Automating your home with Home Assistant

Paulus Schoutsen
OpenIoT 2016, San Diego

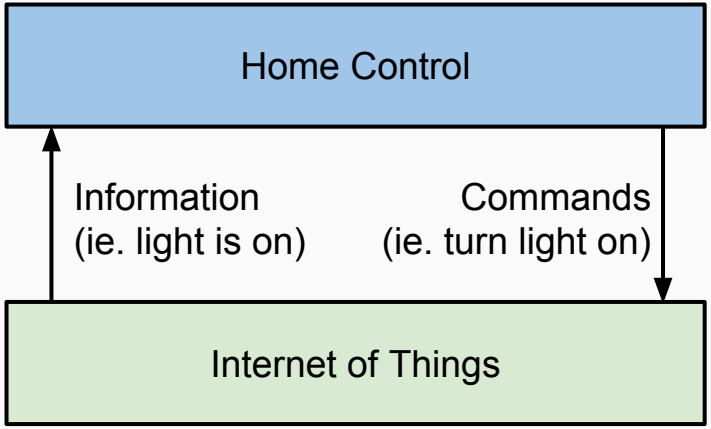
Introduction

- Paulus Schoutsen
- What are we trying to solve
- How is Home Assistant solving it
- Building on top of Home Assistant

Challenges with the Internet of Things

- More and more smart devices in homes without a common standard.
  - Some use WiFi to connect to their own cloud.
  - Some talk one of the many home automation protocols.
- Each device comes with own app.
  - No single place of control.
  - No automation across different products.
- Hubs try to solve this.

What’s the essence of a hub?

