Table 2: Numbers of rooms, devices and apps in each testbed.

Table 3: IoT devices used in the four testbeds, their abbrevia-

Testbed

#Rooms

#Devices

#Smart apps

tion labels, attributes and deployment information.

Abbr.

Device Name

Attributes

Deployment

1

5

23

17

M

2

4

SmartThings

motion

on wall

19

11

Motion Sensor

3

1

6

7

MS

Zooz 4-in-1

motion,

on wall

4

1

6

4

Sensor

illuminance,

humidity

W

SmartThings

water

on bathroom floor

with the real-world devices' states, an alarm is raised report-

Waterleak Sensor

C

SmartThings

contact,

on doors

the

Motion(A)

ing

event

as invalid. Otherwise, the event is

Contact Sensor

acceleration

active

B

SmartThings

button

bedside

accepted and the shadow execution engine changes its simu-

Button

lated motion sensor's state to "active" accordingly. Then, for

L

SmartThings

switch

as ceiling light, lamp

Light Bulb

each accepted event (motion A turns "active" in the example),

PS

SmartThings

presence

in wallet

the shadow execution engine performs the consequential

Arrival sensor

Motion(A)

P

SmarThings

switch, power

to control fan,

checking. It searches all e2e correlations that have Factive

Smart Plug

computer, and lamp

at their anterior places and caches events at their posterior

A

Netatmo

carbonDioxide,

on kitchen

Air Station

sound, humidity

countertop

places in a waiting list. If any event in the list is not received

V

ThreeReality

switch

to control fan

within 60 seconds (consistent with d in hypothesis testing),

Smart Switch

the shadow execution engine reports an anomaly of a missing

Table 4: Automation rules used in Testbed 1.

event. Moreover, an event from a real device also induces an

Index

Smart app rules

event from its derived virtual device (defined in Section 5.3.2)

R1

if the involved condition is true, and the event of the virtual

If M1(active) when Mode(home), then P3(on)

R2

If M2(active) when Mode(home), then P4(on)

device is handled in the same way as that from the real device

R3

If MS1(active), then L1(on) and L2(on)

through contextual and consequential checking.

R4

If MS1(inactive) for 15 minutes, then 1(off) and L2(off)

R5

If MS2(active), then L3(on)

R6

If MS2(inactive) for 10 minutes, then L3(off)

R7

If MS3(active), then L4(on)

6 Evaluation

R8

If MS3(inactive) for 5 minutes, then L4(off)

R9

If MS4(active), then L5(on)

We evaluate HAWatcher with datasets collected from 4 dif-

R10

If MS4(inactive) for 15 minutes, then L5(off)

R11

If B(pressed), then toggle P3 and P4

ferent real-world testbeds as shown in Figure 7. On each

R12

If B(held), then turn off all L and P and Mode(night)

testbed, we spend three weeks collecting dataset for train-

R13

If B(double pressed), turn on P3 and P4 and Mode(home)

R14

ing and one week for testing. We apply collected correla-

If A(CO2 > 950), then P2(on)

R15

If A(CO2 < 950), then P2(off)

tions to each event from the testing datasets to evaluate

R16

If PS1 and PS2 (away), then turn off all L and P and Mode(away)

HAWatcher's performance. We compare HAWatcher with

R17

If PS1 or PS2 (present), then turn on L1, L2, and P1 and Mode(home)

other anomaly detectors. Here, we mainly present evalua-

tion results of Testbed 1. The results of other testbeds are

testbed, we let the resident(s) propose desired automation,

presented in Appendix A.2.

which is fulfilled by us with off-the-shelf IoT devices and

smart apps from the SmartThings official repository. We then

6.1

give them sufficient time to get familiar with the installed home automation before starting data collection.

While there are several existing datasets from smart homes or home activity learning researches, such as [36,37], none of these are collected from appified home testbeds. In addition, these testbeds contain mainly sensor devices but very few actuator devices. These make them unsuitable for evaluating HAWatcher, which is designed to work with appified homes. Next, we describe how we set up our testbeds.

Device Deployment. The device deployment is depicted in Figure 7. We deploy 10 different types of IoT devices as listed in Table 3, including their abbreviation labels. Note that the ThreeReality Smart Switch (denoted as V) can be attached to a wall switch to control traditional devices, such as lights and fans. The smart plug (denoted as P) can be used to control electrical appliances with power plugs; for example, in Testbed 1, P1 and P2 are connected to a TV and a fan, respectively, and P3 and P4 are connected to lamps.

Testbeds and Participants. We deploy SmartThings systems in four homes and Table 2 lists their basic information. Testbeds 1 and 2 each have two residents, and testbeds 3 and 4 have one resident each. The six (6) participants consist of 5 graduate students and 1 undergraduate student including

Automation Rules. We extract automation rules from the installed smart apps in the form of "If trigger when condition,

two females and four males. Two of them are members of

then action". The extracted rules of Testbed 1 are listed in Ta-

our research lab and none paper authors. None of them had

ble 4 (rules of other testbeds are presented in Appendix A.1).

prior experience of using home automation systems. For each

Ethical Concerns and Mitigation. We obtained the IRB