



low Energy COnsumption NETworks

ANNEX I OF THE DELIVERABLE D5.5

PROTOCOL EXTENSIONS

—

TECHNICAL DETAILS

Grant Agreement Number:	258454
Project Acronym:	ECONET
Project Title:	low Energy COnsumption NETworks
Funding Scheme:	Collaborative Project
Starting Date of the Project:	01/10/2010 <i>dd/mm/yyyy</i>
Duration:	36 months (original), 39 months (amendment request)
Project Coordinator:	Name: Raffaele Bolla Phone: +39 010 353 2075 Fax: +39 010 353 2154 e-mail: raffaele.bolla@unige.it

Due Date of Delivery:	M33 <i>Mx</i> (30/06/2013 <i>dd/mm/yyyy</i>)
Actual Date of Delivery:	30/06/2013 <i>dd/mm/yyyy</i>
Workpackage:	WP5 – <i>Green Strategies at the Control Plane</i>
Nature of the Deliverable:	R
Dissemination level:	PU
Editors:	ALU, CNIT, MLX, TEI, NVR, INFO
Version:	1.0

Disclaimer

The information, documentation and figures available in this deliverable are written by the ECONET Consortium partners under EC co-financing (project FP7-ICT-258454) and do not necessarily reflect the view of the European Commission.

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The reader uses the information at his/her sole risk and liability.

Copyright

Copyright © 2013 the ECONET Consortium. All rights reserved.

The ECONET Consortium consists of:

CONSORZIO NAZIONALE INTERUNIVERSITARIO PER LE TELECOMUNICAZIONI,

ALCATEL-LUCENT ITALIA S.p.A.,

MELLANOX TECHNOLOGIES LTD - MLNX,

LANTIQ Deutschland GmbH,

ERICSSON TELECOMUNICAZIONI,

TELECOM ITALIA S.p.A.,

GREEK RESEARCH AND TECHNOLOGY NETWORK S.A.,

NAUKOWA I AKADEMICKA SIEC KOMPUTEROWA,

DUBLIN CITY UNIVERSITY,

TEKNOLOGIAN TUTKIMUSKESKUS VTT,

POLITECHNIKA WARSZAWSKA,

*NETVISOR INFORMATIKAI ES KOMMUNIKACIOS ZARTKORUEN MUKODO
RESZVENYTARSASAG,*

ETHERNITY NETWORKS LTD,

LIGHTCOMM S.R.L.,

INFOCOM S.R.L.

This document may not be copied, reproduced or modified in whole or in part for any purpose without written permission from the ECONET Consortium. In addition to such written permission to copy, reproduce or modify this document in whole or part, an acknowledgement of the authors of the document and all applicable portions of the copyright notice must be clearly referenced.

Table of Contents

DISCLAIMER.....	1
COPYRIGHT	2
TABLE OF CONTENTS.....	3
1 ENERGY CAPABILITIES (PROFILES)	4
2 THE ECONET GAL DATA MODEL FOR TR-069.....	11
3 UPNP PROTOCOL FOR NETWORK CONNECTIVITY PROXY.....	19
3.1 UNIVERSAL PLUG & PLAY (UPnP)	19
3.1.1 UPnP Work Flow	19
3.1.2 Device Description File	20
3.2 NCP SERVICE – SUPPORTED ACTION DETAILS	23
3.2.1 PingRequest Action	23
3.2.2 DhcpRequest Action.....	24
3.2.3 WakeOnConnection Action	25
3.2.4 WakeOnPacket Action.....	26
3.2.5 SendReplyOnPacket Action.....	27
3.2.6 TcpKeepAlive Action.....	28
3.2.7 HeartBeating Action	30
3.2.8 TCPMigrationFreeze Action.....	33
3.2.9 TCPMigrationResume Action	37
3.2.10 Unsubscribe Action.....	41
3.3 LOW POWER SERVICE SUPPORTED ACTIONS DETAILS.....	42
3.3.1 GetPowerManagementInfo Action.....	42
3.3.2 Wakeup Action	43
3.3.3 GoToSleep Action	44
3.3.4 DefineSleepPeriod Action.....	45
4 XML SERVICES DESCRIPTIONS	46
4.1 NCP SERVICE XML DESCRIPTION	46
4.2 LP SERVICE XML DESCRIPTION	58

1 Energy capabilities (profiles)

On_MAX profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	ON	Maximum, Nominal Voltage	see table 1
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	OFF	IEEE1588 enabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_2	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_3	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_4	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_5	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_6	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_7	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_8	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

NOT EXTENSIBLE PROFILE

table 1		
axx-bpm 0x48 0 1000	"t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 1000	"t 1	1V0
axx-bpm 0x48 2 1800	"t 2	1V8 \n"
axx-bpm 0x48 3 1200	"t 3	1V2
axx-bpm 0x48 4 1650	"t 4	1V65 \n"
axx-bpm 0x48 5 2500	"t 5	2V5 \n"
axx-bpm 0x48 6 3300	"t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

On-Normal profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 2
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	On SM: Set 1 to CPLD register RESET_CTRL_1 bit 0 (RSTPHY0) and bit 1 (RSTPHY1).
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	Set 1 to CPLD register RESET_CTRL_7 bit 1-RSTZL30316
Reset_Traffic	OFF	Network Engine enabled	Set 1 to CPLD register RESET_CTRL_8 -bit 0-RSTMACPCIE
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_2	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_3	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_4	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_5	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_6	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_7	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_8	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

NOT EXTENSIBLE PROFILE

table 2		
axx-bpm 0x48 0 950	"t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"t 1	1V0
axx-bpm 0x48 2 1700	"t 2	1V8 \n"
axx-bpm 0x48 3 1100	"t 3	1V2
axx-bpm 0x48 4 1600	"t 4	1V65 \n"
axx-bpm 0x48 5 2350	"t 5	2V5 \n"
axx-bpm 0x48 6 3100	"t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_9G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	ON	Maximum, Nominal Voltage	see table 2
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	on SM: Set 1 to CPLD register RESET_CTRL_1 bit 0 (RSTPHY0) and bit 1 (RSTPHY1). Set 1 to CPLD register RESET_CTRL_7 bit 1 RSTZL30316 Set 1 to CPLD register RESET_CTRL_8 bit 0 RSTMACPCIE
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_2	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_3	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_4	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_5	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_6	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_7	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_8	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

EXTENSIBLE PROFILE

table 2		
axx-bpm 0x48 0 950	"t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"t 1	1V0
axx-bpm 0x48 1 1700	"t 2	1V8 \n"
axx-bpm 0x48 1 1100	"t 3	1V2
axx-bpm 0x48 1 1600	"t 4	1V65 \n"
axx-bpm 0x48 1 2350	"t 5	2V5 \n"
axx-bpm 0x48 1 3100	"t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_8G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 2
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_2	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_3	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_4	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_5	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_6	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_7	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_8	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

EXTENSIBLE PROFILE

table 2		
axx-bpm 0x48 0 950	"t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"t 1	1V0
axx-bpm 0x48 1 1700	"t 2	1V8 \n"
axx-bpm 0x48 1 1100	"t 3	1V2
axx-bpm 0x48 1 1600	"t 4	1V65 \n"
axx-bpm 0x48 1 2350	"t 5	2V5 \n"
axx-bpm 0x48 1 3100	"t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_7G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 2
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_2	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_3	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_4	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_5	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_6	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_7	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_8	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

EXTENSIBLE PROFILE

table 2

axx-bpm 0x48 0 950	"\t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"\t 1	1V0
axx-bpm 0x48 1 1700	"\t 2	1V8 \n"
axx-bpm 0x48 1 1100	"\t 3	1V2
axx-bpm 0x48 1 1600	"\t 4	1V65 \n"
axx-bpm 0x48 1 2350	"\t 5	2V5 \n"
axx-bpm 0x48 1 3100	"\t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_6G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 3
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_2	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_3	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_4	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_5	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_6	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_7	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_8	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

EXTENSIBLE PROFILE

table 3

axx-bpm 0x48 0 950	"\t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"\t 1	1V0
axx-bpm 0x48 2 1700	"\t 2	1V8 \n"
axx-bpm 0x48 3 1100	"\t 3	1V2
axx-bpm 0x48 4 1500	"\t 4	1V65 \n"
axx-bpm 0x48 5 2350	"\t 5	2V5 \n"
axx-bpm 0x48 6 3100	"\t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_5G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 3
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_2	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_3	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_4	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_5	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_6	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_7	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_8	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

EXTENSIBLE PROFILE

table 3		
axx-bpm 0x48 0 950	"\t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"\t 1	1V0
axx-bpm 0x48 1 1700	"\t 2	1V8 \n"
axx-bpm 0x48 1 1100	"\t 3	1V2
axx-bpm 0x48 1 1500	"\t 4	1V65 \n"
axx-bpm 0x48 1 2350	"\t 5	2V5 \n"
axx-bpm 0x48 1 3100	"\t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_4G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 3
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_2	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_3	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_4	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_5	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_6	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_7	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_8	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

EXTENSIBLE PROFILE

table 3		
axx-bpm 0x48 0 950	"\t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"\t 1	1V0
axx-bpm 0x48 1 1700	"\t 2	1V8 \n"
axx-bpm 0x48 1 1100	"\t 3	1V2
axx-bpm 0x48 1 1500	"\t 4	1V65 \n"
axx-bpm 0x48 1 2350	"\t 5	2V5 \n"
axx-bpm 0x48 1 3100	"\t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_3G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 3
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_2	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_3	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_4	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_5	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_6	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_7	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_8	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

EXTENSIBLE PROFILE

table 3		
axx-bpm 0x48 0 950	"\t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"\t 1	1V0
axx-bpm 0x48 1 1700	"\t 2	1V8 \n"
axx-bpm 0x48 1 1100	"\t 3	1V2
axx-bpm 0x48 1 1500	"\t 4	1V65 \n"
axx-bpm 0x48 1 2350	"\t 5	2V5 \n"
axx-bpm 0x48 1 3100	"\t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_2G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 3
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_2	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_3	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_4	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_5	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_6	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_7	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_8	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	
OptPort_9	ON	Port operational state ON, Laser ON, TX and RX ON	
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	

EXTENSIBLE PROFILE

table 3		
axx-bpm 0x48 0 950	"\t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"\t 1	1V0
axx-bpm 0x48 1 1700	"\t 2	1V8 \n"
axx-bpm 0x48 1 1100	"\t 3	1V2
axx-bpm 0x48 1 1500	"\t 4	1V65 \n"
axx-bpm 0x48 1 2350	"\t 5	2V5 \n"
axx-bpm 0x48 1 3100	"\t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Low_power_1G profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Maximum, Nominal Voltage	see table 3
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans at maximum speed	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	OFF	Whole ports SFP in power supply off	
OptPort_1	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x23 1 1 10
OptPort_2	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x23 1 1 20
OptPort_3	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x23 1 1 40
OptPort_4	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x23 1 1 80
OptPort_5	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x24 1 1 01
OptPort_6	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x24 1 1 02
OptPort_7	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x24 1 1 04
OptPort_8	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x24 1 1 08
OptPort_9	OFF	Port operational state ON, Laser OFF, TX OFF and RX ON	On SM: axx-i2c-write2 12 IOEXPS 0x24 1 1 10
OptPort_10	ON	Port operational state ON, Laser ON, TX and RX ON	On SM: to set axx-i2c-write2 12 IOEXPS 0x24 1 1 00

