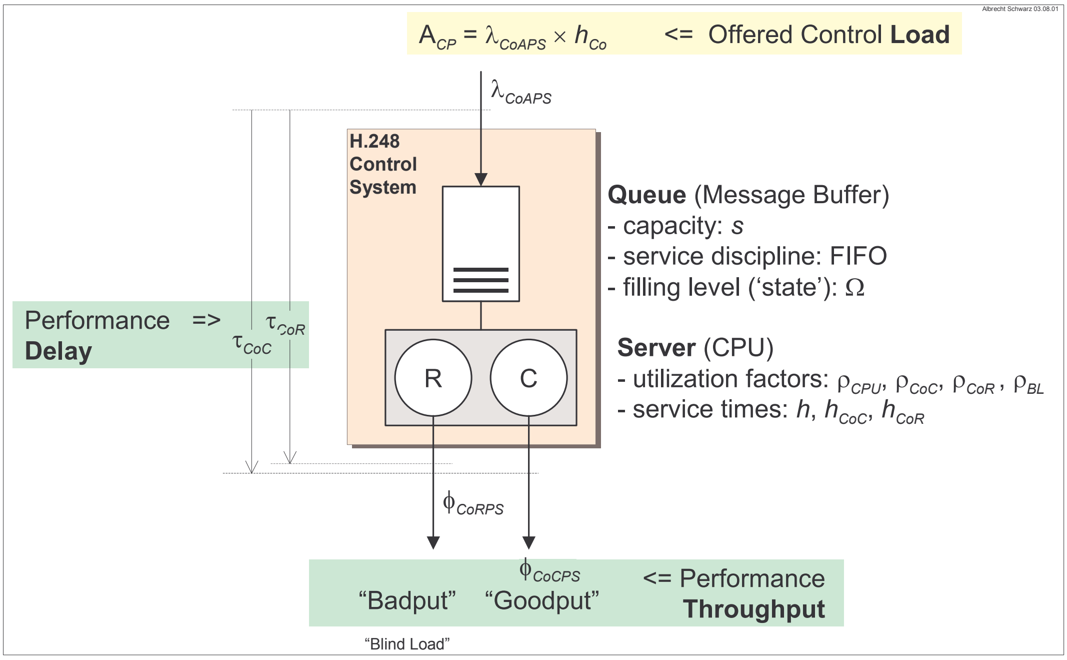

Figure 15: Traffic Model for Overload Considerations

It is obvious that the completion of an H.248 Context consumes much more processing time as any unsuccessful Context handling (see also Equation (22)). The system time τ is resulting from service time h_{Co} and waiting times.

12.2.3 Flow Analysis

The conservation law is valid under **stationary** conditions:

$$\phi_{CoCPS} = \lambda_{CoAPS} - \phi_{CoRPS} \quad [s^{-1}]$$

Equation (21): Conservation Law – Stationary Context Rates

NOTE – Equation (20) from the Lost Context Model is the dimensionless (→ Erl) pedant to the rate (→ s^{-1}) proportions in Equation (21).

12.2.4 Assumptions

12.2.4.1 Process Types

Markov property is assumed for the stochastic arrival and service processes. The traffic model belongs therefore to the class of M/M/1 types. An infinite queue is assumed for later qualitative estimations.

12.2.4.2 Service Times

Unsuccessfully processed or uncompleted H.248 Contexts demanding typically less system resources as Context completion:

$$h_{CoR} = \kappa \cdot h_{CoC}$$
$$h_{CoR} << h_{CoC}$$

Equation (22): Qualitative relation between service times h_{CoR} and h_{CoC}

NOTE – For first-order quantitative estimations may be a factor κ of 10% assumed.

12.2.5 Principle Context Processor Behaviour

The average *Context serving time* h_{Co} depends on the stationary operating point ("equilibrium"), and the corresponding *Context completion rate* ϕ_{CoC} and *rejection rate* ϕ_{CoR}.

$$h_{Co} = f(h_{CoC}, h_{CoR})$$

Equation (23): Average service time per Context $h_{Context}$ as a function of the operating point

This model and assumptions are resulting in a stationary server behaviour, which is very well-known from conventional STM switches (see Q.543 [5]). Figure 16 is illustrating the server utilization factors versus Context attempt arrival rate.