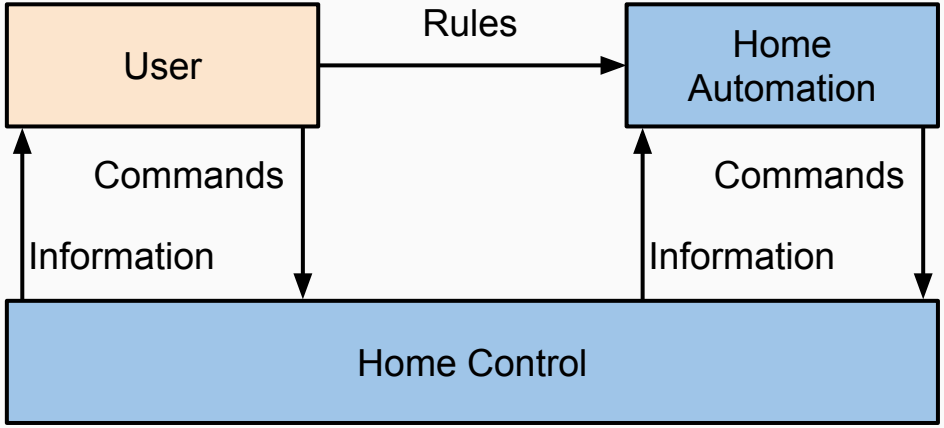


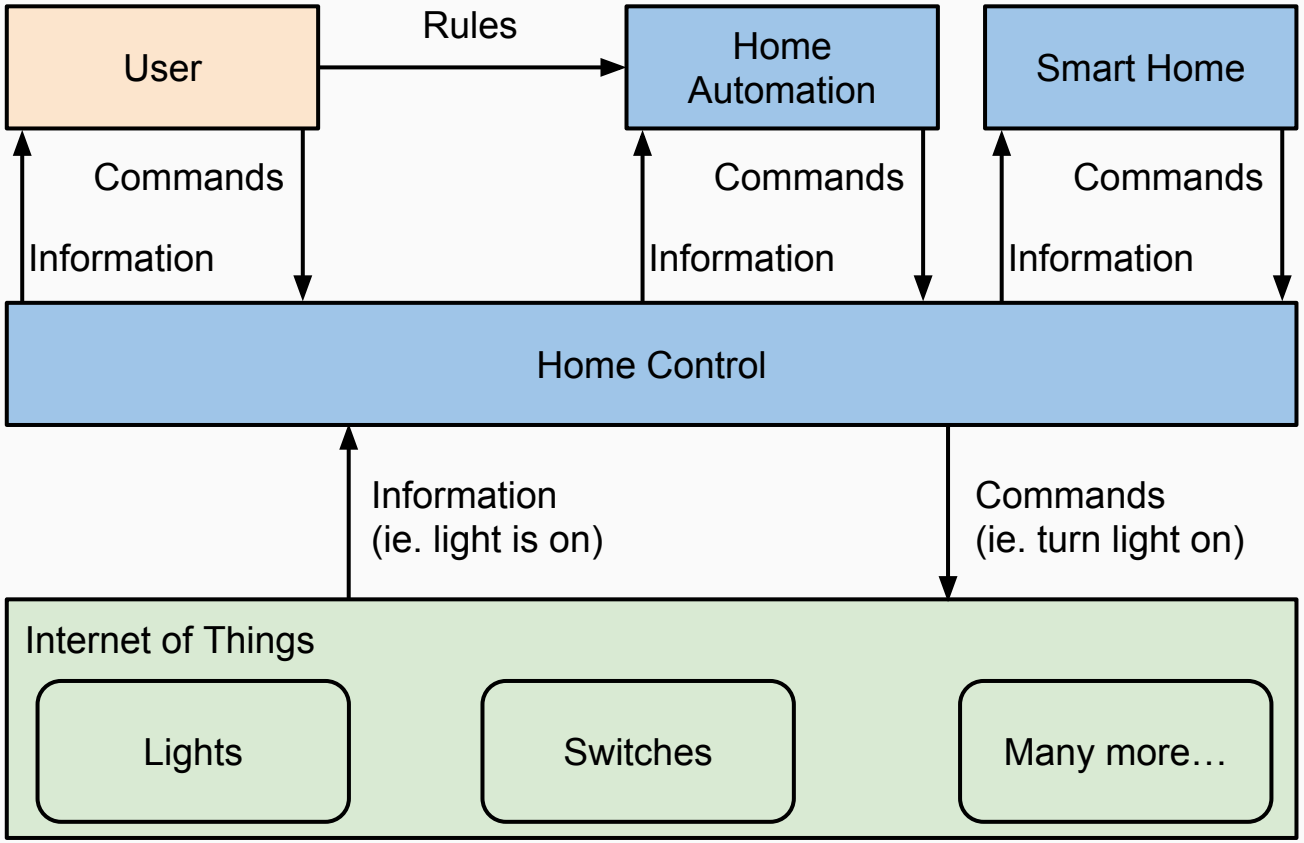
- Devices have a state
  - Light = on
    - Brightness: 120
    - Color: red
  - Media player = playing
    - Volume: 8
    - TV Show: Game of Thrones

- There are events happening:
  - State of light has changed
  - Motion detected

What’s the core of home automation?

- Rule based.
- Triggered by an event.
- Optional conditions based on the current state of devices.
- Action can be to control a device or call an external service.





Different levels of home automation hubs

Home Assistant

Home automation platform running on Python 3.

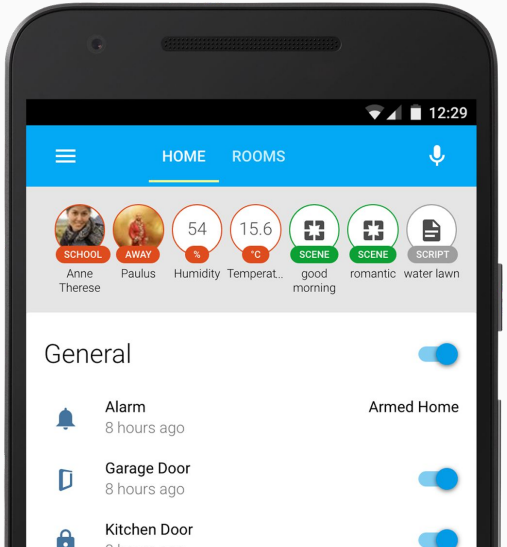
Open-source, MIT license.

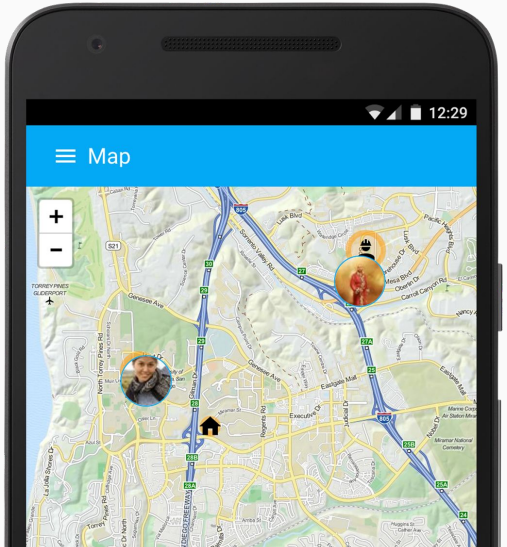
Host it yourself so your data stays yours.

Track people and things on a map.

Responsive web app that runs great everywhere.