EXTENSIBLE PROFILE

table 3		
axx-bpm 0x48 0 950	"t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"t 1	1V0
axx-bpm 0x48 1 1700	"t 2	1V8 \n"
axx-bpm 0x48 1 1100	"t 3	1V2
axx-bpm 0x48 1 1500	"t 4	1V65 \n"
axx-bpm 0x48 1 2350	"t 5	2V5 \n"
axx-bpm 0x48 1 3100	"t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Fast_sleeping profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Reduced Supply Voltage	see table 3
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2 enabled	
Reset_Traffic	OFF	Network Engine enabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	ON	Whole ports SFP in power supply off	
OptPort_1	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_2	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_3	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_4	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_5	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_6	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_7	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_8	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_9	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	
OptPort_10	NOT_APPLICABLE	Port operational state OFF, Laser OFF, TX OFF and RX OFF	

NOT EXTENSIBLE PROFILE

table 3		
axx-bpm 0x48 0 950	"t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"t 1	1V0
axx-bpm 0x48 2 1700	"t 2	1V8 \n"
axx-bpm 0x48 3 1100	"t 3	1V2
axx-bpm 0x48 4 1500	"t 4	1V65 \n"
axx-bpm 0x48 5 2350	"t 5	2V5 \n"
axx-bpm 0x48 6 3100	"t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Deep_sleeping profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	REDUCED	Reduced Supply Voltage	see table 3
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans	
Reset_Synch	ON	IEEE1588 disabled	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2	
Reset_Traffic	ON	Network Engine disabled	
Reset_Board	OFF	Whole board enabled	
Reset_Whole_Ports	ON	Whole ports SFP in power supply off	
OptPort_1	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_2	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_3	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_4	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_5	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_6	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_7	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_8	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_9	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	
OptPort_10	NOT_APPLICABLE	Port operational state OFF,Laser OFF, TX OFF and RX OFF	

NOT EXTENSIBLE PROFILE

table 3		
axx-bpm 0x48 0 950	"t rail: 0 1V0_CPU \n"	
axx-bpm 0x48 1 950	"t 1	1V0
axx-bpm 0x48 1 1700	"t 2	1V8 \n"
axx-bpm 0x48 1 1100	"t 3	1V2
axx-bpm 0x48 1 1500	"t 4	1V65 \n"
axx-bpm 0x48 1 2350	"t 5	2V5 \n"
axx-bpm 0x48 1 3100	"t 6	3V3 \n"

0x48: msg BPM_RAIL_SET_VOLTAGE_REQ

Power_off profile

EA_OBJECT_TYPE	EA_OBJECT_ACTION_TYPE	Description	HW config
Voltage	NOT_APPLICABLE	Supply Voltage	
Clock	NOT_APPLICABLE	Clock Scaling Network Engine	
Fan	NOT_APPLICABLE	Fans	
Reset_Synch	NOT_APPLICABLE	IEEE1588	
Reset_Aux	NOT_APPLICABLE	Stand-by ECOM2	
Reset_Traffic	NOT_APPLICABLE	Network Engine	
Reset_Board	ON	Whole board shut down	Use the following XPRC commands: MSG_AUTO_INVOKE_SLOT__SHUTDOWN_ID in the CAPO process MSG_AUTO_INVOKE_SLOT__RESTART_ID in the CAPO process
Reset_Whole_Ports	NOT_APPLICABLE	Whole ports SFP in power supply off	
OptPort_1	NOT_APPLICABLE	Port operational state OFF	
OptPort_2	NOT_APPLICABLE	Port operational state OFF	
OptPort_3	NOT_APPLICABLE	Port operational state OFF	
OptPort_4	NOT_APPLICABLE	Port operational state OFF	
OptPort_5	NOT_APPLICABLE	Port operational state OFF	
OptPort_6	NOT_APPLICABLE	Port operational state OFF	
OptPort_7	NOT_APPLICABLE	Port operational state OFF	
OptPort_8	NOT_APPLICABLE	Port operational state OFF	
OptPort_9	NOT_APPLICABLE	Port operational state OFF	
OptPort_10	NOT_APPLICABLE	Port operational state OFF	

NOT EXTENSIBLE PROFILE

2 The ECONET GAL data model for TR-069.

```
<!--
Preliminary TR-069 DataModel Definition for ECONet Power Service
NETvisor Ltd.
-->
```

```
<dm:document xmlns:dm="urn:broadband-forum-org:cwmp:datamodel-1-0"
  xmlns:dmr="urn:broadband-forum-org:cwmp:datamodel-report-0-1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:broadband-forum-org:cwmp:datamodel-1-0 cwmp-datamodel-1-0.xsd
    urn:broadband-forum-org:cwmp:datamodel-report-0-1 cwmp-datamodel-report.xsd"
  spec="urn:netvisor-eu:econet-datamodel-1-0">
```

```
<description>
This CWMP data model defines the interface for the management (control and monitoring) of the power usage of CPE devices.
The model is derived from the results of the ECONET (low Energy CONsumption NETworks, FP7-ICT-258454) research project,
sponsored by the EC FP7 presearch program.
```

The models defines methods for controlling the the power economy behavior of devices, and also for monitoring the power modes and power usage of a device. The model allows for implementig these features in several compliance levels, i.e. profiles.

The Baseline profile requires the support for putting te device into one of several power states, and also for reading the monentary value of electric power and accumulated energy usage. These values may either be calculated or measured.

The Component profile provides capabilities for the inspection and control of power behavior of some part of the device.

The History profile extends this by the capability of getting historical data for power modes and usage. This profile makes it posible to implement CPE power monitoring in large, provider-wide scale CWMP-ACS systems, where the frequency of data exchange with the individual CPE-s is rather low, (i.e. several hours or days).

```
</description>
```

```
<import file="tr-069-biblio.xml" spec="urn:broadband-forum-org:tr-069-biblio"/>
```

```
<!--
```

```
<!-- ECONET Data Types
```

```
<!--
```

```
<dataType name="ECONetControlPolicySource">
```

```
<description>
Represents the authority for defining and changing the power states for the device.
Options MUST be in the format defined in {{bibref|TR-106}}. For example: "X_EXAMPLE-COM_MyControlPolicy".
</description>
```

```
<string>
<enumeration value="X_ECONET_Autonomous"/>
<enumeration value="X_ECONET_NetworkControlled"/>
</string>
```

```
</dataType>
```

```
<dataType name="ECONetPowerState">
```

```
<description>
Power/standby states enumerations
</description>
```

```
<string>
<enumeration value="X_ECONET_S3_OFF">
  <description> Mechanically off, with only manual transition supported </description>
</enumeration>
<enumeration value="X_ECONET_S2_SENTRY">
  <description> Using minimal power to monitor inputs for wakeup signals.
    Wakeup time from this state is typically above 1 second</description>
</enumeration>
<enumeration value="X_ECONET_S1_LPIDLE">
  <description> Not processing/forwarding data, but ready to do so within less than 1 second</description>
</enumeration>
<enumeration value="X_ECONET_S0_ACTIVE">
  <description> Ready for immediate processing/forwarding of data at least on some ports/interfaces</description>
</enumeration>
```

```

</string>

</dataType>

<dataType name="ECONetAdminPowerState" base="ECONetPowerState">

  <description>
    Administrative power state, the state defined for the device by the current authority
    Note that this type extends enum constants of {{ECONetPowerState}}.
  </description>

  <string>
    <enumeration value="X_ECONET_DEFAULT">
      <description> The power state has not been set yet</description>
    </enumeration>

  </string>

</dataType>

<dataType name="ECONetPerfMode">

  <description>
    Performace mode (i.e. Power scaling mode), based on definition of ECONET D4.1
  </description>

  <string>
    <enumeration value="X_ECONET_P0_FULLPERF">
      <description> Providing full performance without consideration for power usage </description>
    </enumeration>
    <enumeration value="X_ECONET_P1_BALANCED">
      <description> A reasonable optimisation of performance vs. power usage</description>
    </enumeration>
    <enumeration value="X_ECONET_P2_LPECO">
      <description> Minimal performance provided at minimal power </description>
    </enumeration>
    <enumeration value="X_ECONET_P3_DISABLED">
      <description> This component is completely disabled and provides no service until activated externally</description>
    </enumeration>

  </string>

</dataType>

<dataType name="ECONetEnergyMeter">

  <description>
    A counter-like quantity for representing total the total electrical energy used by the Component.
    The value may be measured or calculated
    This counter is reset whenever the corresponding Uptime metric is reset.
    Units are watts * seconds = Joules
  </description>

  <unsignedLong>
    <units value="Joules" />
  </unsignedLong>

</dataType>

<dataType name="ECONetPowerWattage">

  <description>
    Power used by a Device or Component
  </description>

  <unsignedLong>
    <units value="milliWatts" />
  </unsignedLong>

</dataType>

<dataType name="ECONetEnableState">

  <description>
    Enable/Disable flag.
  </description>

  <boolean />

```

```

</dataType>

<dataType name="ECONetUptime">

  <description>
    Uptime in {{units}}.
  </description>

  <unsignedLong>
    <units value="seconds" />
  </unsignedLong>

</dataType>

<dataType name="ECONetComponentLocalID">

  <description>
    Local ID of Components.
  </description>

  <string>
    <size maxLength="64"/>
    <pattern value="[a-zA-Z_]\w*(\d+)?"/>
  </string>

</dataType>

<!--
===== -->
<!-- ECONET Power Service Model -->
<!--
===== -->

<model name="X_NETvisor-EU_ECONetPowerService:1.0" isService="true">

  <!-- ..... -->
  <!-- ECONET Objects -->

  <object name="X_NETvisor-EU_ECONetPowerService.{i}." access="readOnly" minEntries="1" maxEntries="1" enableParameter="Enabled">
    <description>
      The Service Object of a ECONetPower Service supporting device.
    </description>

    <parameter name="Enabled" access="readWrite">
      <description>
        Enables or disables the entire ECONet Power mechanism.
      </description>
      <syntax>
        <dataType ref="ECONetEnableState"/>
      </syntax>
    </parameter>

    <parameter name="Uptime" access="readOnly">
      <description>
        Uptime in {{units}} of the ECONet Power mechanism if it is enabled.
      </description>
      <syntax>
        <dataType ref="ECONetUptime"/>
      </syntax>
    </parameter>

    <parameter name="AdminPowerState" access="readWrite">
      <description>
        The administrative power/standby state of the ECONet Power mechanisms.
        Write access is only supported in a non-autonomus Control Policy mode
      </description>
      <syntax>
        <dataType ref="ECONetAdminPowerState"/>
        <default type="object" value="X_ECONET_DEFAULT"/>
      </syntax>
    </parameter>

    <parameter name="PowerState" access="readOnly">
      <description>
        The power state currently in effect.
      </description>
      <syntax>
        <dataType ref="ECONetPowerState"/>
      </syntax>
    </parameter>
  </object>