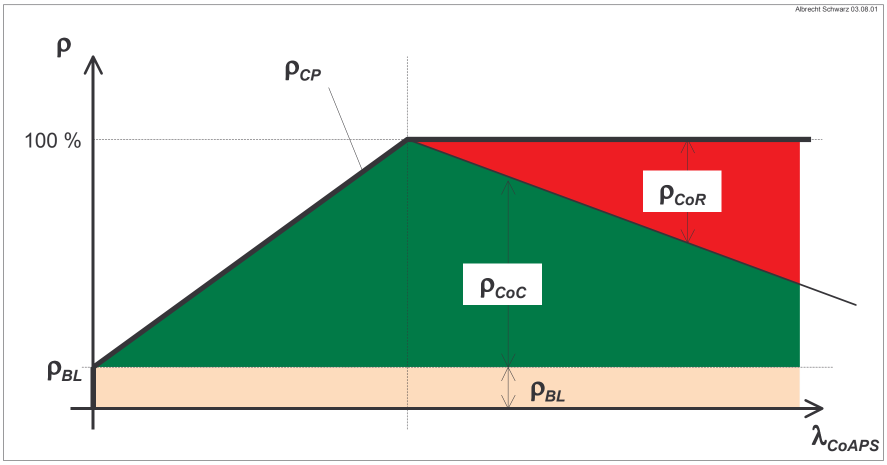

Figure 16: Idealized Context Processor Behaviour – Server Utilization Factors versus Context Arrival Rate

12.2.6 Server Operation Modes – Workload Areas for a Context Processor

The operation mode of an H.248 Context Processor is determined by the Context attempt arrival rate λ_{CoAPS}. Three principal server states by be distinguished:

$$Server_{State} = \begin{cases} \text{Underloaded} & 0 \le \lambda_{CoAPS} \le \lambda_{CoAPS,100\%} \\ \text{Overloaded} & \lambda_{CoAPS,100\%} \le \lambda_{CoAPS} \le \lambda_{CoAPS,Instable} \\ \text{Instable} & \lambda_{CoAPS,Instable} \le \lambda_{CoAPS} \end{cases}$$

Equation (24): Server State – Workload Areas dependent on Arrival Rate λ_{CoAPS}

12.2.6.1 Operation Mode "Unterload"

Editor's note: include derivation for $\lambda_{CoAPS,100\%}$ ...

$$\lambda_{CA,100\%} = \frac{1-(\rho_{BL}+\rho_{HR})}{h_{CC}}$$

Equation (25): Underloaded Server – Right-Hand Limit $\lambda_{CA,100\%}$

12.2.6.2 Operation Mode "Overload"

Editor's note: include derivation for $\lambda_{CoAPS,Instable}$ ...

$$\lambda_{CA,Instable} = \frac{1-(\rho_{BL}+\rho_{HR})}{\kappa \cdot h_{CC}} = \frac{1-(\rho_{BL}+\rho_{HR})}{h_{RC}}$$

Equation (26): Overloaded Server – Right-Hand Limit $\lambda_{CA,Instable}$

For the limit operating point $\lambda_{CA,Instable}$ is $\phi_{CC} = 0$ and hence $\phi_{RC} = \lambda_{CA} = \lambda_{CA,Instable}$.

12.2.6.3 Operation Mode "Instable"

Editor's note: include text

12.2.7 Throughput Estimation

The effective throughput versus control load function $\phi_{CoCPS} = f(\lambda_{CoAPS})$ is resulting in three straight line equations:

$$\phi_{CoCPS} = f(\lambda_{CoAPS}) = \begin{cases} \lambda_{CoAPS} & 0 \le \lambda_{CoAPS} \le \lambda_{CoAPS,100\%} & \text{Underloaded Server} \\ \dfrac{1-(\rho_{BL}+\rho_{HR})}{(1-\kappa)h_{CoC}} - \dfrac{\kappa}{1-\kappa}\lambda_{CoAPS} & \lambda_{CoAPS,100\%} \le \lambda_{CoAPS} \le \lambda_{CoAPS,\text{Instable}} & \text{Overloaded Server} \\ 0 & \lambda_{CoAPS,\text{Instable}} \le \lambda_{CoAPS} & \text{Instable Server} \end{cases}$$

Equation (27): Context Processor Operation Modes –
Straight Line Equation $\phi_{CoCPS} = f(\lambda_{CoAPS})$

NOTE – Every server shouldn't be engineered for 100% utilization. There should be still reserves (also known as headroom) under high load situations. Thus, Context processor reserves are covered by factor ρ_{HR} in Equation (27).

Figure 17 is summarizing the *goodput function* and underlying *server utilization* for the three different workload areas.