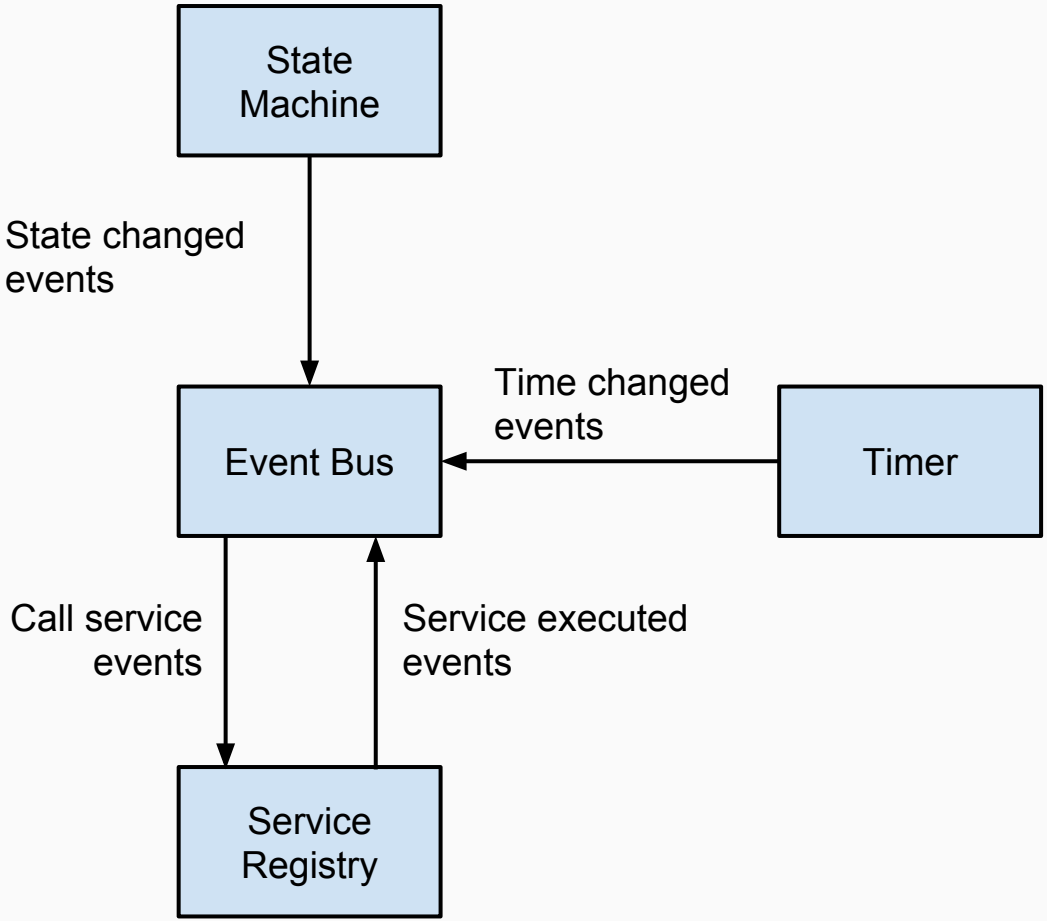
Track the state of your house over time.