```

```

</parameter>

<parameter name="MomentaryWattage" access="readOnly">
  <description>
    Estimation for the momentary power used by the entire Device.
  </description>
  <syntax>
    <dataType ref="ECONetPowerWattage"/>
  </syntax>
</parameter>

<parameter name="MeteredEnergy" access="readOnly">
  <description>
    The current accumulated power usage for the entire Device, (calculated since Uptime was reset).
  </description>
  <syntax>
    <dataType ref="ECONetEnergyMeter"/>
  </syntax>
</parameter>

<parameter name="ControlPolicy" access="readWrite">
  <description>
    The control policy of the ECONet Power service.
  </description>
  <syntax>
    <dataType ref="ECONetControlPolicySource"/>
    <default type="object" value="X_ECONET_Autonomous"/></default>
  </syntax>
</parameter>

<parameter name="ComponentNumberOfEntries" access="readOnly">
  <description>
    The number of instances of {{object}.Component.{i}}.
  </description>
  <syntax>
    <unsignedInt/>
  </syntax>
</parameter>
</object>

<object name="X_NETvisor-EU_ECONetPowerService.{i}.Component.{i}." access="readOnly" minEntries="0" maxEntries="unbounded"
numEntriesParameter="ComponentNumberOfEntries" enableParameter="Enabled">
  <description>
    The Component of a ECONetPower Service supporting device.
    A Component can be any part of the device, including the Device as a whole, in which case the ID is '/'
    LocalID is 'Root'.
  </description>

  <uniqueKey>
    <parameter ref="ID"/>
  </uniqueKey>

  <parameter name="ID" access="readOnly">
    <description>
      Unique ID of the Component, suitable for addressing and exploring the Component hierarchy.
      Recommended categories are
      'Port' for any type of port and communication interface
      'Proc' for any component (like a CPU, memory block, FPGA, MCU) supporting
      some central processing functionality
      'Power' modules providing power supply and conversion, e.g. AC/DC and DC/DC
      converters
      E.g.
      '/' for the Device
      '/Ports/Ata1' the second telephony port
      '/Ports/Wlan0' the first Wlan card
      '/Ports/Usb0' the first USB port
      '/Power/Dc0' a power supply/converter unit within the device
      '/Proc/0' the main CPU unit
    </description>
    <syntax>
      <string>
        <size maxLength="64"/>
      </string>
    </syntax>
  </parameter>

  <parameter name="LocalID" access="readOnly">
    <description>

```

Unique abbreviated ID of the Component.

Eg. Wlan0, Dc0, Usb0

```
</description>
<syntax>
  <dataType ref="ECONetComponentLocalID"/>
</syntax>
</parameter>
```

```
<parameter name="Description" access="readWrite">
```

```
<description>
```

Short free-text description of the Component.

E.g. 'MainBoard CPU', 'Ethernet 4'

```
</description>
<syntax>
  <string>
    <size maxLength="1024"/>
  </string>
</syntax>
</parameter>
```

```
<parameter name="Enabled" access="readWrite">
```

```
<description>
```

Enables or disables the ECONet Power mechanism for the entire Component.

```
</description>
```

```
<syntax>
```

```
<dataType ref="ECONetEnableState"/>
```

```
</syntax>
```

```
</parameter>
```

```
<parameter name="Uptime" access="readOnly">
```

```
<description>
```

Uptime in {{units}} of the ECONet Power mechanism for the Component if it is enabled.

```
</description>
```

```
<syntax>
```

```
<dataType ref="ECONetUptime"/>
```

```
</syntax>
```

```
</parameter>
```

```
<parameter name="PerformanceMode" access="readWrite">
```

```
<description>
```

The actual performance mode/power state of the Component.

Write access is only supported in a non-autonomus Control Policy mode

```
</description>
```

```
<syntax>
```

```
<dataType ref="ECONetPerfMode"/>
```

```
</syntax>
```

```
</parameter>
```

```
<parameter name="MomentaryWattage" access="readOnly">
```

```
<description>
```

Estimation for the momentary power used by the Component.

```
</description>
```

```
<syntax>
```

```
<dataType ref="ECONetPowerWattage"/>
```

```
</syntax>
```

```
</parameter>
```

```
<parameter name="MeteredEnergy" access="readOnly">
```

```
<description>
```

The current accumulated power in {{units}} of the Component, (calculated since its Uptime was reset).

```
</description>
```

```
<syntax>
```

```
<dataType ref="ECONetEnergyMeter"/>
```

```
</syntax>
```

```
</parameter>
```

```
</object>
```

```
<object name="X_NETvisor-EU_ECONetPowerService.{i}.Component.{i}.History." access="readOnly" minEntries="1" maxEntries="1"
enableParameter="Enabled">
```

```
<description>
```

The History definition of the Component of a ECONetPower Service supporting device.

History contains a FIFO list of History items, with each item representing certain time period.

These items are expected to fully and uniquely cover the total time period available in the History.

A Period may be closed (and new Period opened) either periodically (see Period, below), or upon a state

change.

within a 10-minute section, those are recorded in separate Periods.

```

</description>

<parameter name="Enabled" access="readWrite">
  <description>
    Enables or disables the ECONet Power History mechanism for the entire Component.
  </description>
  <syntax>
    <dataType ref="ECONetEnableState"/>
  </syntax>
</parameter>

<parameter name="Interval" access="readWrite">
  <description>
    The interval in { {units} } for History Items.
  </description>
  <syntax>
    <unsignedInt>
      <range minInclusive="1" maxInclusive="3600"/>
      <units value="seconds"></units>
    </unsignedInt>
  </syntax>
</parameter>

<parameter name="MaximumItemNumberOfEntries" access="readWrite">
  <description>
    The maximum number of instances of { {object}.Item.{i}. }.
  </description>
  <syntax>
    <unsignedInt>
      <range minInclusive="1" maxInclusive="100"/>
    </unsignedInt>
  </syntax>
</parameter>

<parameter name="ItemNumberOfEntries" access="readOnly">
  <description>
    The actual/available number of instances of { {object}.Item.{i}. }.
  </description>
  <syntax>
    <unsignedInt/>
  </syntax>
</parameter>
</object>

<object name="X_NETvisor-EU_ECONetPowerService.{i}.Component.{i}.History.Item.{i}."
maxEntries="unbounded" numEntriesParameter="ItemNumberOfEntries">
  <description>
    The Component History Item of a ECONetPower Service supporting device.
  </description>

  <uniqueKey>
    <parameter ref="StartTime"/>
    <parameter ref="EndTime"/>
  </uniqueKey>

  <parameter name="StartTime" access="readOnly">
    <description>
      Start time of the period covered by this history item.
    </description>
    <syntax>
      <dateTime/>
    </syntax>
  </parameter>

  <parameter name="EndTime" access="readOnly">
    <description>
      End time of the period covered by this history item.
    </description>
    <syntax>
      <dateTime/>
    </syntax>
  </parameter>

  <parameter name="PowerState" access="readOnly">
    <description>

```



```

    The power state of the Device.
</description>
<syntax>
  <dataType ref="ECONetPowerState"/>
</syntax>
</parameter>

<parameter name="PerfMode" access="readOnly">
  <description>
    The performance mode of the Component.
  </description>
  <syntax>
    <dataType ref="ECONetPowerState"/>
  </syntax>
</parameter>

<parameter name="PowerStateChangeReason" access="readOnly">
  <description>
    The descriptive reason of power state change.
  </description>
  <syntax>
    <string/>
  </syntax>
</parameter>

<parameter name="MeteredEnergy" access="readOnly">
  <description>
    The energy meter value recorded at EndTime,
  </description>
  <syntax>
    <dataType ref="ECONetEnergyMeter"/>
  </syntax>
</parameter>
</object>

<!-- ..... -->
<!-- ECONET Profiles -->

<profile name="Baseline:1">

  <object ref="X_NETvisor-EU_ECONetPowerService.{i}." requirement="present">
    <parameter ref="Enabled" requirement="readOnly"/>
    <parameter ref="Uptime" requirement="readOnly"/>
    <parameter ref="PowerState" requirement="readWrite"/>
    <parameter ref="AdminPowerState" requirement="readWrite"/>
    <parameter ref="ControlPolicy" requirement="readWrite"/>
    <parameter ref="MomentaryWattage" requirement="readOnly"/>
    <parameter ref="MeteredEnergy" requirement="readOnly"/>
  </object>

</profile>

<profile name="Component:1">

  <object ref="X_NETvisor-EU_ECONetPowerService.{i}." requirement="present">
    <parameter ref="ComponentNumberOfEntries" requirement="readOnly"/>
  </object>

  <object ref="X_NETvisor-EU_ECONetPowerService.{i}.Component.{i}." requirement="present">
    <parameter ref="Enabled" requirement="readOnly"/>
    <parameter ref="ID" requirement="readOnly"/>
    <parameter ref="LocalID" requirement="readOnly"/>
    <parameter ref="Description" requirement="readOnly"/>
    <parameter ref="Uptime" requirement="readOnly"/>
    <parameter ref="PowerState" requirement="readOnly"/>
    <parameter ref="MomentaryWattage" requirement="readOnly"/>
    <parameter ref="MeteredEnergy" requirement="readOnly"/>
  </object>

</profile>

<profile name="History:1">

  <object ref="X_NETvisor-EU_ECONetPowerService.{i}." requirement="present">
    <parameter ref="ComponentNumberOfEntries" requirement="readOnly"/>
  </object>

  <object ref="X_NETvisor-EU_ECONetPowerService.{i}.Component.{i}." requirement="present">

```

```
<parameter ref="ID" requirement="readOnly"/>
<parameter ref="LocalID" requirement="readOnly"/>
<parameter ref="Description" requirement="readOnly"/>
</object>

<object ref="X_NETvisor-EU_ECONetPowerService.{i}.Component.{i}.History." requirement="present">
  <parameter ref="Enabled" requirement="readOnly"/>
  <parameter ref="Interval" requirement="readOnly"/>
  <parameter ref="MaximumItemNumberOfEntries" requirement="readOnly"/>
  <parameter ref="ItemNumberOfEntries" requirement="readOnly"/>
</object>

<object ref="X_NETvisor-EU_ECONetPowerService.{i}.Component.{i}.History.Item.{i}." requirement="present">
  <parameter ref="StartTime" requirement="readOnly"/>
  <parameter ref="EndTime" requirement="readOnly"/>
  <parameter ref="PowerState" requirement="readOnly"/>
  <parameter ref="PowerStateChangeReason" requirement="readOnly"/>
  <parameter ref="EnergyMeter" requirement="readOnly"/>
</object>
</profile>
</model>
</dm:document>
```

3 UPnP Protocol for Network Connectivity Proxy

3.1 Universal Plug & Play (UPnP)

Universal Plug and Play (UPnP) is a set of networking protocols that provide interoperability and seamless communication between the IP based devices. UPnP standard has been adopted in many different products covering printers, scanners, Internet gateways, copier machines, WiFi access points, PC and mobile devices. The UPnP standard allows devices to automatically acquire network configurations and seamlessly discover each other's presence in the local networks and communicate or exchange data [1]. The primary objectives of the UPnP standard are to provide interoperability, auto-discovery and seamless communications between the heterogeneous devices in the residential networks [2].