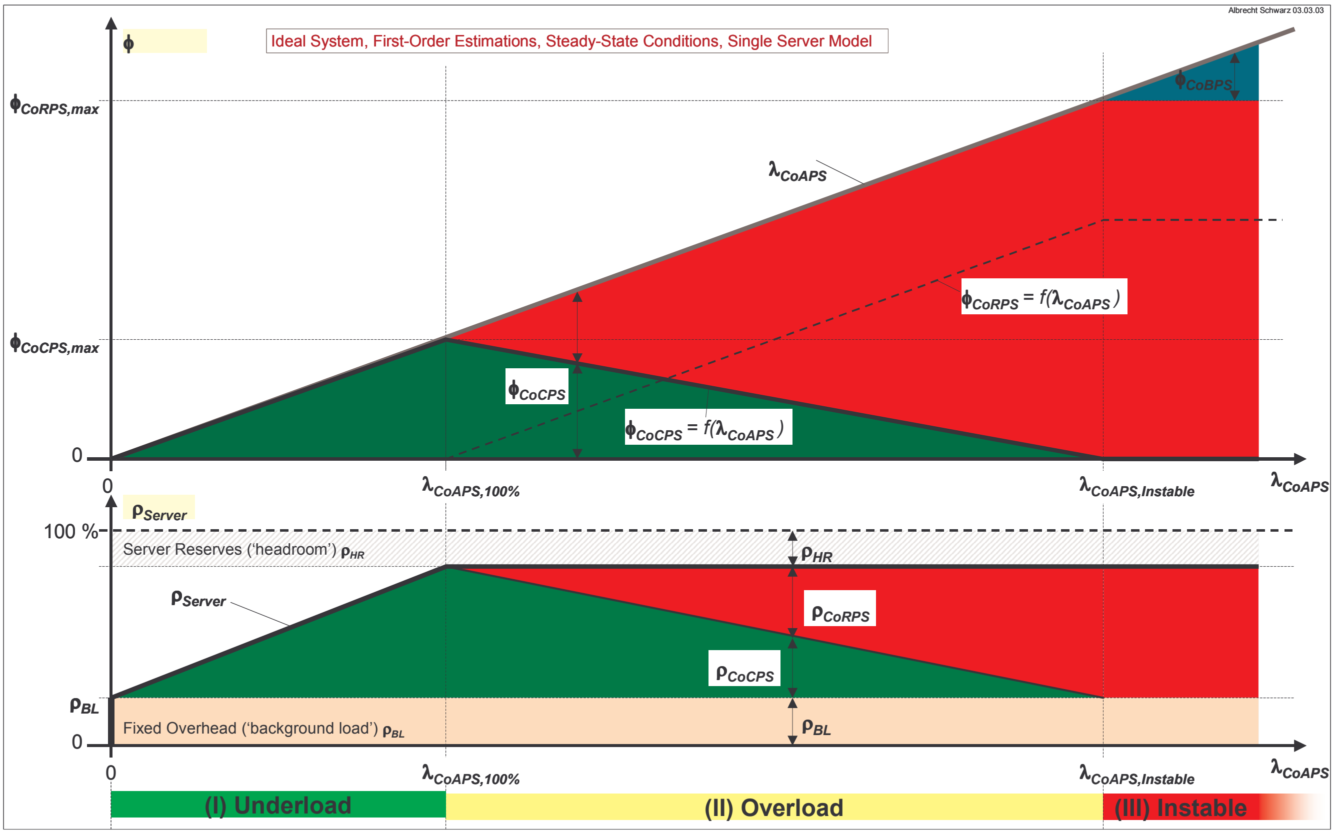

Figure 17: H.248 Context Processor **Operation Modes** – Three Principal Workload Areas

12.2.8 Conclusions

This overload control model allows to distinguish three principle operation modes of an H.248 Context Processor. Linearization, for first-order estimations, is possible within each operation state. The overall server behaviour is very nonlinear.

The maximum Context thoughput or goodput$_{max}$ $\phi_{CoCPS,max}$ is:

$$\phi_{CoCPS,\max} = \phi_{CoCPS}(\lambda_{CoAPS,100\%}) = \frac{1-(\rho_{BL}+\rho_{HR})}{h_{CoC}}$$

Equation (28): Optimal Goodput $\phi_{CoCPS,max}$

12.3 Combined Control/User Plane Model for H.248 Contexts of Type "Circuit-to-X"

A simple estimation model is presented for a specific class of H.248 Context types.

