Pers Ubiquit Comput (2013) 17:783–799
DOI 10.1007/s00779-012-0627-x

ORIGINAL ARTICLE

Configuring and controlling ubiquitous computing infrastructure with semantic connections: a tangible and an AR approach

Bram van der Vlist · Gerrit Niezen · Stefan Rapp · Jun Hu · Loe Feijs

Received: 10 January 2012 / Accepted: 21 September 2012

Abstract In the transition from a device-oriented paradigm toward a more task-oriented paradigm with increased interoperability, people are struggling with inappropriate user interfaces, competing standards, technical incompatibilities, and other difficulties. The current handles for users to explore, make, and break connections between devices seem to disappear in overly complex menu structures displayed on small screens. This paper tackles the problem of establishing connections between devices in a smart home environment, by introducing an interaction model that we call *semantic connections*. Two prototypes are demonstrated that introduce both a tangible and an augmented reality approach toward exploring, making, and breaking connections. In the augmented reality approach, connections between real-world objects are visualized by displaying visible lines and icons from a mobile device containing a pico projector. In the tangible approach, objects in the environment are tagged and can be scanned and interacted with, to explore connection possibilities, and manipulate the connections. We discuss the technical implementation of a pilot study setup used to evaluate both our interaction approaches. We conclude the paper with the results of a user study that shows how the interaction approaches influence the *mental models* users construct after interacting with our setup.

Keywords Personal projection · Semantic web · Ontologies · Smart home · Interaction design

1 Introduction

When Weiser [1] wrote his vision of ubiquitous computing about 20 years ago, he postulated that we will be surrounded by networked displays of various sizes, and that we will use them to explore and access our information and computerized infrastructure. They would simply be there, around us, like a piece of scrap paper or a blackboard, their use woven into the fabric of everyday life. It would be easy to switch between actively using them and barely noticing their mere existence. People would concentrate on their everyday activities, unaware that they are using possibly more than a hundred computers within their vicinity to carry out these activities.

In today's reality, although there are rooms accumulating almost comparable amounts of computers in the form of smart phones, web tablets, TV screens, netbooks, personal computers, and so on, we have not yet achieved seamless operation among them. Each and every one of these devices demands our attention, uses a different user interface, and allows access to none of the other components (or only to very few other components within the room). While many of the devices are, or can be networked, the process of making the actual connections and

---

B. van der Vlist (✉) · G. Niezen · J. Hu · L. Feijs
Department of Industrial Design,
Technische Universiteit Eindhoven,
Eindhoven, The Netherlands
e-mail: b.j.j.v.d.vlist@tue.nl

G. Niezen
e-mail: g.niezen@tue.nl

J. Hu
e-mail: j.hu@tue.nl

L. Feijs
e-mail: l.m.g.feijs@tue.nl

S. Rapp
Conante, Fellenoord 130, 5611 ZB Eindhoven,
The Netherlands
e-mail: rapp@conante.com

exchanging the information between them is painful without extensive networking knowledge. Configuration details and connectivity settings are hidden, deeply nested within menu structures. Even with the connections in place, exchanging the actual information is cumbersome and users have to dig into the file structures to find the files to be exchanged. In contrast, from a user's perspective, the devices should be easy to connect since they are physically close to each other (and can thus be touched or pointed at). The information to be shared might have been on the screens moments ago and could form part of the interaction, depending on the user's intention.

Consider a seemingly simple task, like listening to your music stored on your PC or home stereo system from your mobile phone's headphones in the kitchen. It is practically impossible for many users, despite the principal technical ability of the involved devices and available network technologies. Part of the problem may be attributed to the fact that user interfaces are still highly focused on device-oriented operation. Competing standards and technical incompatibilities exist at the service level, contributing to the problem and making it impossible for non-experts to take full advantage of today's technology.

Some of the irritations that users face today are a consequence of the mechanisms of the market that imply different goals for the stakeholders. Developers of devices need to have a strong device-oriented view, whereas users' goals are often more easily resolved within a system-oriented view. Developers are concerned about the functionality and usability of the device at hand, possibly harmonizing its usage over the range of products provided by this specific manufacturer. Users, on the other hand, find themselves with a set of devices and services from different manufacturers, or even different industries. As an example: Users still have to set the integrated clocks of many devices, even if they are all connected to each other. Although a scanner and a printer make up a nice copier, only selected models offer this combined functionality. If you would want to directly print the image that the video camera is currently sampling, you need a PC and install specific software to do so. Seemingly easy tasks (for an unbiased observer) are not possible, because at development time, nobody thought about it and only minimal cross-device capabilities have been implemented.

One possible solution to solving the interoperability problem at the infrastructure level is a software platform developed within the SOFIA[1] (Smart Objects For Intelligent Applications) project. SOFIA is a European research project within the ARTEMIS framework that attempts to make information in the physical world available for smart services—connecting the physical world with the information world.

The software platform developed within SOFIA consists of a common, semantic-oriented store of information and device capabilities, called a Semantic Information Broker (SIB), and various Knowledge Processors (KPs) that interact with one another through the SIB. Rather than promoting the compatibility within one specific service solution in terms of protocols or software stacks, it addresses information-level compatibility and the collaboration between different producers and consumers of information on a more abstract level. The goal is that devices will be able to interact on a semantic level, utilizing (potentially different) existing underlying services or service architectures. Part of this effort is to define a core ontology that describes commonly used concepts, and also model exemplary domains more completely, in a formal ontology that is expressed in RDF (Resource Description Framework).[2]

Ontologies lend themselves well to describing the characteristics of devices, the means to access such devices, and other technical constraints and requirements that affect incorporating a device into a smart environment [2]. Using an ontology also simplifies the process of integrating different interaction approaches, as the different entities and relationships in the SIB can be referred to unambiguously. Because communication via the SIB is standardized, integrating cross-vendor implementations is also simplified and technical incompatibilities can be captured by the ontology that the user can be made aware of.

We aim to enable users to explore and make configurations on a high semantic level without bothering them with low-level details. We believe this can be achieved by making use of semantic web technologies and ontologies in an interoperability platform as proposed by the SOFIA project. Such a platform may be used to support semantic interaction in a smart home environment, as is described in [3].