3.1.1 UPnP Work Flow

There are two roles in the UPnP architecture: controlled devices and control points (CPs). According to the UPnP Devices Architecture (UDA), the controlled devices are physical entities that implements one or more services and periodically advertise their presence over the network. The service is a logical functional unit that receives instructions from the CP, performs some actions and notifies the CP about the changes in state variables. The CP retrieves the controlled device and its services description, invokes actions implemented by the services, calls service state variables and receives event notifications from the services [1]. The UPnP protocol workflow can be briefly described in the following steps.

Step 0 Addressing: The UPnP device gets an IP address.

Step 1 Discovery: The UPnP controlled device uses Simple Service Discovery Protocol (SSDP) to advertise its services to the CP. Similarly, the CP also uses SSDP to find the controlled devices of interest.

Step 2 Description: The CP retrieves the controlled device and service descriptions from the URL provided by the device in the discovery message. The description is expressed in a XML file that contains complete information about the controlled device.

Step 3 Control: The CP uses Simple Object Access Protocol (SOAP) to send an action to the services offered by the controlled device. The control messages are expressed in the XML format.

Step 4 Eventing: The action performed in response of control messages may causes changes in the value of state variables. Services use eventing messages to publish updates about these state variables. Eventing messages are also expressed in XML format.

Step 5 Presentation: The CP retrieves the controlled device information page from the presentation URL of the controlled device which may allow users to control and/or just view the controlled device status.

[1] UPnP Forum. UPnP Device Architecture 1.0. October 15, 2008. Available: <http://upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0.pdf>.

[2] UPnP Forum. UPnP AV Architecture: 1. June 25, 2002. Available: <http://www.upnp.org/specs/av/UPnP-av-AVArchitecture-v1-20020625.pdf>.

3.1.2 Device Description File

The UPnP device description is an XML file that contains different pieces of vender specific information, definitions of all logical devices, URL for the provided services and URL for both control and eventing [32]. The XML description file for a generic device is illustrated below. Some placeholders are specified by UPnP Forum working committee (Red colored) or by a UPnP vendor (Purple colored). Elements defined by the UDA are colored green.

UPnP Device Description

```
<?xml version="1.0"?>
<root xmlns="urn:schemas-upnp-org:device-1-0" configId="configuration number">
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <device>
    <deviceType>urn:schemas-upnp-org:device:deviceType:v</deviceType>
    <friendlyName>short user-friendly title</friendlyName>
    <manufacturer>manufacturer name</manufacturer>
    <manufacturerURL>URL to manufacturer site</manufacturerURL>
    <modelDescription>long user-friendly title</modelDescription>
    <modelName>model name</modelName>
    <modelNumber>model number</modelNumber>
    <modelURL>URL to model site</modelURL>
    <serialNumber>manufacturer's serial number</serialNumber>
    <UDN>uuid:UUID</UDN>
    <serviceList>
      <service>
        <serviceType>urn:schemas-upnp-
org:service:serviceType:v</serviceType>
        <serviceId>urn:upnp-org:serviceId:serviceID</serviceId>
        <SCPDURL>URL to service description</SCPDURL>
        <controlURL>URL for control</controlURL>
        <eventSubURL>URL for eventing</eventSubURL>
      </service>
      <!-- Declarations of other services if any. -->
    </serviceList>
    <deviceList>
      <!-- Description of embedded devices if any. -->
    </deviceList>
  </device>
</root>
```

Brief description of different description file elements, attributes and values is presented below.

<?xml>

This is required field for all XML files and is case sensitive.

<root>

This is required case sensitive field and must have “urn:schemas-upnp-org:device-1-0” as the value for the xmlns attribute. The configId specifies the configuration number to which the device description belongs.

<specVersion>

This is required field that defines the lowest version of the UPnP architecture on which the device can be implemented. The value is specified in terms of major and minor versions. Both of them should be 1 if the device is implemented on UPnP Device Architecture 1.1.

<device>

This is required field. The device may contain one or more embedded device.

<deviceType>

This is required field and specify the device type. It must begin with “urn:schemas-upnp-org:device:” followed by the standardized device type suffix, a colon, and an integer device version.

<friendlyName>

This is required field and contains short description for end user about the device.

<manufacturer>

This is required field and contains short description about the device manufacturer.

<manufacturerURL>

This is optional field and contains website of the manufacturer.

<modelDescription>

This is optional but recommended field and contains information about the specific device model.

<modelName>

This is required field and mentions the device model.

<modelName>

This is required field and contains the device model number.

<modelURL>

This is optional field and contains website of the device specific model.

<serialNumber>

This is recommended field and contains the device serial number.

<UDN>

This is required field and contains Universally Unique Identifier (UUID) for the device.

<serviceList>

Contains the list of service provided by the device.

<service>

This is optional field and repeated one for each service defined by the device.

<serviceType>

This is required field and specify the type of service. It must begin with “urn:schemas-upnp-org:service:” followed by the standardized service type suffix, colon, and an integer service version.

<serviceId>

This is required field and must be unique within this device description to uniquely identify the service. It must begin with “urn:upnp-org:serviceId:” followed by a service ID suffix.

<SCPDURL>

This is required field and contains URL for service description.

<controlURL>

This is required field and contains relative URL for control.

<eventSubURL>

This is required field and contains relative URL for eventing. This value can be empty if the service has no evented variables.

<deviceList>

This field is required if and only if the root device has embedded devices.

<presentationURL>

This is required field and contains relative URL for presentation of device.

3.2 NCP Service – Supported action details

3.2.1 PingRequest Action

Table 1 describes the required arguments for the PingRequest action. The rest of device description can be retrieved from UUID that are stored by NCP during device registration. The XML definition of PingRequest action is described below.

Table 1. PingRequest action.

| Argument Name | Direction | Description |
|---------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| SubId | Output | SubId is assigned by the NCP to each action during registration. It uniquely identifies registered action and can be used by power managed device to unsubscribe the specific action with the NCP service if desirable. |

Ping Action Description

```

<action>
  <name>PingRequest</name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name>SubId</name>
      <direction>out</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.2.2 DhcpRequest Action

Table 2 describes the required arguments for the DhcpRequest action. The rest of device description can be retrieved from UUID that are stored by NCP during device registration. The XML definition of DhcpRequest action is described below.

Table 2. DhcpRequest action.

| Argument Name | Direction | Description |
|---------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| SubId | Output | SubId is assigned by the NCP to each action during registration. It uniquely identifies registered action and can be used by power managed device to unsubscribe the specific action with the NCP service if desirable. |

DHCP Action Description

```

<action>
  <name> DhcpRequest </name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name>SubId</name>
      <direction>out</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>

```


3.2.3 WakeOnConnection Action

Table 3 describes the required arguments for the WakeOnConnection action. The XML definition of WakeOnConnection action is described below.

Table 3. WakeOnConnection action.

| Argument Name | Direction | Description |
|-----------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| Protocol | Input | The transport protocol for new connections. |
| Port | Input | The port number on which new connection is received. |
| SubId | Output | SubId is assigned by the NCP to each action during registration. It uniquely identifies registered action and can be used by power managed device to unsubscribe the specific action with the NCP service if desirable. |

WakeOnConnection Action Description

```

<action>
  <name>WakeOnConnection</name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name>Protocol</name>
      <direction>in</direction>
      <relatedStateVariable>Protocol</relatedStateVariable>
    </argument>
    <argument>
      <name>Port</name>
      <direction>in</direction>
      <relatedStateVariable>Port</relatedStateVariable>
    </argument>
    <argument>
      <name>SubId</name>
      <direction>out</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.2.4 WakeOnPacket Action

Table 4 describes the required arguments for the WakeOnPacket action. The XML definition of WakeOnPacket action is described below.

Table 4. WakeOnPacket action.

| Argument Name | Direction | Description |
|----------------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| Payload | Input | The pattern to be looked inside packet. |
| TrailerLength | Input | The offset from the end of packet. |
| SubId | Output | SubId is assigned by the NCP to each action during registration. It uniquely identifies registered action and can be used by power managed device to unsubscribe the specific action with the NCP service if desirable. |

WakeOnPacket Action Description

```

<action>
  <name> WakeOnPacket</name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name> Payload </name>
      <direction>in</direction>
      <relatedStateVariable> Payload </relatedStateVariable>
    </argument>
    <argument>
      <name> TrailerLength </name>
      <direction>in</direction>
      <relatedStateVariable> TrailerLength </relatedStateVariable>
    </argument>
    <argument>
      <name>SubId</name>
      <direction>out</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.2.5 SendReplyOnPacket Action

Table 5 describes the required arguments for the SendReplyOnPacket action. The XML definition of SendReplyOnPacket action is described below.

Table 5. SendReplyOnPacket action.

| Argument Name | Direction | Description |
|----------------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| Payload | Input | The pattern to be looked inside packet. |
| TrailerLength | Input | The offset from the end of packet. |
| Reply | Input | The response for the received packet. |
| SubId | Output | SubId is assigned by the NCP to each action during registration. It uniquely identifies registered action and can be used by power managed device to unsubscribe the specific action with the NCP service if desirable. |

SendReplyOnPacket Action Description

```

<action>
  <name> SendReplyOnPacket</name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name> Payload </name>
      <direction>in</direction>
      <relatedStateVariable> Payload </relatedStateVariable>
    </argument>
    <argument>
      <name> Reply </name>
      <direction>in</direction>
      <relatedStateVariable> Reply </relatedStateVariable>
    </argument>
    <argument>
      <name> TrailerLength </name>
      <direction>in</direction>
      <relatedStateVariable> TrailerLength </relatedStateVariable>
    </argument>
    <argument>
      <name>SubId</name>
      <direction>out</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.2.6 TcpKeepAlive Action

Table 6 describes the required arguments for the TCPKeepAlive action. The XML definition of TCPKeepAlive action is described below.

