

Application note

homeLYnk and SONOS

Basic integration and control of popular IP music system



Safety Information

Important Information



Read these instructions carefully before trying to install, configure, or operate this software. The following special messages may appear throughout this bulletin or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of either symbol to a “Danger” or “Warning” safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.



This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

DANGER

DANGER indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury.

WARNING

WARNING indicates a potentially hazardous situation which, if not avoided, can result in death or serious injury.

CAUTION

CAUTION indicates a potentially hazardous situation which, if not avoided, can result in minor or moderate injury.

NOTICE


NOTICE is used to address practices not related to physical injury. The safety alert symbol shall not be used with this signal word.

Please Note

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction, installation, and operation of electrical equipment and has received safety training to recognize and avoid the hazards involved.

Safety Precautions

 WARNING
<p>HAZARD OF INCORRECT INFORMATION</p> <ul style="list-style-type: none">• Do not incorrectly configure the software, as this can lead to incorrect reports and/or data results.• Do not base your maintenance or service actions solely on messages and information displayed by the software.• Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable standards and requirements.• Consider the implications of unanticipated transmission delays or failures of communications links. <p>Failure to follow these instructions can result in death, serious injury, or equipment damage.</p>

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information that is contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Schneider Electric.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

© 2014 Schneider Electric. All rights reserved

Table of Contents

- 1 Introduction 6
- 2 Design 7
- 3 Configuration 8
 - 3.1 Configuration of SONOS player..... 8
 - 3.2 Configuration of homeLYnk 11
 - 3.2.1 Creating communication objects..... 11
 - 3.2.2 Creating/copying scripts 11
 - 3.2.3 Creating visualization 15
 - 3.2.4 Creating KNX control script 17
- 4 Conclusion..... 18
- 5 Appendix 18
 - 5.1 Glossary 18

1 Introduction

This application note describes basic connection between your homeLYnk with SONOS IP player(s).

A glossary is available in the appendix chapter of this document. Please refer to it whenever necessary.

Competencies

This document is intended for readers who have been trained on homeLYnk/ spaceLYnk products. The integration should not be attempted by someone who is new to the installation of either products and scripts installation and modifications are necessary. It is also mandatory to have knowledge of SONOS products and SONOS settings including IP network setting as complete SONOS settings are not included in this AN.

System prerequisites

Software	Version	Download
homeLYnk	1.2 and newer	http://www.schneider-electric.com
SONOS controller for PC	5.0	http://www.sonos.com