12.3.1 Background from Circuit-Switched Networks

There is a 1:1-relation between a call and a bearer connection in circuit-switched networks (CSN). An analog line (ALN), or a TDM circuit, is directly associated with the controlling call. Such a tight coupling leads in the H.248 model to the fact, that certain traffic parameters behind a physical H.248 Termination may be easily combined with control plane parameters. This relation is helpful for engineering H.248 systems in case of C2X Context types.[14]

12.3.2 Traffic Model

Figure 18 shows an example of a combined user/control plane model for an H.248 Media Gateway. The control path shall be modeled by the *single server* model presented in subsection 12.2.2. The server entity is the H.248 *Context Processor* (CP). The MG data path shall be modeled by a *K-server*. The server entity is a *Media Processor* (MP) consisting of *K Media Conversion Units* (MCU). A Media Conversion Unit is responsible for the majority of functions required for service and network interworking.

NOTE – Following terminology shall be applied. *User plane* and *control plane* are used for **system external** interfaces, for instance, DS0/E1/PDH as U-plane interface respectively H.248 as C-plane interface. **System internally** the corresponding terms *data path* and *control path* shall be applied.

[14] C2X may be either session variant C2P (see subsection 5.2.2), or session variant C2C (see subsection 5.2.4).

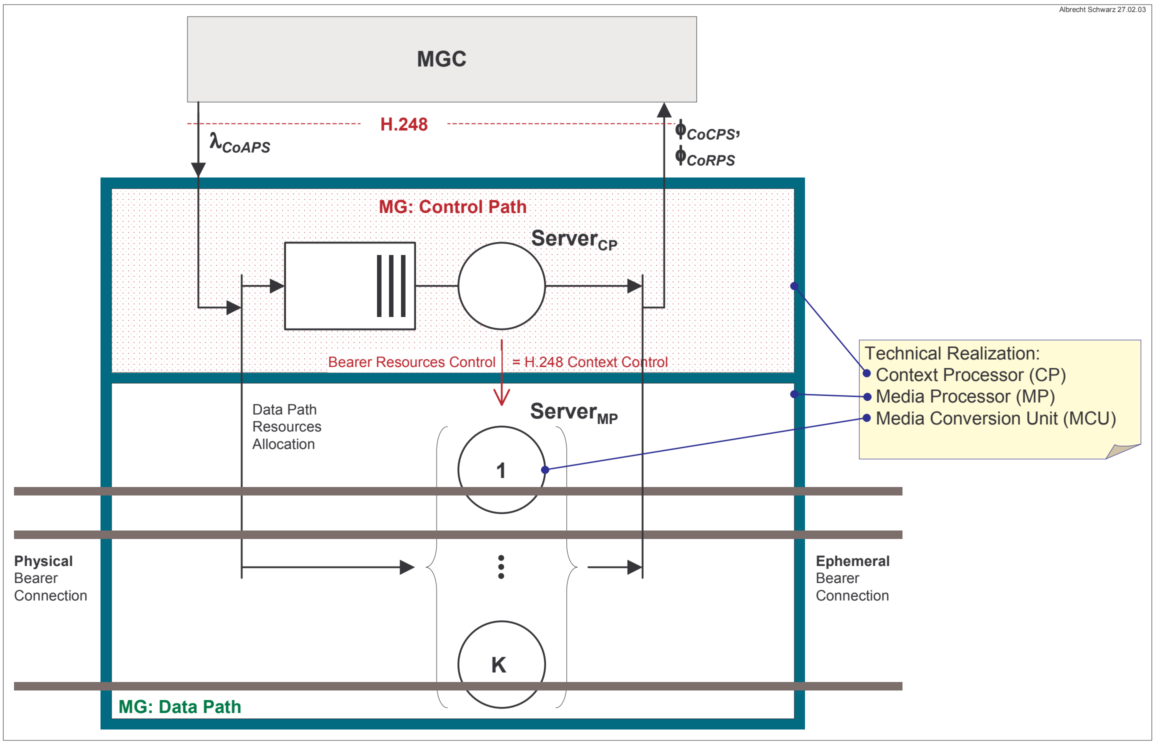

Figure 18: Traffic Model for H.248 MGs with Scope on C2X Type Sessions

The control path model is of type *waiting system*, allowing delayed access of H.248 traffic from MGC to get the MG Context Processor resource. The data path model is of type *loss system*; either there is still a free physical H.248 Termination, or all circuits are occupied (in case of C2X type H.248 Contexts).