Table 6. TCPKeepAlive action.

| Argument Name | Direction | Description |
|-------------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| PeerIP | Input | The IP address of the remote peer of the connection. |
| LocalPort | Input | The port used by the power managed device for the given TCP session. |
| RemotePort | Input | The TCP port used by the remote peer. |
| Behavior | Input | The action in response to new data arrived from remote peer (wake-up, buffer, delay). |
| TimeOut | Input | Maximum delay before the power managed device is woken up in case of new data and buffer/delay Behavior. |
| WakeOnPush | Input | Specifies wake-up when urgent data or push data arrives from the remote peer. |
| SubId | Output | SubId is assigned by the NCP to each action during registration. It uniquely identifies registered action and can be used by power managed device to unsubscribe the specific action with the NCP service if desirable. |

TCPKeepAlive Action Description

```

<action>
  <name> TCPKeepAlive </name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name> PeerIP </name>
      <direction>in</direction>
      <relatedStateVariable> PeerIP </relatedStateVariable>
    </argument>
    <argument>
      <name> LocalPort </name>
      <direction>in</direction>
      <relatedStateVariable> LocalPort </relatedStateVariable>
    </argument>
    <argument>
      <name> RemotePort </name>
      <direction>in</direction>
      <relatedStateVariable> RemotePort </relatedStateVariable>
    </argument>
    <argument>
      <name> Behavior </name>
      <direction>in</direction>
      <relatedStateVariable> Behavior </relatedStateVariable>
    </argument>
    <argument>
      <name> TimeOut </name>
      <direction>in</direction>
      <relatedStateVariable> TimeOut </relatedStateVariable>
    </argument>
    <argument>
      <name> WakeOnPush </name>
      <direction>in</direction>
      <relatedStateVariable> WakeOnPush </relatedStateVariable>
    </argument>
    <argument>
      <name>SubId</name>
      <direction>out</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.2.7 HeartBeating Action

Table 7 describes the required arguments for the HeartBeating action. The XML definition of HeartBeating action is described below.

Table 7. HeartBeating action.

| Argument Name | Direction | Description |
|-----------------------|-----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| PeerIP | Input | The IP address of the remote peer of the connection. |
| Protocol | Input | Represents the transport protocol to be used for building heart-beating messages. |
| LocalPort | Input | The port used by the power managed device. |
| RemotePort | Input | The TCP port used by the remote peer. |
| MsgTemplate | Input | Represents the template for heart-beating message. |
| VariableFields | Input | Represents the position and type of variable fields in the packet template. |
| Period | Input | The time interval at which heartbeat messages must be sent. |
| FirstRun | Input | Time from now when the first heartbeat message must be sent. |
| MsgFilter | Input | The filter for the incoming packets to detect the heart-beat message. |
| FilterOffset | Input | The position from the beginning of data where the filter is applied. |
| OpaqueFields | Input | The fields to be ignored when comparing the received packet with the filter specified by the application. |
| SubId | Output | SubId is assigned by the NCP to each action during registration. It uniquely identifies registered action and can be used by power managed device to unsubscribe the specific action with the NCP service if desirable. |

HeartBeating Action Description (1)

```

<action>
  <name> HeartBeating </name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name> PeerIP </name>
      <direction>in</direction>
      <relatedStateVariable> PeerIP </relatedStateVariable>
    </argument>
    <argument>
      <name> Protocol </name>
      <direction>in</direction>
      <relatedStateVariable> Protocol </relatedStateVariable>
    </argument>
    <argument>
      <name> LocalPort </name>
      <direction>in</direction>
      <relatedStateVariable> LocalPort </relatedStateVariable>
    </argument>
    <argument>
      <name> RemotePort </name>
      <direction>in</direction>
      <relatedStateVariable> RemotePort </relatedStateVariable>
    </argument>
    <argument>
      <name> MsgTemplate </name>
      <direction>in</direction>
      <relatedStateVariable> MsgTemplate </relatedStateVariable>
    </argument>
    <argument>
      <name> VariableFields </name>
      <direction>in</direction>
      <relatedStateVariable> VariableFields </relatedStateVariable>
    </argument>
    <argument>
      <name> Period </name>
      <direction>in</direction>
      <relatedStateVariable> Period </relatedStateVariable>
    </argument>
  </argumentList>

```

HeartBeating Action Description (2)

```

<argument>
  <name> FirstRun </name>
  <direction>in</direction>
  <relatedStateVariable> FirstRun </relatedStateVariable>
</argument>
<argument>
  <name> MsgFilter </name>
  <direction>in</direction>
  <relatedStateVariable> MsgFilter </relatedStateVariable>
</argument>
<argument>
  <name> FilterOffset </name>
  <direction>in</direction>
  <relatedStateVariable> FilterOffset </relatedStateVariable>
</argument>
<argument>
  <name> OpaqueFields </name>
  <direction>in</direction>
  <relatedStateVariable> OpaqueFields </relatedStateVariable>
</argument>
<argument>
  <name>SubId</name>
  <direction>out</direction>
  <relatedStateVariable>SubID</relatedStateVariable>
</argument>
</argumentList>
</action>

```


3.2.8 TCPMigrationFreeze Action

The low power managed devices can request the TCPMigrationFreeze action from the NCP service. This action freezes the active TCP session at low power managed device and migrates and restores the TCP session at the NCP. The application needs to provide all arguments required for the TCPMigrationFreeze action. Table 8 describes the required arguments for the TCPMigrationFreeze action. The XML definition of TCPMigrationFreeze action is described below.

Table 8. TCPMigrationFreeze action.

| Argument Name | Direction | Description |
|---------------------|-----------|-----------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| SubId | Input | The HeartBeating action SubId that is previously registered at NCP by the low power managed device. |
| SessionID | Input | The current working session ID of the NCP. |
| SrcIP | Input | The source IP address of the TCP session. |
| DstIP | Input | The destination IP address of the TCP session. |
| SrcPort | Input | Source port number of the TCP session. |
| DstPort | Input | Destination port number of the TCP session. |
| Proto | Input | The protocol (IP encoding) e.g., TCP. |
| Type | Input | Connection type: Datagram or stream. |
| Family | Input | IP family: INET/INET6/... |
| InQLen | Input | Input queue length. |
| OutQLen | Input | Output queue length. |
| InQSeq | Input | Input queue sequence number. |
| InQBuffSize | Input | Input queue buffer size. |
| InQBuff | Input | Input queue data. |
| OutQSeq | Input | Output queue sequence number. |
| OutQBuffSize | Input | Output queue buffer size. |
| OutQBuff | Input | Output queue data. |
| OptMask | Input | TCPI_OPT_bits. |
| SndWScale | Input | Sender window scale. |
| MssClamp | Input | MSS clamp. |
| HasRcvWScale | Input | Receiver window scale in use. True if the RcvWScale field is valid. |
| RcvWScale | Input | Receiver window scale. |
| HasTimeStamp | Input | Timestamp option in use. True if the TimeStamp field is valid. |
| TimeStamp | Input | Timestamp value for the connection. |

TCPMigrationFreeze Action Description (1)

```

<action>
  <name> TCPMigrationFreeze </name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name> SubId </name>
      <direction>in</direction>
      <relatedStateVariable> SubId </relatedStateVariable>
    </argument>
    <argument>
      <name> SessionID </name>
      <direction>in</direction>
      <relatedStateVariable> SessionID </relatedStateVariable>
    </argument>
    <argument>
      <name> SrcIP </name>
      <direction>in</direction>
      <relatedStateVariable> SrcIP </relatedStateVariable>
    </argument>
    <argument>
      <name> DstIP </name>
      <direction>in</direction>
      <relatedStateVariable> DstIP </relatedStateVariable>
    </argument>
    <argument>
      <name> SrcPort </name>
      <direction>in</direction>
      <relatedStateVariable> SrcPort </relatedStateVariable>
    </argument>
    <argument>
      <name> DstPort </name>
      <direction>in</direction>
      <relatedStateVariable> DstPort </relatedStateVariable>
    </argument>
    <argument>
      <name> Proto </name>
      <direction>in</direction>
      <relatedStateVariable> Proto </relatedStateVariable>
    </argument>
  </argumentList>

```

TCPMigrationFreeze Action Description (2)

```

<argument>
  <name> Type </name>
  <direction>in</direction>
  <relatedStateVariable> Type </relatedStateVariable>
</argument>
<argument>
  <name> Family </name>
  <direction>in</direction>
  <relatedStateVariable> Family </relatedStateVariable>
</argument>
<argument>
  <name> InQLen </name>
  <direction>in</direction>
  <relatedStateVariable> InQLen </relatedStateVariable>
</argument>
<argument>
  <name> OutQLen </name>
  <direction>in</direction>
  <relatedStateVariable> OutQLen </relatedStateVariable>
</argument>
<argument>
  <name> InQSeq </name>
  <direction>in</direction>
  <relatedStateVariable> InQSeq </relatedStateVariable>
</argument>
<argument>
  <name> InQBuffSize </name>
  <direction>in</direction>
  <relatedStateVariable> InQBuffSize </relatedStateVariable>
</argument>
<argument>
  <name> InQBuff </name>
  <direction>in</direction>
  <relatedStateVariable> InQBuff </relatedStateVariable>
</argument>
<argument>
  <name> OutQSeq </name>
  <direction>in</direction>
  <relatedStateVariable> OutQSeq </relatedStateVariable>
</argument>
<argument>
  <name> OutQBuffSize </name>
  <direction>in</direction>
  <relatedStateVariable> OutQBuffSize </relatedStateVariable>
</argument>

```

TCPMigrationFreeze Action Description (3)

```

<argument>
  <name> OutQBuff </name>
  <direction>in</direction>
  <relatedStateVariable> OutQBuff </relatedStateVariable>
</argument>
<argument>
  <name> OptMask </name>
  <direction>in</direction>
  <relatedStateVariable> OptMask </relatedStateVariable>
</argument>
<argument>
  <name> SndWScale </name>
  <direction>in</direction>
  <relatedStateVariable> SndWScale </relatedStateVariable>
</argument>
<argument>
  <name> MssClamp </name>
  <direction>in</direction>
  <relatedStateVariable> MssClamp </relatedStateVariable>
</argument>
<argument>
  <name> HasRcvWScale </name>
  <direction>in</direction>
  <relatedStateVariable> HasRcvWScale </relatedStateVariable>
</argument>
<argument>
  <name> RcvWScale </name>
  <direction>in</direction>
  <relatedStateVariable> RcvWScale </relatedStateVariable>
</argument>
<argument>
  <name> HasTimeStamp </name>
  <direction>in</direction>
  <relatedStateVariable> HasTimeStamp </relatedStateVariable>
</argument>
<argument>
  <name> TimeStamp </name>
  <direction>in</direction>
  <relatedStateVariable> TimeStamp </relatedStateVariable>
</argument>
</argumentList>
</action>

```

3.2.9 TCPMigrationResume Action

The low power managed devices can request the TCPMigrationResume action from the NCP service. This action returns TCP session state from NCP and stores it at low power managed device. Table 9 describes the required arguments for the TCPMigrationResume action. The XML definition of TCPMigrationResume action is described below.

Table 9. TCPMigrationResume action.

| Argument Name | Direction | Description |
|---------------------|-----------|-----------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| SubId | Input | The HeartBeating action SubId that is previously registered at NCP by the low power managed device. |
| SessionID | Input | The current working session ID of the NCP. |
| SrcIP | Output | The source IP address of the TCP session. |
| DstIP | Output | The destination IP address of the TCP session. |
| SrcPort | Output | Source port number of the TCP session. |
| DstPort | Output | Destination port number of the TCP session. |
| Proto | Output | The protocol (IP encoding) e.g., TCP. |
| Type | Output | Connection type: Datagram or stream. |
| Family | Output | IP family: INET/INET6/... |
| InQLen | Output | Input queue length. |
| OutQLen | Output | Output queue length. |
| InQSeq | Output | Input queue sequence number. |
| InQBuffSize | Output | Input queue buffer size. |
| InQBuff | Output | Input queue data. |
| OutQSeq | Output | Output queue sequence number. |
| OutQBuffSize | Output | Output queue buffer size. |
| OutQBuff | Output | Output queue data. |
| OptMask | Output | TCPI_OPT_bits. |
| SndWScale | Output | Sender window scale. |
| MssClamp | Output | MSS clamp. |
| HasRcvWScale | Output | Receiver window scale in use. True if the RcvWScale field is valid. |
| RcvWScale | Output | Receiver window scale. |
| HasTimeStamp | Output | Timestamp option in use. True if the TimeStamp field is valid. |
| TimeStamp | Output | Timestamp value for the connection. |

TCPMigrationResume Action Description (1)