Building on the SOFIA software platform, we propose a user interaction model and two interface solutions. One user interface solution we propose uses a projected augmented reality approach, based on a concept called Spotlight Navigation [4, 5]. Here, a mobile device containing a pico projector visualizes connection possibilities between devices in the environment. By using direct pointing gestures with the device in the user's hand, users can intuitively explore and manipulate the virtual network connections as if they are part of the user's real-world environment. The second user interface solution is a tangible interaction approach, enabling users to physically select devices in their environment and directly view and manipulate the connections in a simple, universal way [6]. We discuss the implementation details of a pilot study setup, which we used to evaluate both our

---

[1] http://www.sofia-project.eu/.

[2] As used in semantic web technologies, and in the construction of natural language user interfaces or speech dialogue systems.

interaction approaches. We conclude the paper with the results of a user study that shows how the interaction approaches influence the *mental models* users construct after interacting with our setup.

2 Related work

The past decade has resulted in many proposals for configuring ubiquitous computing infrastructure and solving the interconnection and interoperability issues with consumer electronics. Early work by Siio et al. [7] explored interactions with connected devices by moving a stylus along paths on a printed map of the infrastructure that is annotated with barcodes. More recent related work presents solutions for simplifying configuration tasks of in-home networks by creating virtual "wires" between physical objects like memory cards [8] that can interconnect devices.

Others propose to introduce tags, tokens, and containers [9, 10] for tangible information exchange. Concepts like "pick-and-drop" [11] and "select-and-point" [12] are used to manage connections and data exchange between computers and networked devices. The introduction of near-field communication, that is, using a near-field channel like radio-frequency identification or infrared communication, allows for direct manipulation of wireless network connections by means of *proximal interactions* [13].

Fitzmaurice [14] states that we should browse, interact, and manipulate electronic information within the context and situation in which the information originated and where it holds strong meaning. He adopted the notion of *situated information spaces* to avoid being overwhelmed by too much information. This means that the information associated with physical objects should be collected, associated, and collocated with those objects. The physical object anchors the information and provides a logical means of partitioning and organizing the associated information.

Rukzio and Holleis [15] discuss the design space of interactions and applications enabled by pico projector units integrated into mobile devices. Three interaction concepts that are directly related to our prototype are as follows: changing the location and orientation of the projecting device; projection acting as a magic lens, revealing a part of a virtual information layers which is much bigger than the actual projection (magic lens metaphor); and projection showing information related to the object on which the projector currently focuses (augmented reality).

Ballagas et al. [16] surveyed interaction techniques that use mobile phones as input devices to ubiquitous computing devices. The survey is based on Foley's taxonomy [17], where input devices are structured according to the graphic subtasks they can perform: position, orient, select, path, quantify, and text entry. During our design phase, we considered and implemented a number of these subtasks for establishing and dismantling connections.

Three physical mobile interaction techniques were evaluated in [18]. *Touching* refers to bringing the user's mobile device into contact with the object the user wishes to interact with. Using *pointing*, it is possible to select a smart object with a mobile device by aiming at it. *Scanning* allows users to get a list of nearby smart objects by using a wireless mechanism. We incorporated variations of the first two techniques in our prototypes.

In [19], an interaction technique called *stitching* is described that allows users to combine pen-operated mobile devices utilizing wireless networking, by using pen gestures that span multiple displays. For example, a user can transfer photos from one device to another by drawing a path from the photo collection on one device to another device, skipping the bezels of the screens. An advantage of this technique is that since there is natural order implied by the gesture, stitching establishes which device is the sender and which is the receiver. In our augmented reality and tangible UI approach, we made use of implied natural order in a similar manner.

A projector-camera system was used to augment smart objects with additional information by Molyneaux et al. [20]. The system is capable of locating and tracking the objects and projecting information onto the objects themselves, by aligning the projection with the object's surfaces.

Beardsley et al. [21] developed a handheld projection system that lets users create opportunistic displays on any suitable nearby surface. Fiducial markers and a camera are used to calculate and transform the projected image plane. They also described the interaction technique of selecting a physical region of interest, based on the "hold-and-drag" motion of a projected cursor.

Another line of research investigates how to combine functionalities of several devices or services in a fully automatic way, to solve explicitly stated user goals without involving the user in the combination [22, 23]. In contrast to these efforts, the Speakeasy project [24] follows the recombinant computing approach and allows end users to specify *connections* between entities to transfer data. In direct connection mode, users can discover, control, and connect any available component on the network. Recombinant computing is a system design philosophy that follows a bottom–up approach to creating computing environments, where individual entities form part of an elastic, always-changing whole. These entities are designed to be able to interact and interoperate with one another *without having prior knowledge of one another*. They expose simple programmatic interfaces called recombinant interfaces, which govern how they can be made to interoperate.

3 Software platform and semantic modelling

The SOFIA software platform, which forms the backbone of our contribution, utilizes the blackboard architectural pattern to share information between smart devices, rather than have the devices explicitly send messages to one another. When this information is also stored according to some ontological representation, it becomes possible to share information between devices that do not share the same representation model, using the semantics of that information [25].

Ontologies are used to enable the exchange of information without requiring a priori standardization. The first core component of the SOFIA platform is called Smart-M3 and an open source implementation is available online.[3] A notable feature of the SOFIA platform is the capability to subscribe to the changes of data (stored as triples) in the data store and be notified every time these triples are updated, added, or removed.

Smart-M3 takes the blackboard and publish/subscribe concepts and reimplements them in a lightweight manner suitable for small, mobile devices called KPs. These KPs can operate autonomously and anonymously by sharing information through an information store. The SIB is the information store of the smart space and contains the blackboard, ontologies and required service interfaces for the KPs. Figure 1 shows a simplified overview of the Smart-M3 infrastructure.

A DL-based (Description Logic) ontology was created in OWL, the Web Ontology Language. In the ontology, all user interaction within the system is described in terms of interaction events. To enable our semantic connections interaction model (introduced in more detail after this section), the connections between the devices need to be modelled. A `connectedTo` relationship can be added or removed between two existing devices in the ontology. It should be noted that this relationship is both *symmetric* and *irreflexive*.[4]

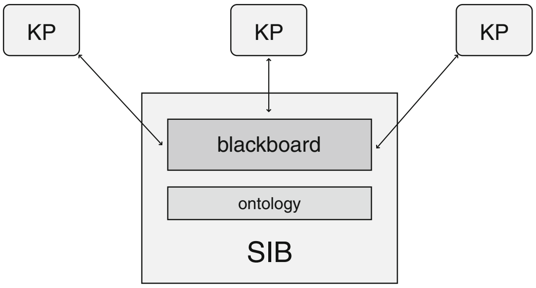

Fig. 1 Smart-M3 infrastructure model, showing the interaction between the main components: Knowledge Processors (KPs) and the Semantic Interaction Broker (SIB)

4 Semantic connections

We defined the term *semantic connections* [26] to refer to meaningful connections and relationships between entities in an ubiquitous computing environment. These connections are invisible by default, but can be viewed and manipulated on demand, using a special-purpose device or application. Semantic connections make up a structural layer of inter-entity relationships on top of the network architecture. The connections can be the real, physical connections (e.g., wired or wireless connections that exist between devices), or conceptual connections that seem to be there from a user's perspective. Their context (what things they connect) is pivotal for their meaning.