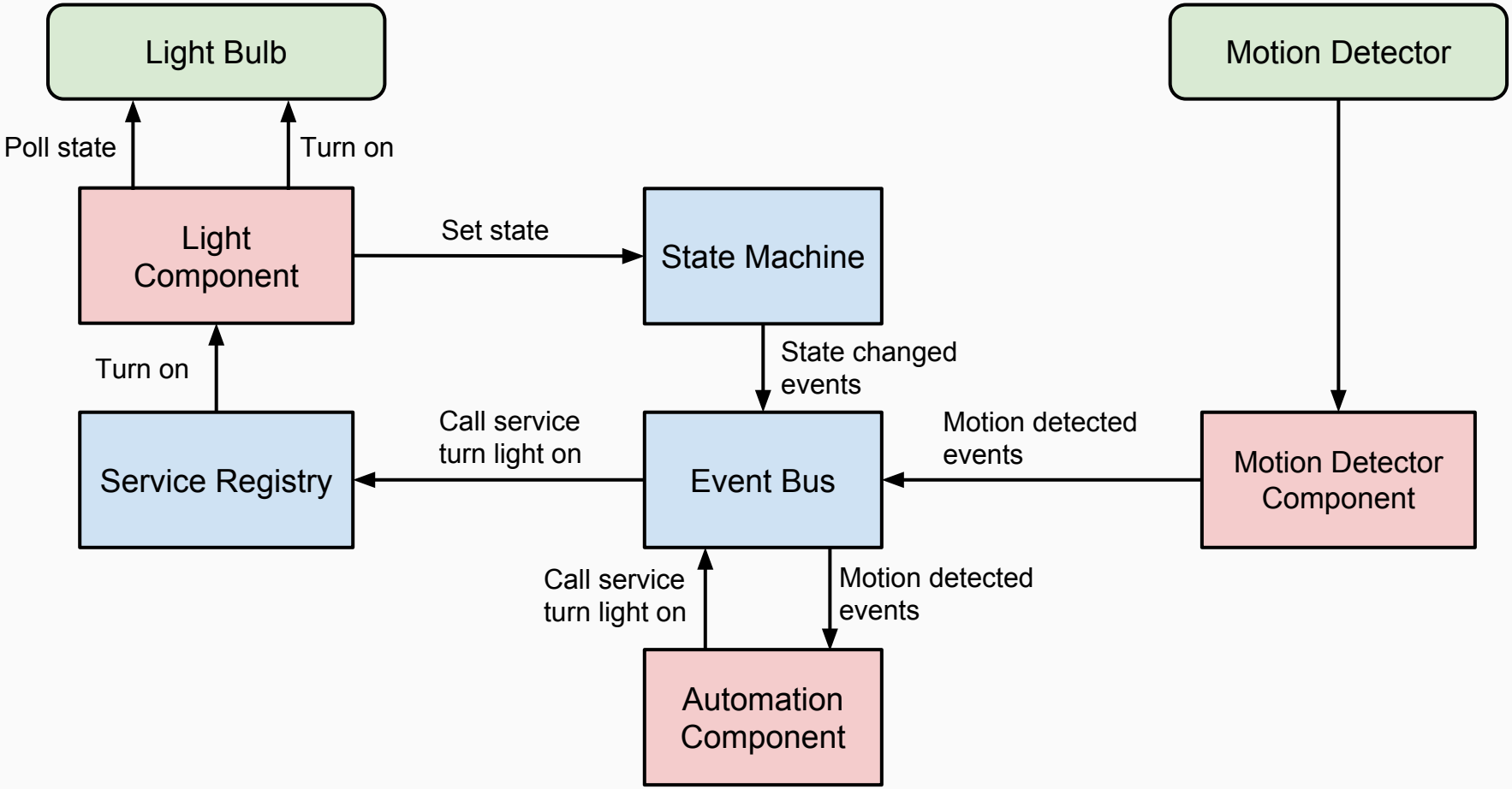




Core Architecture

- Event driven system.
- No notion of lights or switches. Just entities and their state:
  - Identifier
  - State
  - Attributes
  - Last updated (anything)
  - Last changed (state)
- Service Registry to track available services.
- Timer to notify time changed.





Interaction between Home Assistant core, components and the Internet of Things

There’s a component for that.

- Device Types (light, switch, etc)
- Presence Detection
- Entity Organization
- Automation
- Record History
- Third-Party Services Integrations
- Export Data

Find all 242 components and platforms on home-assistant.io/components

Alexa / Amazon Echo Voice	Arduino DIY	Bitcoin Sensor	Forecast.io Weather
Google Cast Media Player	Honeywell Thermostat Thermostat	IFTTT Automation	InfluxDB History
Kodi Media Player	MQTT Hub	MySensors Hub	Nest Hub
Nmap Presence Detection	Owntracks Presence Detection	Philips Hue Light	Plex Media Player
PushBullet Notifications	Sonos Media Player	Wink Hub	

Support for devices in Home Assistant

- An entity component per device type
- Exports abstract base class (ABC) that connects to Home Assistant core.
  - Defines information to be exposed (is_on, rgb_color, brightness)
  - Defines methods to be called (turn_on, turn_off)
- Each protocol/cloud/etc implemented as a platform
  - Expose devices as objects derived from ABC.

Example sensor platform

Create file as:

`<config>/custom_components/sensor/example.py`

In configuration.yaml file:

```
sensor:
  platform: example
```

```python
from homeassistant.const import TEMP_CELCIUS
from homeassistant.helpers.entity import Entity


def setup_platform(hass, config, add_devices, discovery_info=None):
    add_devices([ExampleSensor()])


class ExampleSensor(Entity):
    @property
    def name(self):
        return 'Temperature'

    @property
    def state(self):
        return 23

    @property
    def unit_of_measurement(self):
        return TEMP_CELCIUS
```

For source code, see:
https://gist.github.com/balloob/3e8ae00a2354f4e889c0

Example switch platform

- Based on existence of a file.
- File path passed in via config.