```

<action>
  <name> TCPMigrationResume </name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name> SubId </name>
      <direction>in</direction>
      <relatedStateVariable> SubId </relatedStateVariable>
    </argument>
    <argument>
      <name> SessionID </name>
      <direction>in</direction>
      <relatedStateVariable> SessionID </relatedStateVariable>
    </argument>
    <argument>
      <name> SrcIP </name>
      <direction>out</direction>
      <relatedStateVariable> SrcIP </relatedStateVariable>
    </argument>
    <argument>
      <name> DstIP </name>
      <direction>out</direction>
      <relatedStateVariable> DstIP </relatedStateVariable>
    </argument>
    <argument>
      <name> SrcPort </name>
      <direction> out </direction>
      <relatedStateVariable> SrcPort </relatedStateVariable>
    </argument>
    <argument>
      <name> DstPort </name>
      <direction> out </direction>
      <relatedStateVariable> DstPort </relatedStateVariable>
    </argument>
    <argument>
      <name> Proto </name>
      <direction> out </direction>
      <relatedStateVariable> Proto </relatedStateVariable>
    </argument>
  </argumentList>

```

TCPMigrationResume Action Description (2)

```

<argument>
  <name> Type </name>
  <direction> out </direction>
  <relatedStateVariable> Type </relatedStateVariable>
</argument>
<argument>
  <name> Family </name>
  <direction> out </direction>
  <relatedStateVariable> Family </relatedStateVariable>
</argument>
<argument>
  <name> InQLen </name>
  <direction> out </direction>
  <relatedStateVariable> InQLen </relatedStateVariable>
</argument>
<argument>
  <name> OutQLen </name>
  <direction> out </direction>
  <relatedStateVariable> OutQLen </relatedStateVariable>
</argument>
<argument>
  <name> InQSeq </name>
  <direction> out </direction>
  <relatedStateVariable> InQSeq </relatedStateVariable>
</argument>
<argument>
  <name> InQBuffSize </name>
  <direction> out </direction>
  <relatedStateVariable> InQBuffSize </relatedStateVariable>
</argument>
<argument>
  <name> InQBuff </name>
  <direction> out </direction>
  <relatedStateVariable> InQBuff </relatedStateVariable>
</argument>
<argument>
  <name> OutQSeq </name>
  <direction> out </direction>
  <relatedStateVariable> OutQSeq </relatedStateVariable>
</argument>
<argument>
  <name> OutQBuffSize </name>
  <direction> out </direction>
  <relatedStateVariable> OutQBuffSize </relatedStateVariable>
</argument>

```

TCPMigrationResume Action Description (3)

```

<argument>
  <name> OutQBuff </name>
  <direction> out </direction>
  <relatedStateVariable> OutQBuff </relatedStateVariable>
</argument>
<argument>
  <name> OptMask </name>
  <direction> out </direction>
  <relatedStateVariable> OptMask </relatedStateVariable>
</argument>
<argument>
  <name> SndWScale </name>
  <direction> out </direction>
  <relatedStateVariable> SndWScale </relatedStateVariable>
</argument>
<argument>
  <name> MssClamp </name>
  <direction> out </direction>
  <relatedStateVariable> MssClamp </relatedStateVariable>
</argument>
<argument>
  <name> HasRcvWScale </name>
  <direction> out </direction>
  <relatedStateVariable> HasRcvWScale </relatedStateVariable>
</argument>
<argument>
  <name> RcvWScale </name>
  <direction> out </direction>
  <relatedStateVariable> RcvWScale </relatedStateVariable>
</argument>
<argument>
  <name> HasTimeStamp </name>
  <direction> out </direction>
  <relatedStateVariable> HasTimeStamp </relatedStateVariable>
</argument>
<argument>
  <name> TimeStamp </name>
  <direction> out </direction>
  <relatedStateVariable> TimeStamp </relatedStateVariable>
</argument>
</argumentList>
</action>

```


3.2.10 Unsubscribe Action

The low power managed devices can request to Unsubscribe an action at the NCP service. The Unsubscribe action enables the NCP to de-register the previously subscribed action. Table 10 describes the required arguments for the Unsubscribe action. The XML definition of Unsubscribe action is described below.

Table 10. Unsubscribe action.

| Argument Name | Direction | Description |
|---------------|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| UUID | Input | UUID is required to uniquely identify the low power managed device that has requested the action. |
| SubId | Input | SubId is assigned by the NCP to each action during registration. It uniquely identifies registered action and can be used by power managed device to unsubscribe the specific action with the NCP service. |

Unsubscribe Action Description

```

<action>
  <name> Unsubscribe </name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name>SubId</name>
      <direction>in</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.3 Low Power Service Supported Actions details

3.3.1 GetPowerManagementInfo Action

Table 13 describes the required arguments for the GetPowerManagementInfo action. The XML definition of GetPowerManagementInfo action is described below.

Table 13. GetPowerManagementInfo action.

| Argument Name | Direction | Description |
|--------------------------|-----------|----------------------------------------------------------------------------------------------------------------------------------------|
| WakeUpMethod | Output | Bearer-dependent method to wake up the device. |
| PowerSupplyStatus | Output | The description of power supplies status, both external and internal (connected/disconnected, battery charge, battery lifetime, etc.). |

GetPowerManagementInfo Action Description

```

<action>
  <name> GetPowerManagementInfo </name>
  <argumentList>
    <argument>
      <name>WakeUpMethod</name>
      <direction>out</direction>
      <relatedStateVariable> WakeUpMethod </relatedStateVariable>
    </argument>
    <argument>
      <name>PowerSupplyStatus</name>
      <direction> out </direction>
      <relatedStateVariable> PowerSupplyStatus </relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.3.2 Wakeup Action

Table 14 describes the required arguments for the Wakeup action. The XML definition of Wakeup action is described below.

Table 14. Wakeup action.

| Argument Name | Direction | Description |
|-------------------|-----------|----------------------------|
| PowerState | Output | Power state of the device. |

WakeUp Action Description

```

<action>
  <name> Wakeup </name>
  <argumentList>
    <argument>
      <name>PowerState</name>
      <direction>out</direction>
      <relatedStateVariable> PowerState </relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.3.3 GoToSleep Action

The NCP can request the GoToSleep action from the LP service. This action provides recommendation to the low power managed device to enter into specified sleep state for the specified period. The low power managed device may decline to sleep if it is busy or performing some operations. Table 15 describes the required arguments for the GoToSleep action. The XML definition of GoToSleep action is described below.

Table 15. GoToSleep action.

| Argument Name | Direction | Description |
|-------------------------------|-----------|--------------------------------------------------------------|
| RecommendedSleepPeriod | Input | The recommended duration to sleep. |
| RecommendedPowerState | Input | The recommended sleep state. |
| SleepPeriod | Output | The sleeping period before the device autonomously wakes up. |
| PowerState | Output | Power state of the device. |

GoToSleep Action Description

```

<action>
  <name> GoToSleep </name>
  <argumentList>
    <argument>
      <name>RecommendedSleepPeriod</name>
      <direction>in</direction>
      <relatedStateVariable> SleepPeriod </relatedStateVariable>
    </argument>
    <argument>
      <name> RecommendedPowerState</name>
      <direction> in </direction>
      <relatedStateVariable> PowerState </relatedStateVariable>
    </argument>
    <argument>
      <name>SleepPeriod</name>
      <direction>out</direction>
      <relatedStateVariable> SleepPeriod </relatedStateVariable>
    </argument>
    <argument>
      <name> PowerState</name>
      <direction> out </direction>
      <relatedStateVariable> PowerState </relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

3.3.4 DefineSleepPeriod Action

Table 16 describes the required arguments for the DefineSleepPeriod action. The XML definition of DefineSleepPeriod action is described below.

Table 16. DefineSleepPeriod action.

| Argument Name | Direction | Description |
|------------------------------------|-----------|----------------------------------------------------|
| RecommendedSleepPeriodStart | Input | The recommended start time for the sleep duration. |
| RecommendedSleepPeriodEnd | Input | The recommended end time for the sleep duration. |
| RecommendedPowerState | Input | The recommended sleep state. |

DefineSleepPeriod Action Description

```

<action>
  <name> DefineSleepPeriod </name>
  <argumentList>
    <argument>
      <name>RecommendedSleepPeriodStart</name>
      <direction>in</direction>
      <relatedStateVariable> SleepPeriodStart </relatedStateVariable>
    </argument>
    <argument>
      <name>RecommendedSleepPeriodEnd</name>
      <direction>in</direction>
      <relatedStateVariable> SleepPeriodEnd </relatedStateVariable>
    </argument>
    <argument>
      <name> RecommendedPowerState</name>
      <direction> in </direction>
      <relatedStateVariable> PowerState </relatedStateVariable>
    </argument>
  </argumentList>
</action>

```

4 XML Services Descriptions

4.1 NCP Service XML Description