The term "semantics" refers to the meaningfulness of the connections. We consider the type of connection, which often has the emphasis now (e.g., WiFi, Bluetooth or USB) not to be the most relevant, but that what the connection can do for someone—its functionality—even more. They have informative properties, that is, they are perceivable in the physical world and have sensory qualities that inform users about their properties and uses. However, these physical qualities might be hidden at some times, or only accessed on demand, by a special-purpose interaction device.

The digital counterparts of semantic connections are modelled in an ontology. There may be very direct mappings, for example, a connection between two real-world entities may be modelled by a `connectedTo` relationship between the representations of these entities in an ontology. Semantic connections exist in both the physical and the digital world and can exist between objects, people, and places. Not only objects and devices have meaning in a system of networked devices—according to [27], physical location within the home and device ownership (or usage) are of central importance for understanding and describing home networks by users. The be more precise, we consider the following entities:

- artifacts;
- smart objects;
- sensors;
- UI elements;
- places;

---

[3] Available from http://sourceforge.net/projects/smart-m3/.

[4] A symmetric property is its own inverse, which means that if we indicate a `connectedTo` relationship from device A to device B, device B will also have a `connectedTo` relationship to device A. An irreflexive property is a property that never relates an individual to itself. This allows us to restrict our model by not allowing a `connectedTo` relationship from a device to itself.

- smart spaces;
- persons.

Semantic connections have properties like directionality, transitivity, and modality (i.e., what things they carry). The rationale behind semantic connections is to rely on:

- the meaning of existing objects to provide meaning for the relationships between the objects and the resulting meaning of the *networked* objects.
- the power of natural mapping and locality, using real objects and locations to provide meaning for the connections that are created between the objects and (object) locations.
- inherent, augmented, and functional feedback and feedforward (using terminology as proposed in [28]) to strengthen the meaning of the connections and the emerging functionality.

Crucial to our approach is to make the gap between user goal and action smaller. If we consider streaming music from one device to another, "streaming" now consists of multiple actions that do not necessarily make sense. In our view, this single high-level goal should have one (or at least as few as possible) single high-level action(s). That single action should carry the meaning of its goal. By using the physical world as interaction space and using the real location of the objects, we are reducing the need to identify the devices from a list with names or rely on other forms of representation.

5 The Connector: a tangible approach

In our previous work [29, 30], a tangible user interface (TUI) approach was introduced. The *Interaction Tile* described in the paper was used as an exploratory prototype to explore our notion of semantic connections. In this tangible approach, smart devices in the home are represented by cubes (that are used in combination with an Interaction Tile). The Interaction Tile shows the connection possibilities with a high level of semantic abstraction, hiding the complexities of wireless network technology. By interacting with the Interaction Tile and cubes, semantic connections can be built, redirected, cut, or bypassed.

We developed a new smart object to interact with the semantic connections, which we called the *Connector*. The Connector has similar functionality to the Interaction Tile and the Spotlight Navigation device, which will be described in the next section. The Connector can be used to explore and manipulate semantic connections between different devices in the home environment. It is a handheld device that identifies devices, by scanning RFID tags that are located on the devices themselves. By holding the Connector on top of the tag, users can explore the connection possibilities that are visualized with lights on top of the Connector. After holding the device in the RFID field for a moment, the device-ID is locked and the other device to be connected can be selected in a similar fashion. With a push-to-click action, a connection between two devices can be established. For removing an existing connection, the ring on the lower part of the device should be pulled until it clicks.

5.1 Design

The cylindrical shape of the Connector (Fig. 2) is loosely inspired on that of a loupe or hand lens. By moving the Connector over a tag, the connection possibilities can be "read" from the top of the cylinder. The display consists of two rings (made up of LEDs), each divided into 4 segments. The Connector supports several actions. You can move it over an object or tag to see whether it is active. A device or object can be selected by holding the Connector close to or on a tag until the selection sequence is completed. The Connector can be compressed by pushing the top and the lower part together, and it can be pulled, by pulling the lower part and the top part away from one another until it clicks.

When the tag is in the range of the Connector's RFID field, it reads the tag and the first (yellow) light segment on top of the Connector will light up, serving as feedback that the Connector recognizes the device. After holding the Connector over a device tag for a moment, a sequence starts, lighting up the second, third, and fourth segment of the inner ring. This can be seen as feedforward to hold the Connector over the tag until it has been selected and all four segments are lit. After the device is recognized and selected, another device may be selected in a similar

Fig. 2 The Connector prototype and a smart phone used as a media player

Fig. 3 Image showing the Connector scanning a colored lighting lamp

fashion. Now, the second ring of lights will start lighting up in sequence and one should wait until both rings are fully lit. Removing the Connector from the tag prematurely cancels the selection process. Figure 3 shows someone scanning a colored lighting lamp with the Connector.

When a connection between the selected devices is possible, both rings start flashing green. When no connection is possible, they will turn red. When a connection between the devices you scanned already exists, the rings will turn green. To make the connection, the Connector is compressed by pushing the top and lower part together, or by pushing the Connector down on the device it is touching, until it clicks. To remove an existing connection between two scanned devices, the ring on the lower part of the Connector should be pulled until it clicks. The rings will show a red light to indicate that the connection has been broken. The segments will turn off once the Connector is moved away from the device. Performing the opposite action of what is required to make or break a connection, cancels the procedure.

5.2 Prototype

The Connector prototype is made out of four separate 3D-printed pieces. The lower part and the top part of the Connector can be moved inward and outward serving as a two-way spring-loaded switch. The prototype packages all the necessary components into one integrated device which is wirelessly connected to a computer using a Bluetooth connection.

