This and That

# Abstract

This document holds several pieces of information concerning the development of 004\_HomeMatic\_EXE.

# Sockets

## Establishing a connection

In order to gain a remote connection to the HomeMatic device, port 80 (HTTP-Server) must be used.

## Create a request

A HTTP request may be the right choice for gaining information from the remote host/server.

### HTTP-Request Format

## IPHostEntry

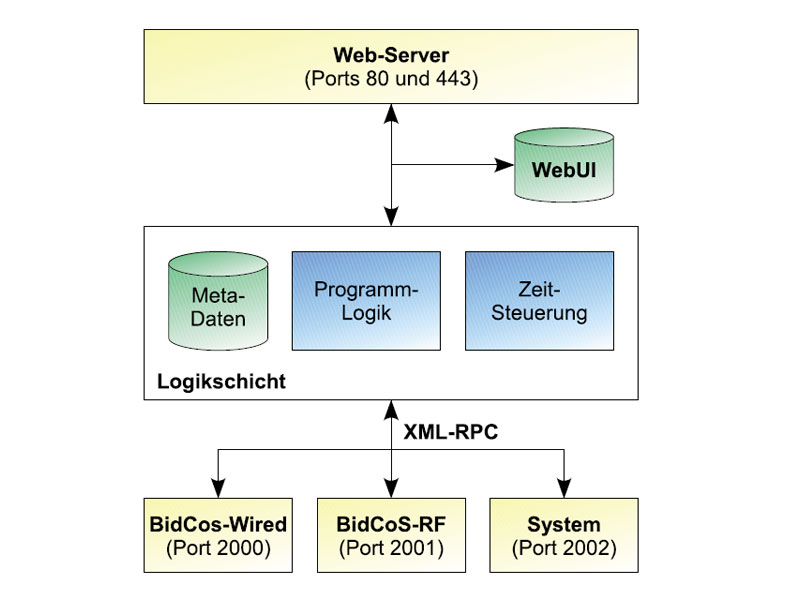
This class associates a DNS hostname with an array of aliases and an array of matching IP addresses.

* AddressList:
  + [0]: IPv6 address
  + [1]: IPv4 address
* Aliases:
  + contains all aliases

# HomeMatic Device

It is possible to gain access to the HomeMatic device via XML-RPC. Therefore, port 80 (HTTP-port) is available for network communication.

The subsequent picture shows the HomeMatic’s architecture:



Further information concerning the HomeMatic device can be seen by using the following link:

https://www.elv.de/Via-Netzwerk-auf-Homematic-zugreifen-–-XML-RPC-SchnittstelleHomematic®/x.aspx/cid\_726/detail\_30789

## Meta-Daten

This part of the logical layer contains data, which are listed below.

* device names
* rooms
* “Gewerke”
* Favourites

## Class DeviceDescription

By reading the current device list from the HomeMatic device an array of DeviceDescription objects is returned. One object identifies a logical device, whereby a logical device is either a physical device, or a channel of a physical device.

* Physical device: JEQ0345797
* Channel:
  + JEQ0345797:0
  + JEQ0345797:1
  + JEQ0345797:2
  + JEQ0345797:3
  + …

A physical device usually consists of several channels. The main channel of a physical device is channel 0 (i.e.: JEQ0345797:0)

### Properties

Class DeviceDescription contains the following properties:

* Address: string (number of ID)
* Type: string

### Channels

A channel usually holds more than just one value. That’s why the XmlRpc method returns an object of type object.

# Criteria of Demands

Any occurring exception must be handled with a try-catch-block!

## Class: HomeMaticHandler

* Attributes
  + …
* Operations
  + Actuator(address : string) : Actuator
  + Url(string) : void
  + Port(int) : void
  + Start() : void
  + KeepAlive() : bool
  + GetAllAdresses() : string[]

## Class: Actuator

* Attributes
  + address : string
  + channel : string
    - concatedAddress : string
  + dataPoint : string
* Operations
  + SetChannel(string) : void
  + SetDataPoint() : void
  + GetValue() : object
  + SetValue(value : object) : void