```
<?xml version="1.0" encoding="UTF-8"?>
<scpd xmlns="urn:schemas-upnp-org:service-1-0">
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <actionList>
    <action>
      <name>PingRequest</name>
      <argumentList>
        <argument>
          <name>Uuid</name>
          <direction>in</direction>
          <relatedStateVariable>UUID</relatedStateVariable>
        </argument>
        <argument>
          <name>SubId</name>
          <direction>out</direction>
          <relatedStateVariable>SubID</relatedStateVariable>
        </argument>
      </argumentList>
    </action>
    <action>
      <name> DhcpRequest </name>
      <argumentList>
        <argument>
          <name>Uuid</name>
          <direction>in</direction>
          <relatedStateVariable>UUID</relatedStateVariable>
        </argument>
        <argument>
          <name>SubId</name>
          <direction>out</direction>
          <relatedStateVariable>SubID</relatedStateVariable>
        </argument>
      </argumentList>
    </action>
    <action>
      <name>WakeOnConnection</name>
      <argumentList>
        <argument>
          <name>Uuid</name>
          <direction>in</direction>
          <relatedStateVariable>UUID</relatedStateVariable>
        </argument>
        <argument>
          <name>Protocol</name>
          <direction>in</direction>
          <relatedStateVariable>Protocol</relatedStateVariable>
        </argument>
        <argument>
          <name>Port</name>

```

```

        <direction>in</direction>
        <relatedStateVariable>Port</relatedStateVariable>
    </argument>
    <argument>
        <name>SubId</name>
        <direction>out</direction>
        <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
</argumentList>
</action>
<action>
    <name> WakeOnPacket</name>
    <argumentList>
        <argument>
            <name>Uuid</name>
            <direction>in</direction>
            <relatedStateVariable>UUID</relatedStateVariable>
        </argument>
        <argument>
            <name> Payload </name>
            <direction>in</direction>
            <relatedStateVariable> Payload </relatedStateVariable>
        </argument>
        <argument>
            <name> TrailerLength </name>
            <direction>in</direction>
            <relatedStateVariable> TrailerLength </relatedStateVariable>
        </argument>
        <argument>
            <name>SubId</name>
            <direction>out</direction>
            <relatedStateVariable>SubID</relatedStateVariable>
        </argument>
    </argumentList>
</action>
<action>
    <name> SendReplyOnPacket</name>
    <argumentList>
        <argument>
            <name>Uuid</name>
            <direction>in</direction>
            <relatedStateVariable>UUID</relatedStateVariable>
        </argument>
        <argument>
            <name> Payload </name>
            <direction>in</direction>
            <relatedStateVariable> Payload </relatedStateVariable>
        </argument>
        <argument>
            <name> Reply </name>
            <direction>in</direction>
            <relatedStateVariable> Reply </relatedStateVariable>
        </argument>
        <argument>
            <name> TrailerLength </name>
            <direction>in</direction>
            <relatedStateVariable> TrailerLength </relatedStateVariable>
        </argument>
    </argumentList>

```

```

    <argument>
      <name>SubId</name>
      <direction>out</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>
<action>
  <name> TCPKeepAlive </name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>
      <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
      <name> PeerIP </name>
      <direction>in</direction>
      <relatedStateVariable> PeerIP </relatedStateVariable>
    </argument>
    <argument>
      <name> LocalPort </name>
      <direction>in</direction>
      <relatedStateVariable> LocalPort </relatedStateVariable>
    </argument>
    <argument>
      <name> RemotePort </name>
      <direction>in</direction>
      <relatedStateVariable> RemotePort </relatedStateVariable>
    </argument>
    <argument>
      <name> Behavior </name>
      <direction>in</direction>
      <relatedStateVariable> Behavior </relatedStateVariable>
    </argument>
    <argument>
      <name> TimeOut </name>
      <direction>in</direction>
      <relatedStateVariable> TimeOut </relatedStateVariable>
    </argument>
    <argument>
      <name> WakeOnPush </name>
      <direction>in</direction>
      <relatedStateVariable> WakeOnPush </relatedStateVariable>
    </argument>
  </argumentList>
</action>
<action>
  <name> HeartBeating </name>
  <argumentList>
    <argument>
      <name>SubId</name>
      <direction>out</direction>
      <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
  </argumentList>
</action>
<action>
  <name> HeartBeating </name>
  <argumentList>
    <argument>
      <name>Uuid</name>
      <direction>in</direction>

```



```

        <relatedStateVariable>UUID</relatedStateVariable>
    </argument>
    <argument>
        <name> PeerIP </name>
        <direction>in</direction>
        <relatedStateVariable> PeerIP </relatedStateVariable>
    </argument>
    <argument>
        <name> Protocol </name>
        <direction>in</direction>
        <relatedStateVariable> Protocol </relatedStateVariable>
    </argument>
    <argument>
        <name> LocalPort </name>
        <direction>in</direction>
        <relatedStateVariable> LocalPort </relatedStateVariable>
    </argument>
    <argument>
        <name> RemotePort </name>
        <direction>in</direction>
        <relatedStateVariable> RemotePort </relatedStateVariable>
    </argument>
    <argument>
        <name> MsgTemplate </name>
        <direction>in</direction>
        <relatedStateVariable> MsgTemplate </relatedStateVariable>
    </argument>
    <argument>
        <name> VariableFields </name>
        <direction>in</direction>
        <relatedStateVariable> VariableFields </relatedStateVariable>
    </argument>
    <argument>
        <name> Period </name>
        <direction>in</direction>
        <relatedStateVariable> Period </relatedStateVariable>
    </argument>
    <argument>
        <name> FirstRun </name>
        <direction>in</direction>
        <relatedStateVariable> FirstRun </relatedStateVariable>
    </argument>
    <argument>
        <name> MsgFilter </name>
        <direction>in</direction>
        <relatedStateVariable> MsgFilter </relatedStateVariable>
    </argument>
    <argument>
        <name> FilterOffset </name>
        <direction>in</direction>
        <relatedStateVariable> FilterOffset </relatedStateVariable>
    </argument>
    <argument>
        <name> OpaqueFields </name>
        <direction>in</direction>
        <relatedStateVariable> OpaqueFields </relatedStateVariable>
    </argument>
    <argument>

```

```

        <name>SubId</name>
        <direction>out</direction>
        <relatedStateVariable>SubID</relatedStateVariable>
    </argument>
</argumentList>
</action>
<action>
    <name>TCPMigrationFreeze</name>
    <argumentList>
        <argument>
            <name>Uuid</name>
            <direction>in</direction>
            <relatedStateVariable>UUID</relatedStateVariable>
        </argument>
        <argument>
            <name>SubId</name>
            <direction>in</direction>
            <relatedStateVariable>SubId</relatedStateVariable>
        </argument>
        <argument>
            <name>SessionID</name>
            <direction>in</direction>
            <relatedStateVariable>SessionID</relatedStateVariable>
        </argument>
        <argument>
            <name>SrcIP</name>
            <direction>in</direction>
            <relatedStateVariable>SrcIP</relatedStateVariable>
        </argument>
        <argument>
            <name>DstIP</name>
            <direction>in</direction>
            <relatedStateVariable>DstIP</relatedStateVariable>
        </argument>
        <argument>
            <name>SrcPort</name>
            <direction>in</direction>
            <relatedStateVariable>SrcPort</relatedStateVariable>
        </argument>
        <argument>
            <name>DstPort</name>
            <direction>in</direction>
            <relatedStateVariable>DstPort</relatedStateVariable>
        </argument>
        <argument>
            <name>Protocol</name>
            <direction>in</direction>
            <relatedStateVariable>Protocol</relatedStateVariable>
        </argument>
        <argument>
            <name>Type</name>
            <direction>in</direction>
            <relatedStateVariable>Type</relatedStateVariable>
        </argument>
        <argument>
            <name>Family</name>
            <direction>in</direction>
            <relatedStateVariable>Family</relatedStateVariable>
    </argumentList>
</action>

```

```

</argument>
<argument>
  <name> InQLen </name>
  <direction>in</direction>
  <relatedStateVariable> InQLen </relatedStateVariable>
</argument>
<argument>
  <name> OutQLen </name>
  <direction>in</direction>
  <relatedStateVariable> OutQLen </relatedStateVariable>
</argument>
<argument>
  <name> InQSeq </name>
  <direction>in</direction>
  <relatedStateVariable> InQSeq </relatedStateVariable>
</argument>
<argument>
  <name> InQBuffSize </name>
  <direction>in</direction>
  <relatedStateVariable> InQBuffSize </relatedStateVariable>
</argument>
<argument>
  <name> InQBuff </name>
  <direction>in</direction>
  <relatedStateVariable> InQBuff </relatedStateVariable>
</argument>
<argument>
  <name> OutQSeq </name>
  <direction>in</direction>
  <relatedStateVariable> OutQSeq </relatedStateVariable>
</argument>
<argument>
  <name> OutQBuffSize </name>
  <direction>in</direction>
  <relatedStateVariable> OutQBuffSize </relatedStateVariable>
</argument>
<argument>
  <name> OutQBuff </name>
  <direction>in</direction>
  <relatedStateVariable> OutQBuff </relatedStateVariable>
</argument>
<argument>
  <name> OptMask </name>
  <direction>in</direction>
  <relatedStateVariable> OptMask </relatedStateVariable>
</argument>
<argument>
  <name> SndWScale </name>
  <direction>in</direction>
  <relatedStateVariable> SndWScale </relatedStateVariable>
</argument>
<argument>
  <name> MssClamp </name>
  <direction>in</direction>
  <relatedStateVariable> MssClamp </relatedStateVariable>
</argument>
<argument>
  <name> HasRcvWScale </name>

```

```

        <direction>in</direction>
        <relatedStateVariable> HasRcvWScale </relatedStateVariable>
    </argument>
    <argument>
        <name> RcvWScale </name>
        <direction>in</direction>
        <relatedStateVariable> RcvWScale </relatedStateVariable>
    </argument>
    <argument>
        <name> HasTimeStamp </name>
        <direction>in</direction>
        <relatedStateVariable> HasTimeStamp </relatedStateVariable>
    </argument>
    <argument>
        <name> TimeStamp </name>
        <direction>in</direction>
        <relatedStateVariable> TimeStamp </relatedStateVariable>
    </argument>
</argumentList>
</action>
<action>
    <name> TCPMigrationResume </name>
    <argumentList>
        <argument>
            <name> Uuid</name>
            <direction>in</direction>
            <relatedStateVariable> UUID</relatedStateVariable>
        </argument>
        <argument>
            <name> SubId </name>
            <direction>in</direction>
            <relatedStateVariable> SubId </relatedStateVariable>
        </argument>
        <argument>
            <name> SessionID </name>
            <direction>in</direction>
            <relatedStateVariable> SessionID </relatedStateVariable>
        </argument>
        <argument>
            <name> SrcIP </name>
            <direction>out</direction>
            <relatedStateVariable> SrcIP </relatedStateVariable>
        </argument>
        <argument>
            <name> DstIP </name>
            <direction>out</direction>
            <relatedStateVariable> DstIP </relatedStateVariable>
        </argument>
        <argument>
            <name> SrcPort </name>
            <direction> out </direction>
            <relatedStateVariable> SrcPort </relatedStateVariable>
        </argument>
        <argument>
            <name> DstPort </name>
            <direction> out </direction>
            <relatedStateVariable> DstPort </relatedStateVariable>
        </argument>
    </argumentList>
</action>

```