The Connector contains the following main components:

- Arduino Stamp 02
- Innovations ID-12 125 kHz RFID reader
- SparkFun BluetoothMateGold
- 8 bi-color LEDs
- Switches
- 3.3 V LiPo battery (850 mAh)

6 Spotlight Navigation: an augmented reality approach

Spotlight Navigation can be used to explore and manipulate connections between smart devices. With Spotlight Navigation, connection information contained in the smart space is projected into the real world, augmenting the real environment with virtual information, making it intuitively perceivable for users. Spotlight Navigation projects icons close to the actual devices in physical space. It allows for the creation of new connections simply by drawing lines between these icons, using a “pick-and-drop” action with a push button on the prototype (press and hold the button when pointing at one device, move over the second device, and release the button). Additionally, the connection possibilities are projected between devices that allow for a connection, by changing the color of the projected line (while the connection is being drawn) from yellow to green when the line’s end is moved over the frame of the targeted device. When a connection is impossible, the connecting line will turn red and disappears as soon as the button is released.

6.1 Design

Spotlight Navigation was invented as an intuitive way of accessing large data spaces through handheld digital projection devices. Rather than directly projecting the equivalent of a small LCD display, Spotlight Navigation continuously projects a small portion of a much larger virtual pane or data space. It is the device’s orientation that defines which part of the larger pane is selected for display. This is done in such a way that the virtual data appears to have a fixed location in the real world. By moving the projector’s light spot over the wall, users make portions of the data space visible through intuitive, direct pointing gestures. This intuitiveness stems from the fact that the projected content always stays roughly at the same physical place, regardless of the orientation of the device. It becomes visible depending on whether it is in the projector’s light cone or not. In other words, users have the impression that they are illuminating a part of a conceptually unbounded virtual data space, just as if they would be looking at hieroglyphs on a huge wall in a tomb with a flashlight. As people are familiar with operating flashlights, the operation needs no or little training. When accessing a data space with the device, users can zoom in and out of the data space by using a scroll wheel control, resulting in a pan-and-zoom user interface.

To visualize the semantic connections in physical space, we rely on the symbolic meaning of color, where green color means “proceed” and red means the opposite. Using green, yellow, and red lines, we aim at referring to the “existence” of a connection, the “possibility” of a connection, or to indicate that a connection is not possible. Figure 4 shows the projection when connecting two devices together.

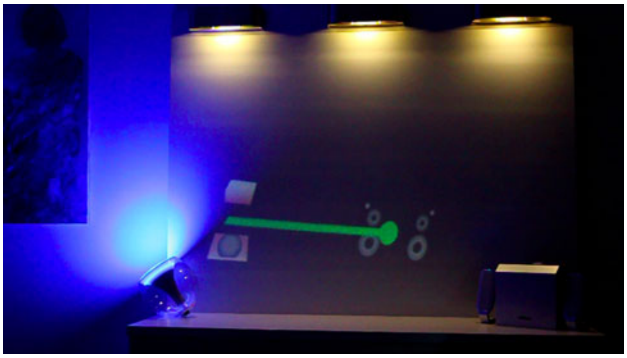

Fig. 4 Example of projected display when two devices are connected together

Fig. 5 Spotlight Navigation prototype

With Spotlight Navigation, devices are identified by their physical location, relying strongly on *natural mapping*. Connections are created simply by drawing lines between the devices. An erasing gesture with the Spotlight Navigation device pointed at an existing connection, breaks the connection.

Besides giving access to a large visual information space from a portable handheld device, Spotlight Navigation is also useful in projected augmented reality. Augmented reality (AR) is frequently associated with the use of head-mounted displays (HMDs) that users must wear, through which they see both their real-world environment and superimposed virtual information that augments the real world [31]. More recently, smart phones have been used to reach similar results. By using the built-in camera and IMU, augmented information can be shown on the phone's display, together with the raw camera image. However, the augmentation of the real world with virtual information can also be achieved by means of projection, without the use of an HMD or camera phone. Here, virtual content is directly projected onto the real world by using a digital projector. With Spotlight Navigation's flashlight metaphor and orientation measurement, such projected AR becomes portable. Due to their physical characteristics, projected AR and HMD-based AR are different in their ability to work in different light conditions, as well as in their ability to augment information on strongly textured or uneven surfaces. A big advantage of using Spotlight Navigation over using HMDs however is that the device is easy to use on an occasional ad hoc basis, as it is easier to pick up and lay down the portable projector than it is to put on and take off the HMD.

6.2 Prototype

On a technical level, the operation is achieved through continuously measuring the orientation, and optionally also the position, of the device. Our prototype is using an inertial navigation module, also called an inertial measurement unit (IMU) that directly measure the orientation by means of accelerometers, gyroscopes, and an electronic compass.

The Spotlight Navigation prototype is a fully embedded setup integrated into a 3D-printed casing. The design of the casing was targeted at getting the smallest possible setup that could run on the integrated batteries. A dummy ring was added to the prototype to strengthen the semantics of a mobile projector. Figure 5 shows the prototype. Our current setup consists of the following components:

- OMAP3530 board (IGEP module)
- Pico projector (Microvision SHOWWX)
- Orientation sensor (Sparkfun 9DOF Razor IMU)
- scroll wheel (with button press functionality)
- two additional buttons
- two 3.7 V Li-ion batteries (Nokia BL5J)

The OMAP3530 processor contains a 3D graphics core (PowerVR) that is capable of rendering the connection visualizations and device icons in real time. Our current prototype still requires the object positions to be manually configured in space, as it did not contain a camera. By using a camera, as is planned for future versions, our intention is to recognize the identity and physical location of each device, so that it is no longer necessary to align the projected object icon with the location of its associated device. The visualization software is derived from the original PC version of Spotlight Navigation and uses OpenGL ES 2.0 to interface with the graphics hardware. An advantage of the laser-based SHOWWX projector over DLP/LED-based projectors is that there is no need to re-adjust the focus for different projection distances.

7 Hardware infrastructure and software components

We now describe the hardware and software used to create a prototype system that implements the interaction model and two interaction approaches described in previous sections. The prototype system consists of many different hardware and software components, built on top of the SOFIA interoperability platform (IOP). The SOFIA IOP is based on a blackboard architectural model (as explained in Sect. 3) that implements the ideas of space-based computing [32]. It consists of two main components: a SIB that acts as a common, semantic-oriented store of information and device capabilities, and KPs, virtual and physical smart objects that interact with one another through the SIB. Various SIB implementations exist that conform to the M3 specification. Smart-M3, the first open source reference implementation released in 2009, was already mentioned. The SIB implementation used in the prototype system is called ADK-SIB (Application Development Kit SIB) and was developed within the SOFIA project.