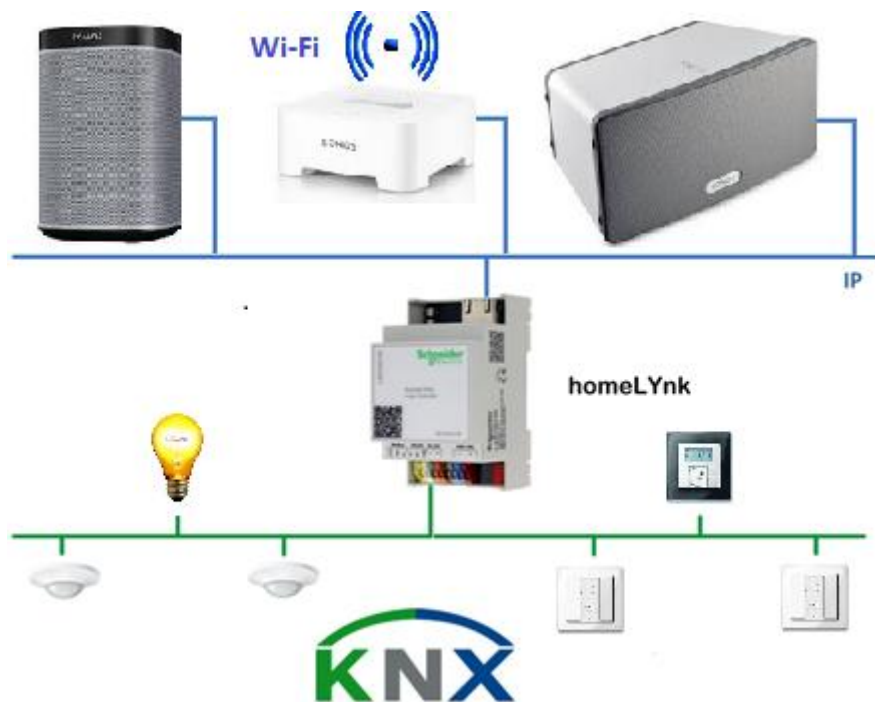
Table 1: System prerequisites

Hardware	Description
SONOS Play:1	Compact wireless speaker
SONOS Play:3	Wireless speaker
SONOS BRIDGE	Multi-room connection for wireless speakers

Table 2: SONOS hardware

2 Design

SONOS is a system of Hi-Fi wireless speakers and audio components. It unites your digital music collection in one app that you control from many devices. Thanks to its IP communication it is able to communicate and be controlled from homeLYnk.



Picture 1: homeLYnk and SONOS connection topology

You can have more than one SONOS components connected in your IP network. Each of them is controlled separately through its IP address.

3 Configuration

3.1 Configuration of SONOS player

You must finish basic configuration of SONOS to be able to control SONOS components from homeLYnk.

Basic configuration of SONOS including:

- 1) Download and installation of SONOS application running on PC and/or tablets mobile phones
- 2) Connect SONOS(s) to IP network (LAN or W-LAN)
- 3) Add SONOS(s) to the player list of bound devices
- 4) Associate music archive with SONOS(s)
- 5) Run SONOS application and create play lists/queues
- 6) Get SONOS(s) IP address for homeLYnk setting

- 1) Free application can be downloaded directly from SONOS web page:

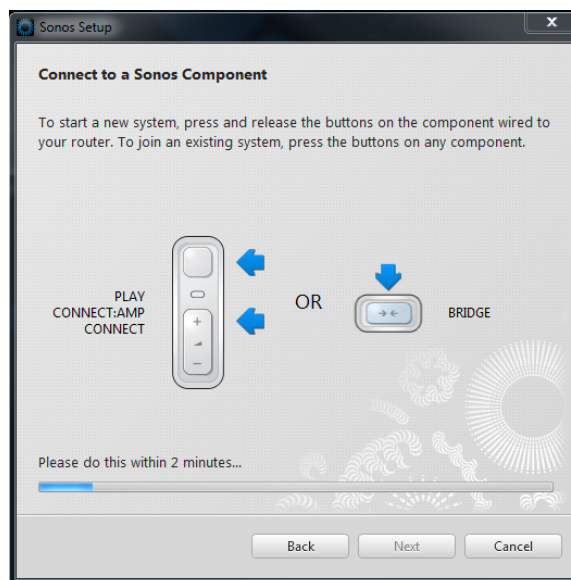
https://sonos.custhelp.com/app/answers/detail/a_id/2425

Install application suitable for device you are using (PC, Android, iOS).

- 2) Connect SONOS(s) to your IP network. Firewall and internet security setting are specified here:

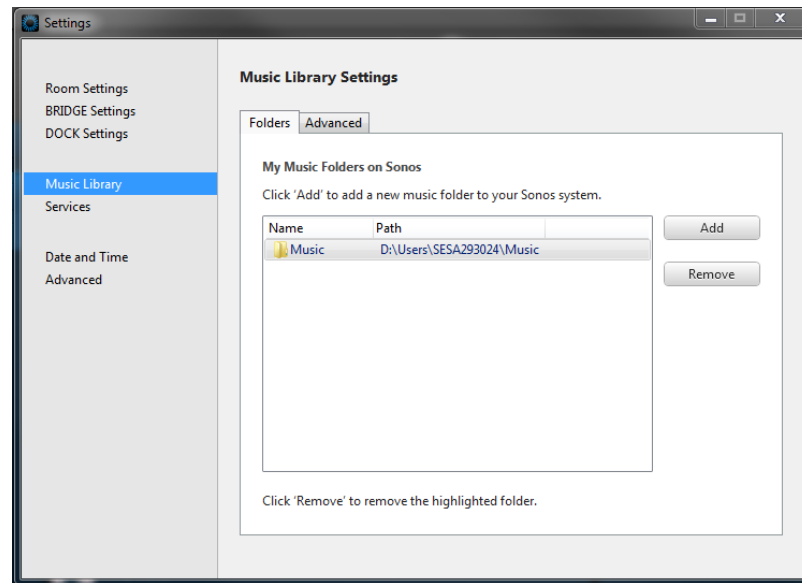
https://sonos.custhelp.com/app/answers/detail/a_id/692