Create file as:

```
<config>/custom_components/
switch/example.py
```

In configuration.yaml file:

```
switch:
  platform: example
  file_path: /tmp/ha_switch
```

```python
import os
from homeassistant.helpers.entity import ToggleEntity


def setup_platform(hass, config, add_devices, discovery_info=None):
    add_devices([FileSwitch(config['file_path'])])


class FileSwitch(ToggleEntity):
    def __init__(self, path):
        self.path = path
        self.update()

    @property
    def name(self):
        return os.path.basename(self.path)

    @property
    def is_on(self):
        return self._state

    def turn_on(self, **kwargs):
        open(self.path, 'a').close()

    def turn_off(self, **kwargs):
        os.remove(self.path)

    def update(self):
        self._state = os.path.isfile(self.path)
```

For source code, see:
https://gist.github.com/balloob/3e8ae00a2354f4e889c0

Components can also provide automation

- Keep two entities in opposite states.

Create file as:

`<config>/custom_components/xor_automation.py`

In configuration.yaml file:

```
xor_automation:
  entity_1: switch.ac
  entity_2: switch.heater
```

```python
from homeassistant.helpers.event import track_state_change
from homeassistant.components import is_on, toggle

DOMAIN = 'xor_automation'


def setup(hass, config):
    entity_1 = config[DOMAIN]['entity_1']
    entity_2 = config[DOMAIN]['entity_2']

    def state_changed(entity_id, old_state, new_state):
        other = entity_2 if entity_id == entity_1 else entity_1

        if is_on(hass, entity_id) == is_on(hass, other):
            toggle(hass, other)

    # Ensure current state is opposite
    if is_on(hass, entity_1) == is_on(hass, entity_2):
        toggle(hass, entity_2)

    track_state_change(hass, [entity_1, entity_2], state_changed)

    return True
```

For source code, see:
https://gist.github.com/balloob/3e8ae00a2354f4e889c0

Example automation configuration

- Turn light on 1 hour before sunset if anyone is home.
- Using automation component.

- 1 or more triggers.
- 0 or more conditions.
  - Time
  - Event
  - State
  - Sun
  - Zone (GPS)
  - MQTT

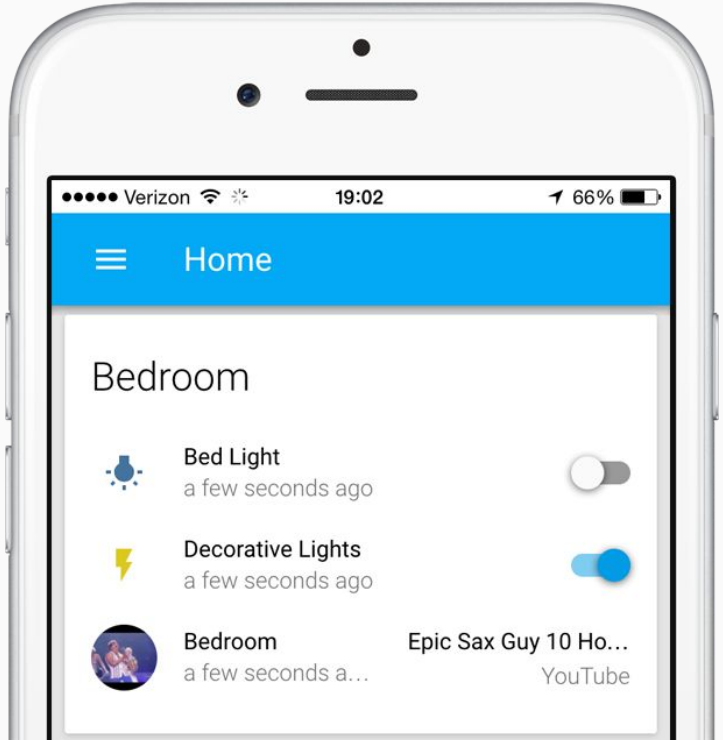
```yaml
automation:
  trigger:
    platform: sun
    event: sunset
    offset: '-01:00:00'
  condition:
    platform: state
    entity_id: group.all_devices
    state: 'home'
  action:
    service: homeassistant.turn_on
    entity_id: group.living_room
```

Thanks for supporting open-source

- GitHub for hosting our source code and website.
- TravisCI for providing continuous integration.
- Coveralls for keeping track of our test coverage.
- Pivotal Tracker for roadmap management.
- Gitter for hosting our community chat.

- AppFolio for allowing me to go to these conferences.
- Home Assistant community for being active, friendly and helpful.

Home Assistant



Try the demo on your phone at home-assistant.io/demo

On your computer with Python 3:

```
pip3 install homeassistant
hass --open-ui
```

Visit our website for more information and tutorials: home-assistant.io