The ADK-SIB is a Jena-based[5] SIB written in Java and runs on the OSGi (Open Services Gateway initiative) framework. Some modifications were made to the standard ADK-SIB provided by the SOFIA project, such as reasoning support added using the TopBraid SPIN API 1.2.0.[6] Reasoning on information contained within the SIB was performed using SPIN[7] (SPARQL Inferencing Notation).

The KPs, which are software components running on the various devices, communicate with the SIB through SSAP (Smart Space Access Protocol) messages [32] over TCP/IP. SSAP consists of a number of operations to insert, update, and subscribe to information in the SIB. These operations are encoded using XML.

The setup was built in an environment that approximates a real-world home environment for these kinds of devices. Two wireless routers were placed in two different rooms, bridged with an ethernet network cable. One router was configured to act as a DHCP server, while the other acted as a network bridge. All components were connected to the network via 802.11g wireless. In Fig. 6, a schematic overview of the hardware and software components in the pilot study setup is given.

A *Connector KP* was developed to control the Connector device over the Bluetooth interface, while a *Spotlight Navigation KP* controls the Spotlight Navigation device. The Connector was placed in a downstairs room, together with wall-wash lighting (controlled by a *Non-Functional Lighting (NFL) KP*), a presence sensor (controlled by a *Presence KP*), a mobile phone with media capabilities (controlled by a *Music Player KP*), and a colored lighting lamp (controlled by *Lighting Device (LD) KP 1*). The Spotlight Navigation device was placed in an upstairs room, together with another colored lighting lamp (controlled by *LD KP 2*) and a Functional Lighting device with presence sensor (controlled by a *Functional Lighting KP*). A virtual *Sound/Light Transformer (SLT) KP* was developed to transform the audio signals produced by the mobile phone into lighting information that could be rendered by the colored lighting lamp. The system specifications of each component used in the prototype system is shown in Table 1.

8 Pilot evaluation

The prototypes based on our two interaction approaches were evaluated in a pilot user study using the prototype system described in the previous section. This pilot was composed of demonstrators made by the different partners in the SOFIA project and was conducted with users in a setting that resembles a real home.

In order to get enough insights to improve the system, seven groups consisting of three people each were asked to interact with the system, during which their experiences were recorded. The two interaction prototypes presented in this article were part of a larger test setup which was evaluated during a full week of experiments. In addition to the measurement tools that are described in Sect. 8.3, we also performed several performance measurements of the software architecture and held a questionnaire among the developers that worked on the Smart Home pilot. The performance measurements show, among other things, that the software architecture is fast enough to respond to a user's inquiry request (e.g., checking whether a certain connection is possible or already exists, or other user actions). The response time was measured to be well within the maximum allowable value of 2 s, as is defined in [33]. A full description of the performance measurements can be found in [34]. In this paper, we focus on the results which are relevant for evaluating the interaction concepts.

During the pilot, users experienced a smart space with various automated and interactive appliances and devices (which we refer to as smart objects). The appliances in the smart space are interoperable, sensitive to changes in their environment and exchange information with one another. There exists several explicit and implicit relationships between the smart objects, of which some can be explicitly viewed or manipulated with the Spotlight Navigation device (available in the study of the pilot setup upstairs) or the Connector device (available in the living room of the pilot setup downstairs).

8.1 Participants

Twenty-one participants were recruited in seven groups of three friends. Selection was based on age (between 20 and

---

[5] http://jena.sourceforge.net/.

[6] http://topbraid.org/spin/api/.

[7] http://www.spinrdf.org.