- 3) Add SONOS(s) to the player list of bound devices.



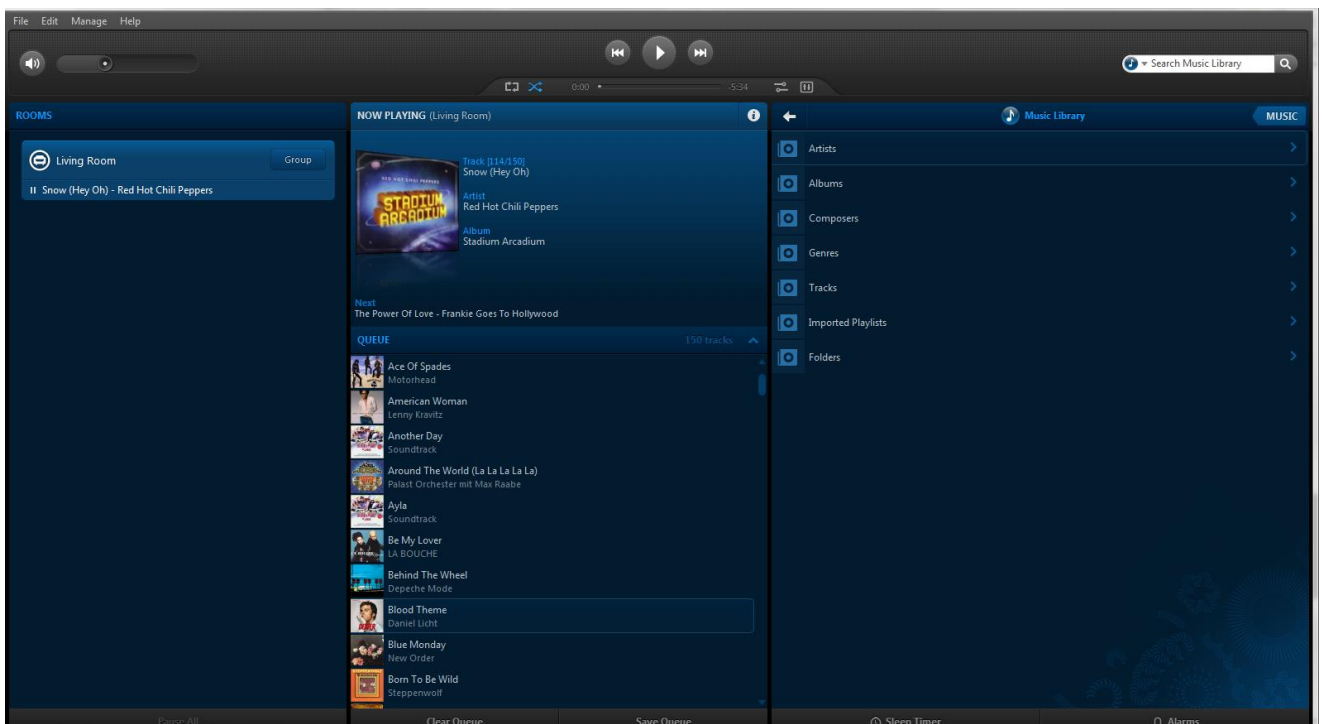
Picture 2 SONOS connection

4) Associate music archive with SONOS(s) in *Manage / Music library* tab :