```

<argument>
  <name> Protocol </name>
  <direction> out </direction>
  <relatedStateVariable> Protocol </relatedStateVariable>
</argument>
<argument>
  <name> Type </name>
  <direction> out </direction>
  <relatedStateVariable> Type </relatedStateVariable>
</argument>
<argument>
  <name> Family </name>
  <direction> out </direction>
  <relatedStateVariable> Family </relatedStateVariable>
</argument>
<argument>
  <name> InQLen </name>
  <direction> out </direction>
  <relatedStateVariable> InQLen </relatedStateVariable>
</argument>
<argument>
  <name> OutQLen </name>
  <direction> out </direction>
  <relatedStateVariable> OutQLen </relatedStateVariable>
</argument>
<argument>
  <name> InQSeq </name>
  <direction> out </direction>
  <relatedStateVariable> InQSeq </relatedStateVariable>
</argument>
<argument>
  <name> InQBuffSize </name>
  <direction> out </direction>
  <relatedStateVariable> InQBuffSize </relatedStateVariable>
</argument>
<argument>
  <name> InQBuff </name>
  <direction> out </direction>
  <relatedStateVariable> InQBuff </relatedStateVariable>
</argument>
<argument>
  <name> OutQSeq </name>
  <direction> out </direction>
  <relatedStateVariable> OutQSeq </relatedStateVariable>
</argument>
<argument>
  <name> OutQBuffSize </name>
  <direction> out </direction>
  <relatedStateVariable> OutQBuffSize </relatedStateVariable>
</argument>
<argument>
  <name> OutQBuff </name>
  <direction> out </direction>
  <relatedStateVariable> OutQBuff </relatedStateVariable>
</argument>
<argument>
  <name> OptMask </name>
  <direction> out </direction>

```

```

        <relatedStateVariable> OptMask </relatedStateVariable>
    </argument>
    <argument>
        <name> SndWScale </name>
        <direction> out </direction>
        <relatedStateVariable> SndWScale </relatedStateVariable>
    </argument>
    <argument>
        <name> MssClamp </name>
        <direction> out </direction>
        <relatedStateVariable> MssClamp </relatedStateVariable>
    </argument>
    <argument>
        <name> HasRcvWScale </name>
        <direction> out </direction>
        <relatedStateVariable> HasRcvWScale </relatedStateVariable>
    </argument>
    <argument>
        <name> RcvWScale </name>
        <direction> out </direction>
        <relatedStateVariable> RcvWScale </relatedStateVariable>
    </argument>
    <argument>
        <name> HasTimeStamp </name>
        <direction> out </direction>
        <relatedStateVariable> HasTimeStamp </relatedStateVariable>
    </argument>
    <argument>
        <name> TimeStamp </name>
        <direction> out </direction>
        <relatedStateVariable> TimeStamp </relatedStateVariable>
    </argument>
</argumentList>
</action>
<action>
    <name> Unsubscribe </name>
    <argumentList>
        <argument>
            <name> Uuid </name>
            <direction> in </direction>
            <relatedStateVariable> UUID </relatedStateVariable>
        </argument>
        <argument>
            <name> SubId </name>
            <direction> in </direction>
            <relatedStateVariable> SubID </relatedStateVariable>
        </argument>
    </argumentList>
</action>
</actionList>
<serviceStateTable>
    <stateVariable>
        <name> UUID </name>
        <sendEventsAttribute> no </sendEventsAttribute>
        <dataType> string </dataType>
    </stateVariable>
    <stateVariable>
        <name> Protocol </name>

```

```

    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>string</dataType>
</stateVariable>
<stateVariable>
    <name>Port</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui2</dataType>
</stateVariable>
<stateVariable>
    <name>Address</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>string</dataType>
</stateVariable>
<stateVariable>
    <name>Payload</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>string</dataType>
</stateVariable>
<stateVariable>
    <name>Time</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui2</dataType>
</stateVariable>
<stateVariable>
    <name>Packet</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>string</dataType>
</stateVariable>
<stateVariable>
    <name>Reply</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>string</dataType>
</stateVariable>
<stateVariable>
    <name>TrailerLength</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
    <name>SubId</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
    <name>PeerIP</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>string</dataType>
</stateVariable>
<stateVariable>
    <name>LocalPort</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
    <name>RemotePort</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
</stateVariable>

```

```

<stateVariable>
  <name>Behavior</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>string</dataType>
</stateVariable>
<stateVariable>
  <name>TimeOut</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>WakeOnPush</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>boolean</dataType>
</stateVariable>
<stateVariable>
  <name>MsgTemplate</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>string</dataType>
</stateVariable>
<stateVariable>
  <name>VariableFields</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>string</dataType>
</stateVariable>
<stateVariable>
  <name>Period</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>FirstRun</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>MsgFilter</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>string</dataType>
</stateVariable>
<stateVariable>
  <name>FilterOffset</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>OpaqueFields</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>SessionID</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>SrcIP</name>
  <sendEventsAttribute>no</sendEventsAttribute>

```



```

    <dataType>string</dataType>
</stateVariable>
<stateVariable>
  <name>DstIP</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>string</dataType>
</stateVariable>
<stateVariable>
  <name>SrcPort</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>DstPort</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>Type</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>Proto</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>Family</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>InQLen</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>OutQLen</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>InQSeq</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>InQBuffSize</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>ui4</dataType>
</stateVariable>
<stateVariable>
  <name>InQBuff</name>
  <sendEventsAttribute>no</sendEventsAttribute>
  <dataType>string</dataType>
</stateVariable>
<stateVariable>

```

```

    <name>OutQSeq</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
  </stateVariable>
  <stateVariable>
    <name>OutQBuffSize</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
  </stateVariable>
  <stateVariable>
    <name>OutQBuff</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>string</dataType>
  </stateVariable>
  <stateVariable>
    <name>OptMask</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
  </stateVariable>
  <stateVariable>
    <name>SndWScale</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
  </stateVariable>
  <stateVariable>
    <name>MssClamp</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
  </stateVariable>
  <stateVariable>
    <name>HasRcvWScale</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>boolean</dataType>
  </stateVariable>
  <stateVariable>
    <name>RcvWScale</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
  </stateVariable>
  <stateVariable>
    <name>HasTimeStamp</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>boolean</dataType>
  </stateVariable>
  <stateVariable>
    <name>TimeStamp</name>
    <sendEventsAttribute>no</sendEventsAttribute>
    <dataType>ui4</dataType>
  </stateVariable>
</serviceStateTable>
</scpd>

```

4.2 LP Service XML Description

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<scpd xmlns="urn:schemas-upnp-org:service-1-0">
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <actionList>
    <action>
      <name>GetPowerManagementInfo</name>
      <argumentList>
        <argument>
          <name>WakeUpMethod</name>
          <direction>out</direction>
          <relatedStateVariable> WakeUpMethod </relatedStateVariable>
        </argument>
        <argument>
          <name>PowerSupplyStatus</name>
          <direction>out</direction>
          <relatedStateVariable> PowerSupplyStatus </relatedStateVariable>
        </argument>
      </argumentList>
    </action>
    <action>
      <name> Wakeup </name>
      <argumentList>
        <argument>
          <name>PowerState</name>
          <direction>out</direction>
          <relatedStateVariable> PowerState </relatedStateVariable>
        </argument>
      </argumentList>
    </action>
    <action>
      <name>GoToSleep</name>
      <argumentList>
        <argument>
          <name>RecommendedSleepPeriod</name>
          <direction>in</direction>
          <relatedStateVariable> SleepPeriod </relatedStateVariable>
        </argument>
        <argument>
          <name> RecommendedPowerState</name>
          <direction>in</direction>
          <relatedStateVariable> PowerState </relatedStateVariable>
        </argument>
        <argument>
          <name>SleepPeriod</name>
          <direction>out</direction>
          <relatedStateVariable> SleepPeriod </relatedStateVariable>
        </argument>
        <argument>
          <name> PowerState</name>
          <direction>out</direction>
          <relatedStateVariable> PowerState </relatedStateVariable>
        </argument>
      </argumentList>
    </action>
    <action>
      <name>DefineSleepPeriod</name>

```

```

    <argumentList>
      <argument>
        <name>RecommendedSleepPeriodStart</name>
        <direction>in</direction>
        <relatedStateVariable> SleepPeriodStart </relatedStateVariable>
      </argument>
      <argument>
        <name>RecommendedSleepPeriodEnd</name>
        <direction>in</direction>
        <relatedStateVariable> SleepPeriodEnd </relatedStateVariable>
      </argument>
      <argument>
        <name> RecommendedPowerState</name>
        <direction>in</direction>
        <relatedStateVariable> PowerState </relatedStateVariable>
      </argument>
    </argumentList>
  </action>
</actionList>
<serviceStateTable>
  <stateVariable>
    <name>ExternalPowerSupplySource</name>
    <sendEventsAttribute>yes</sendEventsAttribute>
    <dataType>i4</dataType>
    <defaultValue> 0 </defaultValue>
  </stateVariable>
  <stateVariable>
    <name>BatteryLow</name>
    <sendEventsAttribute>yes</sendEventsAttribute>
    <dataType>boolean</dataType>
    <defaultValue> false </defaultValue>
  </stateVariable>
  <stateVariable>
    <name>PowerSupplyStatus</name>
    <sendEventsAttribute> no </sendEventsAttribute>
    <dataType> string </dataType>
    <defaultValue> active </defaultValue>
  </stateVariable>
  <stateVariable>
    <name> SleepPeriod </name>
    <sendEventsAttribute> no </sendEventsAttribute>
    <dataType>i4</dataType>
    <defaultValue> -1 </defaultValue>
  </stateVariable>
  <stateVariable>
    <name> SleepPeriodStart </name>
    <sendEventsAttribute> yes </sendEventsAttribute>
    <dataType>i4</dataType>
    <defaultValue> -1 </defaultValue>
  </stateVariable>
  <stateVariable>
    <name> SleepPeriodEnd </name>
    <sendEventsAttribute> yes </sendEventsAttribute>
    <dataType>i4</dataType>
    <defaultValue> -1 </defaultValue>
  </stateVariable>
  <stateVariable>
    <name> PowerState sendEvents="yes" </name>

```

```
<sendEventsAttribute> no </sendEventsAttribute>
<dataType> string </dataType>
<defaultValue> 0 </defaultValue>
<allowedValueList>
  <allowedValue> 0 </allowedValue>
  <allowedValue> 1 </allowedValue>
  <allowedValue> 2 </allowedValue>
  <allowedValue> 3 </allowedValue>
  <allowedValue> 4</allowedValue>
</allowedValueList>
</stateVariable>
<stateVariable>
  <name> WakeUpMethod </name>
  <sendEventsAttribute> no </sendEventsAttribute>
  <dataType> string </dataType>
  <defaultValue> WOL </defaultValue>
</stateVariable>
</serviceStateTable>
</scpd>
```