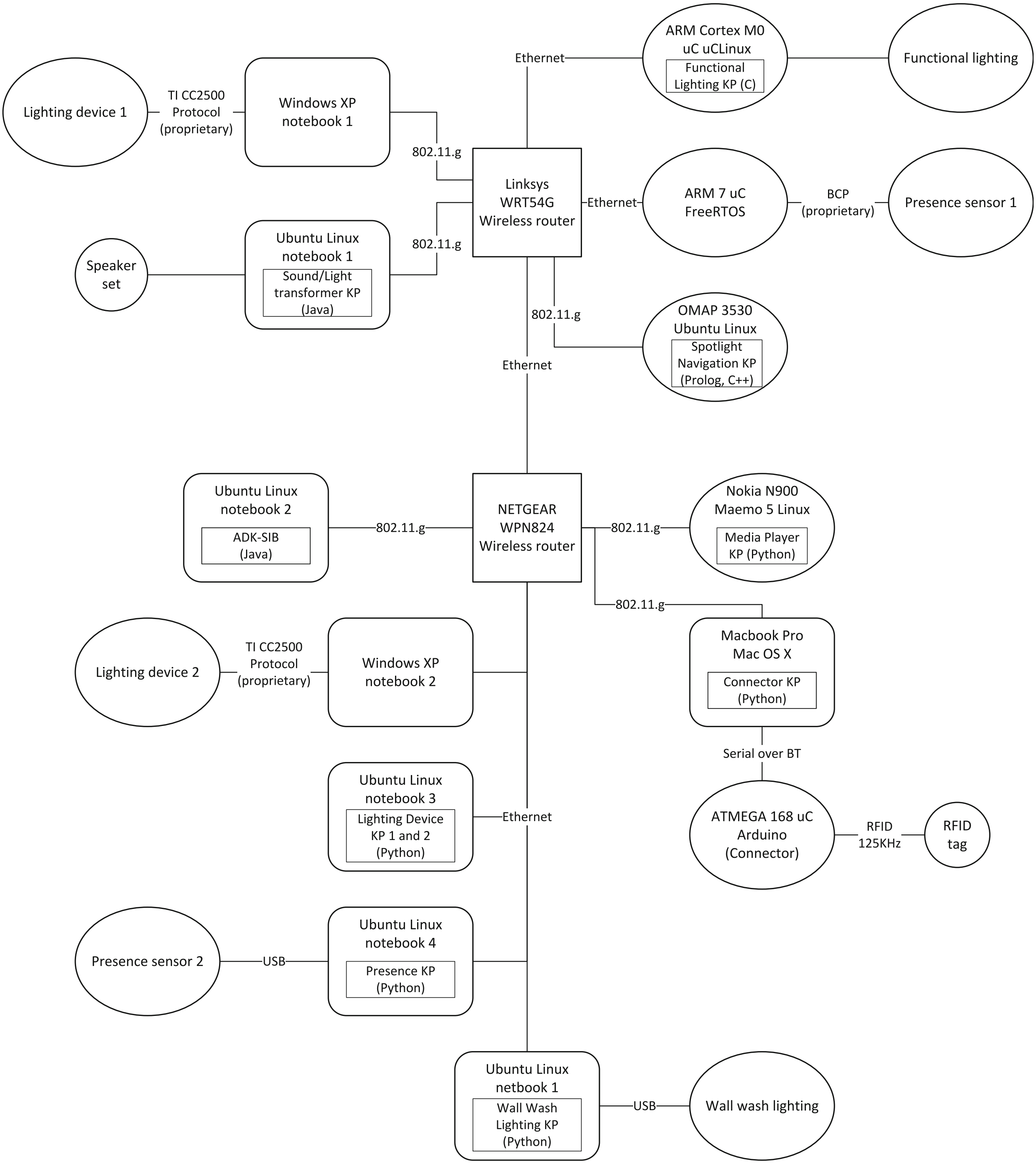

Fig. 6 A schematic overview of the hardware and software components in the pilot study setup

35), availability during the week of the pilot, and their mutual friendships. Of the recruited 21 participants that successfully completed the trials, 13 were male and 8 were female. Their age ranged from 23 to 34, with an average age of 28.5. Nine participants were living alone and 11 were cohabiting. The median score of self-report familiarity with interactive systems was 6 on a 1-through-7 scale.

Table 1 System specifications of components used in prototype system

Component	CPU	Operating system	Language
SIB	Intel Core 2 Duo 2.8 GHz	Ubuntu Linux	Java
SLT KP	Intel Core 2 Duo 2.2 GHz	Ubuntu Linux	Java
Connector KP	Intel Core 2 Duo 2.6 GHz	Mac OS X	Python
Music Player KP	ARM Cortex-A8	Maemo Linux	Python
Presence KP	Intel Pentium M	Ubuntu Linux	Python
NFL KP	Intel Pentium M	Ubuntu Linux	Python
Spotlight Navigation KP	OMAP 3540	Ubuntu Linux	Prolog and C++
Functional Lighting KP	ARM M0, ARM 7	uCLinux, FreeRTOS	C
LD KP 1 and 2	Intel Pentium	Windows, Linux	Python, C++

8.2 Materials

Figure 7 shows a brief overview of the different parts of the system. The experiment took place in two rooms, i.e., the study and the living room of the Experience Lab on the High Tech Campus in Eindhoven. The facilities and infrastructure of the Experience Lab were used to set up the prototype system and to collect observation data (video and audio recordings).

In the smart home pilot, media content is shared among several devices in a smart home setting. Music can be shared between a mobile device, a stereo speaker set and a Philips Living Colors lamp that can render the mood of the music with colored lighting. The music experience is also shared remotely between friends living in separate homes through the Living Colors lamp. This light and music information is shared between the two lighting devices. Other lighting sources, like the smart Functional Lighting

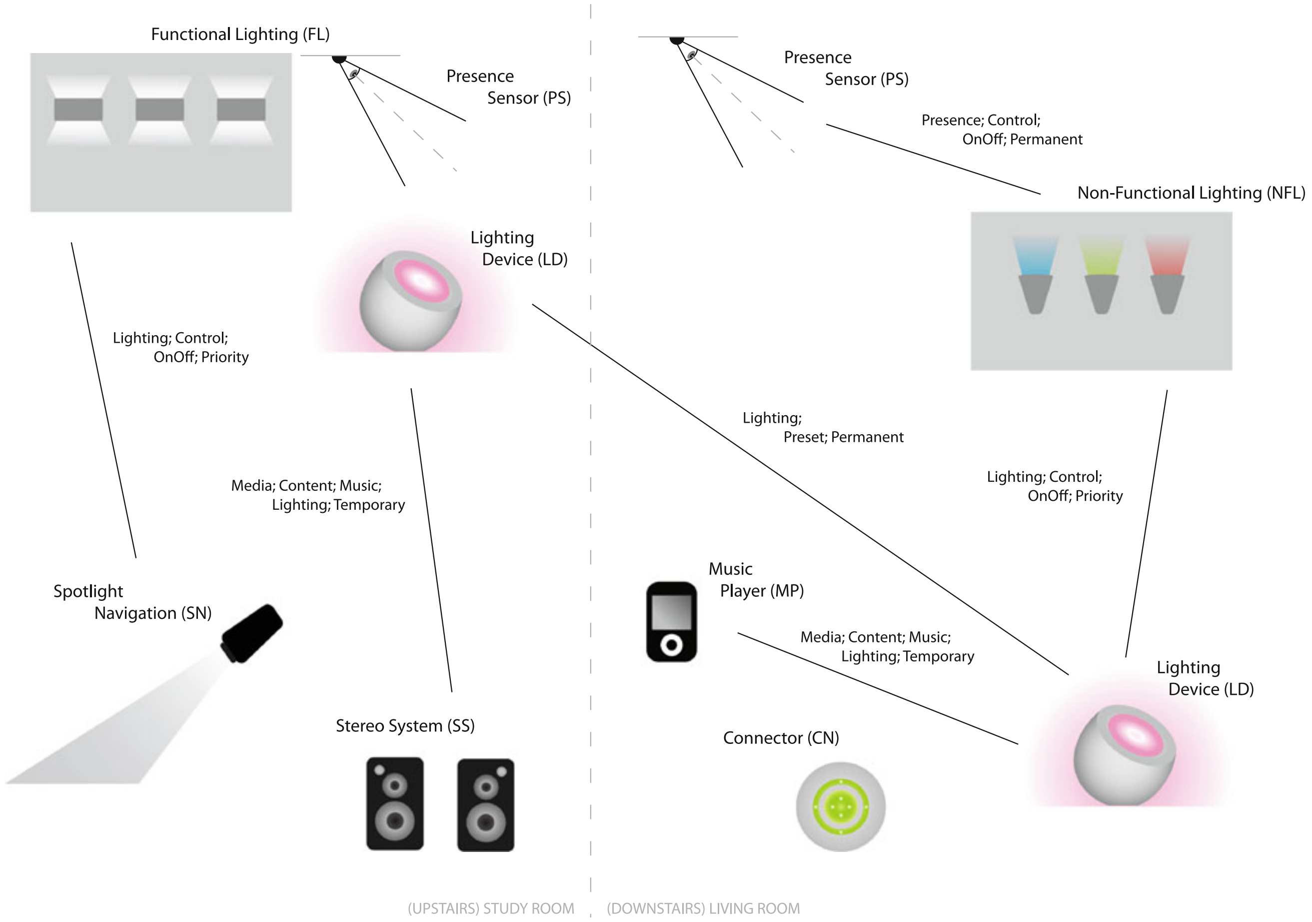

Fig. 7 The devices and their connections as used in the system

(FL, Fig. 7) and the smart wall-wash lights (NFL, Fig. 7) are sensitive to user presence and the use of other lighting sources in the environment. The full setup was built using the SOFIA software platform as described earlier in Sects. 3 and 7. The smart home pilot follows the following scenario:

> Mark and Dries enter their home. The intelligent lighting system detects their presence, and switches the lights on, and notifies the smart space about user presence. The decorative wall-wash lights are in turn notified of user presence by the smart space and turn themselves on. Mark and Dries start listening to music. They would like to try to render the music on a lighting device to also create some visual effects accompanying the music. They query the smart space and find out that the lighting device can render these light effects. They make a connection between the music player and the lighting device using the Connector. The light starts being rendered on the lighting device. To put the focus on the lighting device, the decorative wall-wash lights in the room automatically dim themselves down. At the same time, the light pattern also starts being rendered on the remote lighting device, where Mark's sister Sofia can observe the same light effects in her own house.
>
> At another location: after a while, Sofia is curious and wants to listen to the music that Mark and Dries are listening to. She connects her lighting device to her stereo using Spotlight Navigation, and the same song plays on her surround sound system.

8.3 Measurements

During the pilot, several measurement instruments were employed. Participants were asked to rate the pilot setup on three different scales; the HED/UT scale [35], the Perceived Control scale [36], and a questionnaire developed by the SOFIA project for internal use. The mental models that users developed during their interaction with the system were recorded using the teach-back protocol [37], and the participants attitudes toward the system were recorded with a semi-structured interview. Because the HED/UT scale and the Perceived Control scale were targeted at the entire system, we do not discuss their results in this paper.

8.3.1 Mental models

Mental models were extracted using the teach-back protocol. Because users' mental models consist of both semantic and procedural knowledge about the system they were interacting with, teach-back questions can be subdivided into "what is?" questions focusing on semantic knowledge, and "how to?" questions focusing on procedural knowledge [37]. Using such questions, adjusted to our specific situation and research goal, we aimed to extract the semantic and procedural concepts that were relevant for our users.

Participants were asked to explain to the researcher what they thought the system was and was for, including listing all the components and the relationships and connections between the components they thought made up the system. By asking for the perceived connections and relationships between the components, we aim to gain a better understanding of how users conceptualize the cause-and-effect relationships between their actions and the responses of the various devices in the smart home. This includes the information that is exchanged between these devices. By asking the participants to explain to the researcher how to perform a specific task with the system, we aim to get insights into how well the participants understood the necessary steps and devices involved to achieve their goal. To support and communicate their answers to both types of questions to the researchers (and for recording purposes), participants were asked to make drawings, schematics, or use a textual representation.

8.3.2 Interview

In order to gain a deeper insight into the things that occurred during the experiment sessions and record the users' general opinion, a semi-structured interview was conducted. Using a list of open questions as a structure, participants were evoked to share their experiences with the test setup and think along for possible improvements. During the interview, the researchers also asked questions based on specific behavior or actions of the participants that they observed during the trial.

8.4 Procedure

Participants had already received written information about the experiment together with an official invitation by e-mail. After the participants were welcomed in the Experience Lab and were briefed, they received and signed an informed consent form and were asked to fill out a pre-experiment survey. This survey included demographic questions and a self-report scale of familiarity with interactive systems like (tablet) PCs and smart phones.

The groups of three participants were split up in two groups of which two participants were lead to the living room area to perform the role of Mark and Dries, and one participant was taken to the study to perform the role of Sofia (these names will be used later to identify the different treatment groups). All participants were introduced to the devices which they had to interact with before the experiment started.

During the experiment, the participants were asked to perform a series of predefined tasks that revealed the functionality of the system. Every participant received these tasks on paper and was asked to think aloud, or for the participants in the living room (Mark and Dries), to share and discuss their thoughts during the whole experiment. After they performed the tasks, they were asked to freely explore the system to deepen their understanding and check their assumptions of its operation. They could continue this free exploration until they thought they understood the system's operation and would notify the researcher that they had finished. The researchers (one in the living room and one in the study) sat down in the back of the room during the entire session and were available in case anything went wrong.

After interacting with the demo, the participants were asked questions to elicit their mental models and were interviewed. The Mark and Dries characters were interviewed together, and they could openly discuss their opinions and mental models. Some of the participants agreed on their answers and agreed on one drawn representation of their shared mental model. Others disagreed and created their own representation. The duration of each trial was approximately 50–60 min, including briefing, instructions, filling out the questionnaires, and the closing interview.

9 Results

9.1 Mental models

Of the 21 participants who participated in the pilot, we collected 18 mental models. The teach-back protocol with the Sofia characters ($n = 7$) resulted in 7 unique mental models, while for the Mark ($n = 7$) and Dries ($n = 7$) characters, we obtained 11 mental models, of which three were shared. We will first give an overview of the overall results of the mental models, followed by a more detailed description of the mental models recorded from Sofia characters and the Mark en Dries characters (which we treated as one group).

9.1.1 Completeness

Out of all the mental models, 15 did not note that presence detection was used: 7 out of 7 for the Sofia characters and 8 out of 11 for the Mark and Dries characters. Of the three that included presence detection in their drawings, one was a shared model and the other two were from the same session.

A few other components of the system that were in the study and the living room were occasionally not included in the mental models. This includes the non-functional lighting (NFL) in the living room, the relationship between the NFL and the Living Color (LC) light (the NFL would dim down when the LC was active), the Functional Lighting (FL) in the study upstairs, and the dimming of the FL when the Spotlight Navigation was in use. The NFL was missing in 2 mental models, as was the connection between the NFL and the LC. These 2 mental models were from the same session. For the mental models of the Sofia characters, 1 out of 7 missed the FL and 2 were missing the connection between the FL and the Spotlight Navigation.