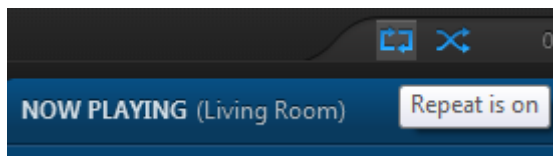
Picture 3 SONOS music library

5) Run SONOS application and create play lists/queues:



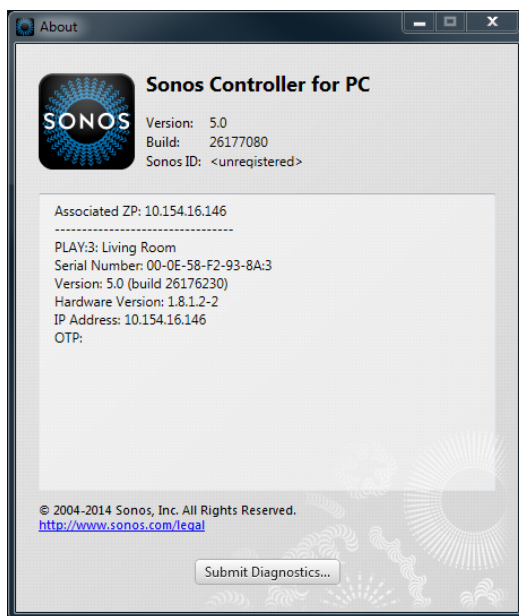
Picture 4 SONOS application interface

Note: homeLYnk can only play lists /queues previously made in SONOS application. If lists/queues are empty or already played, no songs will be played. It is recommended to switch Repeat function ON to prevent reaching the end of playlist.



Picture 5 detail of repeat function button

6) Get SONOS(s) IP address for homeLYnk setting in *Help / About my SONOS system* TAB:



Picture 6 SONOS player About page

For more detailed instructions visit web pages of SONOS section support:

<http://www.sonos.com/support>

3.2 Configuration of homeLYnk

SONOS player is controlled through KNX object specified in control script. Following steps are necessary for control:

- 1) Creating communication Objects
- 2) Creating/copying SONOS control scripts
- 3) Creating visualization (optional)
- 4) Creating KNX control script (optional)

3.2.1 Creating communication objects

Three objects are used in simple version of control.

- a) Track name in 14 byte ASCII format (16.002)
- b) Volume in % scale format (5.001)
- c) Common control in 1 byte format (5.1). This object is controlling all functions by assigned value.

See SONOS control script for details.

It is recommended to use higher group address range for easy recognition and prevention of conflicts.

6/6/x group addresses are used in sample scripts

Note: List of possible further objects e.g. mute/un-mute, artist or album title are pre-made in SONOS feedback script and can be added easily if needed.

3.2.2 Creating/copying scripts

Three scripts are used for SONOS control.

1) SONOS feedback script

SONOS feedback script is binding SONOS and homeLYnk . Set SONOS IP address in the field on line 65 as obtained in step 6 of SONOS configuration. Modify group addresses if needed.

```
--[[
-- TODO: /scada/resources/icons/no_albumart.jpg does not exist, need to be created
-- url for actual albumart (SONOS_IP = e.g. 192.168.0.30, port = 1400 )
--      displays image in original sizes: /scada/vis/sonos-
albumart?id=192.168.0.30_1400
```

```

--      displays image with width set to 175px:           /scada/vis/sonos-
albumart?id=192.168.0.30_1400&width=175
--      displays image with width and height set to 175px: /scada/vis/sonos-
albumart?id=192.168.0.30_1400&height=175&width=175

-- Code preview for automatic SONOS discovery
local _, item
local discovered = Sonos.discovery()
log(discovered)

for _, item in pairs(discovered or {}) do
    log(Sonos{ address = item.address, port = item.port }:info())
end

-- SONOS object creation
local sns = Sonos{
    address = "192.168.0.124",
    -- port = 1400, -- optional, default port is 1400
}

-- useful methods
sns:upnpCmd(cmd, param)
sns:parseXml(s)

sns:play(speed)           -- optional, default speed = 1
sns:stop()
sns:pause()
sns:next()
sns:prev()
sns:getVolume()
sns:setVolume(value)      -- in range <0, 100>
sns:getMute()
sns:setMute(value)        -- true/false
sns:getBass()
sns:setBass(value)        -- in range <-10, 10>
sns:getTreble()
sns:setTreble(value)      -- in range <-10, 10>
sns:getLoudness()
sns:setLoudness(value)    -- true/false
sns:getHeadphoneConnected()
sns:getSupportsOutputFixed()
sns:getOutputFixed()
sns:setOutputFixed(value) -- true/false

sns:info()                -- provides info about sonos useful for visualization
--]]

function knxStore(address, value, write)
    if nil ~= value then
        if write then
            grp.write(address, value)
        else
            grp.update(address, value)
        end
    end
end