The forking element at the ingress side shall point out, that a new H.248 Context attempt is MG-internally mapped on two service requests for the Context Control Processor and the Media Processor respectively ("a successful CSN call immediately needs a circuit").

The synchronization element at the egress side is related to the fact, that a completed H.248 Context event leads to the simultaneous de-allocation of the corresponding Media Conversion Unit.

NOTE 1– Technical realization: a Control Processor is typically realized by one or more **general purpose CPU**(s); a Media Processor may be for instance a **DSP device**, or a **DSP channel** in case of a high capacity DSP device.

NOTE 2– The qualitative traffic model is applicable for small and high capacity Media Gateways. The MG-internal organisation of the Media Conversion Units is not relevant in the scope of this recommendation. Principally there are three architectural approaches, primarily for H.248 Media Gateways intended for access or core network deployment: (1) circuit interface dedicated MCUs, (2) packet interface dedicated MCUs, or (3) interface independent MCU clusters ("resource pool").

12.3.2.1 Context Processor (CP) and Media Processor (MP) – Service Times

The traffic model implies that a MCU is allocated to a H.248 Context for the whole Context lifetime. Thus, the **MCU service time** $h_{MCU,Context}$ and **MP service time** $h_{MP,Context}$ is equal to the **Context holding time** C_OHT.

$$h_{MP,Context} = h_{MCU,Context} = C_O HT \quad \text{[s]}$$

Equation (29): Mean MCU/MP **service time** per **basic H.248 Context**

The principle proportion between the corresponding service times in control and data path of a H.248 MG system is:

$$h_{CP,Context} << h_{MP,Context}$$

Equation (30): Ratio between CP and MP service times

12.3.2.2 Context Processor (CP) and Media Processor (MP) – Capacity Ratio

The ideal **MCU capacity** $\mu_{MCU,Context,max}$ is, based Equation (29):

$$\mu_{MCU,Context,\max} = \frac{1}{C_O HT} \quad \text{[s}^{-1}\text{]}$$

Equation (31): Media Conversion Unit – Ideal **service rate** $\mu_{MCU,Context,max}$

The complete **MP** Context processing **capacity** $\mu_{MP,Context,max}$ is:

$$\mu_{MP,Context,\max} = K \cdot \mu_{MCU,Context,\max} = \frac{K}{C_O HT} \quad \text{[s}^{-1}\text{]}$$

Equation (32): Media Processor – Ideal **service rate** $\mu_{MP,Context,max}$

12.3.3 CSN Circuit Load versus Context Holding Time

One Media Conversion Unit is needed to serve a single circuit-switched interface. In case of a call, a MCU is allocated to the corresponding CSN interface.[15] A (concentrated or multiplexed) CSN interface is engineered for a mean **capacity** $A_{CSN,IF,Engineered}$ (also known as **link load** or **concentration factor**):

$$A_{CSN,IF,Engineered} = 1 - 0.x \quad \text{[Erl]}$$

Equation (33): CSN Interface – Engineered Load $A_{CSN,IF,Engineered}$

NOTE – Typical values for $A_{CSN,IF,Engineered}$ are in the range of 0.4 … 0.9 Erlang.

---

15 Circuit-Switched Network (CSN) interface types: analog line, analog trunk, digital line (= ISDN BRI), or digital trunk. H.248 Termination type ALN is intended for analog CSN interfaces, and type TDM is used for digital CSN interfaces.

12.3.4 CSN Circuit Load versus Context Control Load

The performance between H.248 MG control path and data path has to be appropriately balanced. The underlying design rule is that the system bottleneck may be chiefly the Media Processor. Meaning that the Context Processor should have still processing resources even then the Media Processor is fully occupied. This engineering concept has a feedback on the H.248 control load.

The meaningful maximum rate of Context attempts per second $\lambda_{CoAPS,Engineered}$ is (based on Equation (32) and Equation (33):

$$\lambda_{CoAPS,Engineered} = A_{CSN,IF,Engineered} \cdot \frac{K}{C_O HT} \quad [s^{-1}]$$

Equation (34): Context Processor – **Context Attempts Per Second** $\lambda_{CoAPS,Engineered}$

The resulting **Context** Control **Processor load** $A_{CP,Engineered}$ is (see also Equation (10), page 29):

$$A_{CP,Engineered} = \lambda_{CoAPS,Engineered} \cdot h_{CP,Context} \quad [Erl]$$

Equation (35): Context Processor – Engineered Load $A_{CP,Engineered}$

The corresponding **Media Processor load** $A_{MP,Engineered}$ is (based on Equation (29)):

$$A_{MP,Engineered} = \lambda_{CoAPS,Engineered} \cdot h_{MP,Context} = \lambda_{CoAPS,Engineered} \cdot C_O HT \quad [Erl]$$

Equation (36): Media Processor – Engineered Load $A_{MP,Engineered}$

In case of a load balancing mechanism for MCU resources within the MP will the resulting mean **Media Conversion Unit load** $A_{MCU,Engineered}$ correspond to:

$$A_{MCU,Engineered} = \frac{A_{MP,Engineered}}{K} \quad [Erl]$$

Equation (37): Media Conversion Unit – Engineered Load $A_{MCU,Engineered}$

12.3.5 Context Processor Performance versus Media Processor Farm Size

The Media Processor consists of *K* Media Conversion Units. The factor *K* shall be denoted as 'farm size' parameter.

The theoretical maximum capacities in control and data path are:

- Context Processor: $A_{CP,max}$ = 1 Erl. (for the single server model)
- Media Processor: $A_{MP,max}$ = *K* Erl. (for the K-server model)

The engineered CSN link load $A_{CSN,IF,Engineered}$ is typically resulting from network planning, for instance, engineering a link for certain grade of service parameters (like blocking probability). For specific MP architectures may be the farm size factor reduced through benefiting from economy of scales effect.

Editor's note: include example for economy of scale based farm size dimensioning

12.3.6 Exemplary Calculations

Some exemplary triangle relations between User plane capacity, MG data path size, and MG control performance.

12.3.6.1 MG Size Variation – ϕ_{CoCPS} = f(*K*)

The size of MGs may vary between small capacity towards high capacity systems. The size factor affects data and control path dimensioning. Farm size factor *K* is the prime data path parameter for C2X MG types.

How is the required control performance of the H.248 Context Processor dependent of the MG size? Equation (34), and the fact that every Context attempt must be completed, leads to following functional behaviour ϕ_{CoCPS} = f(*K*):

$$\phi_{CoCPS,Engineered}(K) = \frac{A_{CSN,IF,Engineered}}{C_O HT} \cdot K \quad [\mathrm{s}^{-1}]$$

Equation (38): Context Processor Performance as function of *K*

The control performance is **linearly** related with the CSN interface capacity, under the assumption of fixed concentration factor ($A_{CSN,IF,Engineered}$ = const.) and fixed Context holding time ($C_O HT$ = const.).

12.3.6.2 Link Load Variation – ϕ_{CoCPS} = f($A_{CSN,IF}$)

Equation (38) provides also the dependency of engineered concentration level at MG circuit interfaces:

$$\phi_{CoCPS,Engineered}(A_{CSN,IF}) = \frac{K}{C_O HT} \cdot A_{CSN,IF,Engineered} \quad [\mathrm{s}^{-1}]$$

Equation (39): Context Processor Performance as function of $A_{CSN,IF}$

The control performance is **linearly** related with the CSN interface concentration level, under the assumption of fixed MP farm size (K = const.) and fixed Context holding time ($C_O HT$ = const.).

12.3.6.3 Context Holding Time Variation – $\phi_{CoCPS} = f(C_O HT)$

The probability distributions functions for Context holding times are dependent of many parameters. Equation (38) provides also the principle dependency of the control performance from data path resource holding times:

$$\phi_{CoCPS,Engineered}(C_O HT) = K \cdot A_{CSN,IF,Engineered} \cdot \frac{1}{C_O HT} \quad [s^{-1}]$$

Equation (40): Context Processor Performance as function of $C_O HT$

The control performance is **hyperbolically** related with the mean Context holding time, under the assumption of fixed MP farm size (K = const.) and fixed concentration factor ($A_{CSN,IF,Engineered}$ = const.). This **non-linear behaviour** shall be elaborated in more detail in subsequent section 12.4.

12.4 Effective Throughput versus Context Holding Time – $\phi_{CoCPS} = f(C_O HT)$

The H.248 Context holding times are very service, market, and/or operator specific. Varying the mean holding time is impacting the Context processor performance. The overload control model of clause 12.2 allows the derivation of the principle behavior.