```

```

end

local sns = Sonos("10.154.16.142")
local info = sns:info() or {}
log("Sonos:info", info)

if nil ~= next(info) then
    storage.set(sns.storageId, info)

    -- actualize volume
    if info.control.volume then
        -- save to storage to avoid cyclic writing back to SONOS
        storage.set("665_sonos_vol", {val = tonumber(info.control.volume), timestamp =
os.time()})
        knxStore("6/6/5", info.control.volume)
    end
    -- actualize mute
    --knxStore("6/1/0", info.control.mute)

    -- actualize play/pause
    if info.control.state then
        --knxStore("6/1/1", info.control.state == "PLAYING")
    end

    -- actualize time
    -- knxStore("6/1/3", info.status.RelTime)

    local metadata = info.status.TrackMetaData or {}

    -- actualize track
    local title = metadata.title or ""
    knxStore("6/6/4", title .. string.rep(" ", 14 - #title))

    -- actualize artist
    -- knxStore("6/1/5", metadata.albumArtist)

    -- actualize album
    -- knxStore("6/1/6", metadata.album)
end

```

Note: Use all commented commands if needed. Example: If you have DHCP addresses, SONOS can be automatically detected.

2) SONOS volume script for volume control

```
-- fallback to sonos update
local volume = storage.get("665_sonos_vol")

-- log("SONOS volume changed fallback", volume, os.time(), event.getvalue(),
grp.getvalue("6/6/5"))

if not volume or ((volume.timestamp or 0) + 3) < os.time() or (volume.val ~=
event.getvalue()) then
    grp.update("6/6/6", 11)
    storage.set("665_sonos_vol", nil)
end
```

Note: Modify group address if needed.

3) SONOS control for control of all basic functions

Set SONOS IP address on the first line as obtained in step 6 of SONOS configuration. Modify group addresses if needed. Uncomment and use loudness and bass functions if needed.

```
local sns = Sonos("10.154.16.142")
local action = grp.getvalue(event.dst)

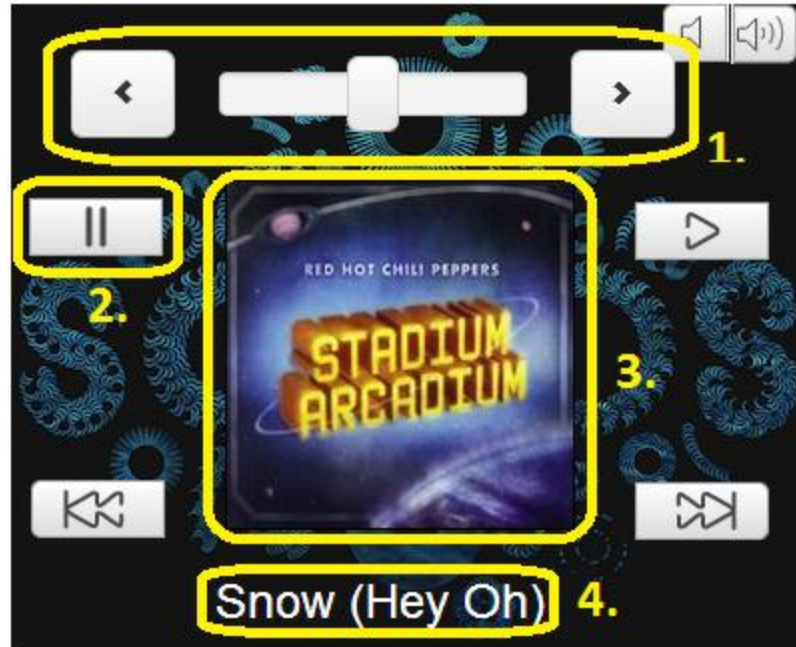
-- log("SONOS control action " .. action)

if 0 == action then          -- stop
    sns:stop()
elseif 1 == action then      -- play
    sns:play()
elseif 2 == action then      -- pause
    sns:pause()
elseif 3 == action then      -- next song
    sns:next()
elseif 4 == action then      -- previous song
    sns:prev()
elseif 5 == action then      -- mute
    sns:setMute(true)
elseif 6 == action then      -- unmute
    sns:setMute(false)
elseif 7 == action then      -- set loudness on
    sns:setLoudness(true)
elseif 8 == action then      -- set loudness off
    sns:setLoudness(false)
--[[ not used yet
elseif 9 == action then      -- set bass (range <-10, 10>)
    sns:setBass(grp.getvalue("x/y/z"))
elseif 10 == action then     -- set treble (range <-10, 10>)
    sns:setTreble(grp.getvalue("x/y/z"))
--]]
elseif 11 == action then     -- set volume (range <0, 100>)
    sns:setVolume(grp.getvalue("6/6/5"))
```