12.4.1 Derivation

Editor's note: include derivation

12.4.2 Results

The mean Completion rate for H.248 Contexts as a function of the H.248 Context holding time, - $\phi_{CoCPS} = f(C_O HT)$ -, is given for the three workload areas of the H.248 Context Control Processor by Equation (41):

$$\phi_{CoCPS} = f(C_O HT) = \begin{cases} \lambda_{CoAPS} = \dfrac{1}{C_O HT} & \text{for } C_O HT \geq \hat{h}_{CoC} \quad \text{Underloaded Server} \\ \dfrac{1-(\rho_{BL}+\rho_{HR})}{(1-\kappa)h_{CoC}} - \dfrac{\kappa}{1-\kappa} \cdot \dfrac{1}{C_O HT} & \text{for } \hat{h}_{RC} \leq C_O HT \leq \hat{h}_{CoC} \quad \text{Overloaded Server} \\ 0 & \text{for } C_O HT < \hat{h}_{CoR} \quad \text{Instable Server} \end{cases}$$

Equation (41): Context Throughput $\phi_{CoCPS} = f(C_O HT)$; with blind load handling; incl. static overhead & reserves

NOTE – The differences between Equation (40) and Equation (41) are: Equation (40) is firstly only valid for an underloaded Context Processor, and secondly derived from the specific control/data path traffic model for Circuit-to-X H:248 Contexts. Whereas Equation (41) is fairly general by considering MG control path only, or even applicable as model for an MGC level Context Processor.

With boundary value $\hat{h}_{CoC}$:

$$\hat{h}_{CoC} = \frac{1}{1-(\rho_{BL} + \rho_{HR})} h_{CoC}$$

Equation (42): Limit Parameter $\hat{h}_{CoC}$

and boundary value $\hat{h}_{CoR}$:

$$\hat{h}_{CoR} = \frac{1}{1-(\rho_{BL} + \rho_{HR})} h_{CoR}$$

Equation (43): Limit Parameter $\hat{h}_{CoR}$

Figure 19 is illustrating the functional behaviour characterized by Equation (41).

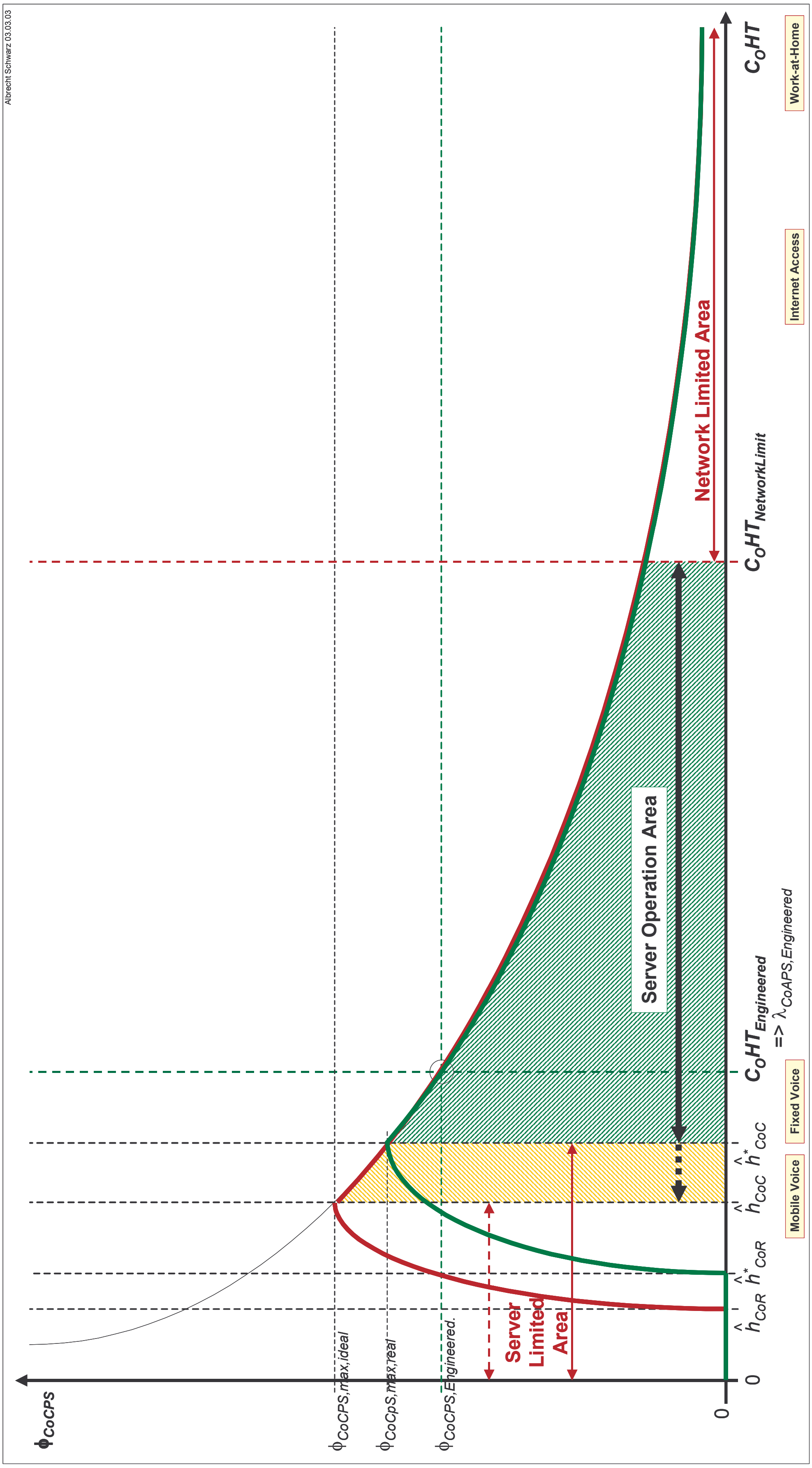

Figure 19: **Recommended Operation Area** for H.248 Context Processor

NOTE – At the bottom of **Figure 19** are some qualitative values for mean Context holding times of several services pointed out. Typically is $C_OHT_{MobileVoice} < C_OHT_{FixedVoice} < C_OHT_{InternetAccess} < C_OHT_{Work\text{-}at\text{-}Home}$ for the expectation values of the corresponding underlying probability distribution functions.

The system is **engineered** for the **operation point** $\{C_OHT_{Engineered} \mid \lambda_{CoAPS,Nominal}\}$, where $\lambda_{CoAPS,Nominal}$ (or $\lambda_{CoAPS,Engineered}$) specifies the nominal load or engineered capacity (in terms of the Context attempt arrival rate).

12.4.3 Conclusions

Equation (41) may be interpreted in the following way:

- Strong **nonlinear** dependency of achievable Context processing capacity versus average Context holding time (C_OHT)
- Range of applicable average C_OHTs is limited by theoretical maximum system capacity, and engineered network capacity!
- Linear relationship assumptions are only applicable for "very small" C_OHT ranges. Linearization should be applied with utmost caution!
- Network engineering: uncertainties concerning support of wider ranges of average C_OHT values (e.g., due to specific service distribution, call mixes, etc.) has to be supported by broader scalability ranges of Context Processor capacities.
- There is a **hyperbolical** relationship between effective throughput and holding time in the normal operation mode of the Context Processor (= state ‘underload’)

The useful Context Processor operation area is bounded by respecting network and system limitations.

NOTE – More background on network limited area and system limited area is indicated in [19], see also Fig. 5-3/GR-517-CORE.

13 Appendix III

Examples of Control Processing Capacity Computations

For further study.

14 Living List Items

14.1 Virtual Media Gateways

14.2 Exemplary Methodologies for Measuring Context Processing Capacities

14.3 Performance Impact of Binary versus Text Encoding

14.4 MG Context Processing Performance during Overload Situations

A Q.543 enhanced load regulation should be possible due to the potential co-operation principle between the MGC and the MG.

14.5 Repeated Context Attempts

e.g., model according E.501 Annex A

14.6 Measurement Points (MP), Reference Events

14.7 Grade of Service Requirements for H.248 MG

~~**Annex A**~~

###

15 Document History

Document History	
Issue	Notes
H.248.BHCoA Ed. 0.1	Initial version to WP2/16 Experts Meeting, Beijing, May 2004 Location: http://ftp3.itu.int/av-arch/avc-site/2001-2004/0405_Bei/AVD-2456.zip
H.Sup.BHCoA Ed. 0.2	Input version to WP2/16 Experts Meeting, Marysville, Feb/March 2005 Editorial enhancements with respect to Ed. 0.1 Location: http://ftp3.itu.int/av-arch/avc-site/2005-2008/0502_Mel/AVD-2699.zip
H.Sup.BHCoA Ed. 0.3	
H.Sup.BHCoA Ed. 0.4	Input version to ITU-T SG16 Plenary Meeting, July 2005 Location :

ChangeLog:

-