end


3.2.3 Creating visualization

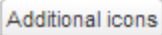
Object specified and used in scripts can be linked with visualization. Widget form is used in sample visualization.



Picture 7 SONOS sample widget

1. Volume control. Slide bar linked to SONOS volume object (6/6/5).
2. Control icons. Pause function is shown as an example. Fixed value "2" is sent when you click on icon. All used icons are included in the Application note folder as Player_Icons .zip

Main object:	6/6/6 SONOS contr 
Status object:	Use main object
Custom name:	Pause
Read-only:	<input type="checkbox"/>
Hide in Smartphone:	<input type="checkbox"/>
Hide background:	<input checked="" type="checkbox"/>
Send fixed value:	2
No bus write:	<input type="checkbox"/> In PC/Tablet/Smartphone
Pin code:	
Widget:	No widget
Display mode:	Icon
Default icon:	PauseOFF.svg
Smartphone icon:	
Additional classes:	
Show control:	<input type="checkbox"/> Inline in PC/Tablet



Picture 8 SONOS object's settings

- Album cover is frame in homeLYnk visualization which is pointing to the picture stored in the music archive

Example:

http://10.154.20.26/scada/vis/sonos-albumart?id=10.154.16.146_1400&height=175&width=175

First IP address is address of the homeLYnk second IP address is address of the SONOS with port 1400, height and width is defining size of the picture.

Source:	Url
Url:	http://10.154.20.26/scada/
Frame size:	200 200
Custom name:	
Hide in Smartphone:	<input type="checkbox"/>
Additional classes:	

Picture 9 frame settings

4. Track name displaying object 6/6/4 which is set in SONOS feedback script.

Note: Track length is limited to 14 ACSII characters due to limit of homeLYnk KNX object. Non-ASCII characters are not displayed.

3.2.4 Creating KNX control script

Short event based script is necessary for control of each SONOS function directly with KNX objects:

```
if event.getvalue() then
  grp.update("6/6/6", 1)
end
```

This script will start SONOS playback, when bonded group is switched ON.

4 Conclusion

Above Application note describe basic connection of homeLYnk and SONOS. homeLYnk is controlling only SONOS players properly set in their controlling software.

5 Appendix

5.1 Glossary

The following table describes the acronyms and defines the specific terms used in this document.

Abreviation	Description
ASCII	American Standard Code for Information Interchange – symbol table
DHCP	Dynamic Host Configuration Protocol

Table 3:glossary

Schneider Electric Industries SAS

Head Office

35, rue Joseph Monier

92506 Rueil-Malmaison Cedex

FRANCE

www.schneider-electric.com