9.1.2 Semantic connections concepts

During the user experiments, some of the participants noticed and discussed interesting networking concepts like transitivity and directionality. These concepts were also considered in the semantic connections interaction model, but were not implemented in the pilot. Despite the absence of these concepts, participants did intentionally (or sometimes perhaps unintentionally) draw them in their mental models or discuss them. Among the concepts of our interest are directionality, transitivity, priority, and the temporary or persistent nature of the connections.

Transitivity was noted in 3 of the mental models and directionality in 9 of them. Two mental models indicated a notion of priority in their mental models, concerning one out of multiple conflicting connections to have priority over the others. Two persons discussed the persistence of connections, wondering when connections would stop existing (for instance when the person would take a mobile device out of the house) or indicating, what they described as a permanent connection, distinctively from the other non-permanent connections.

9.1.3 Organizational layout

We identified 3 types of organizational layouts in the way people draw their mental models. The majority used a physical/spatial way of describing their mental model, of which we identified 8 as being fully spatial (the main structure of the network is based on the physical location of the components) and another 8 mental models have what seems to be an arbitrary mapping, using the physical appearance of the components to identify them in the drawing. Some of these representations include spatial information but it is not used as their main structure. We label these hybrid layouts. There are 2 mental models that show a logical way of representing the network and its components using blocks and labels to identify the components. Figure 8 shows an logical organizational structure, while Fig. 9 shows a spatial one. Similar ways of organizing mental models were found in [27].

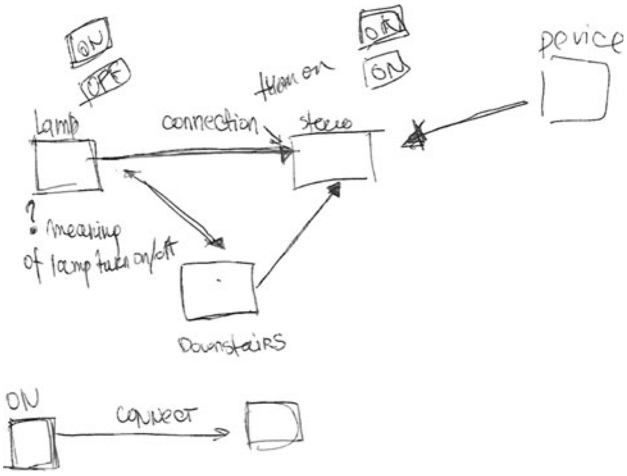

Fig. 8 Mental model drawing of a Sofia character with a logical representation

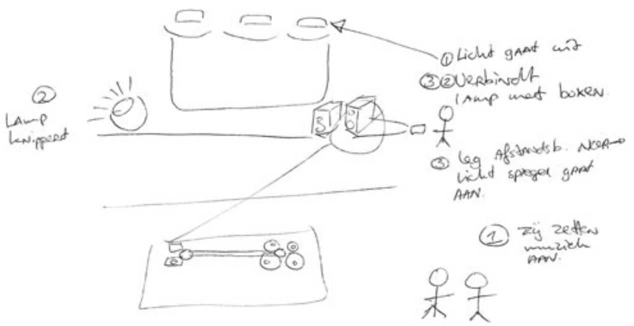

Fig. 9 Mental model drawing of a Sofia character with a spatial representation

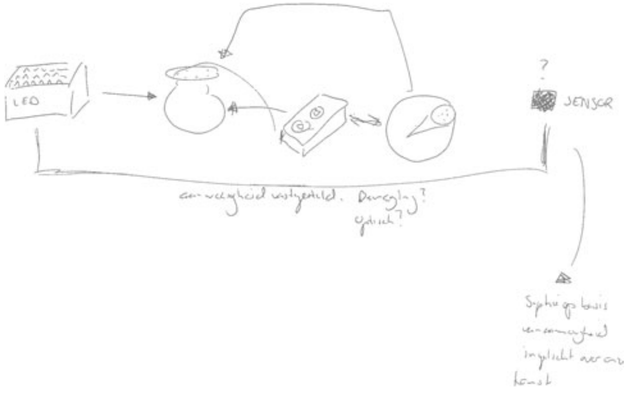

Fig. 10 Mental model drawing of a Mark/Dries character with the Connector as a central entity

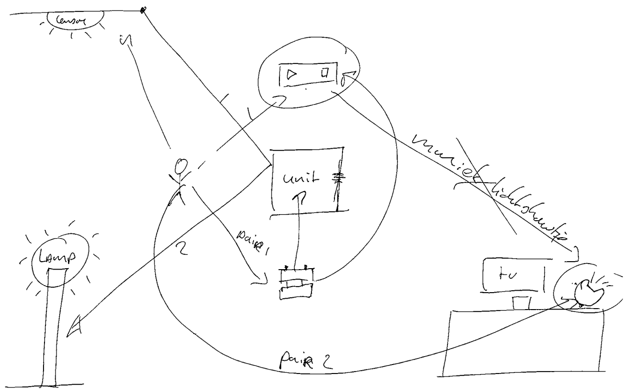

Fig. 11 Mental model drawing of a Mark/Dries character with an invisible central entity

9.1.4 Network structure

For the mental models of the Mark and Dries characters, we observed 3 main trends in the structure of the networks they drew. We distinguished between network structures that define a central entity (which is close to the actual network architecture), network structures that have a mainly peer-to-peer structure, and a mixed infrastructure which both have peer-to-peer connections and connections going through a central entity (the Connector object). All of these mental models of the network are compatible with the actual situation in the pilot. We observed 5 mental models with a central entity, 4 with mainly peer-to-peer connections, and 2 with a mixed structure. Figure 11 shows a network structure with a central, invisible entity, while Fig. 10 shows a network structure with the Connector as the central entity.

For the Sofia characters, we mainly observed two different network structures: A daisy-chained one (every component connects to one or two others in a serial manner) found in 5 mental models and a parallel structure (where connections had a more parallel nature) which occurred 2 times. What is interesting to note was that the Spotlight Navigation device was often seen as an entity that was not connected to the network, while the Connector object was in all cases considered part of the network (and in some cases even as being the central entity).

9.2 Interviews

From the interviews, we observed a few trends. Some trends were to be expected, while others were more surprising. Many participants were disappointed in the limited functionality of the current setup. Although the participants were enthusiastic about the ease with which the connections between devices could be made, they were disappointed that they could only control the connections between two devices, despite the fact that there were many more devices and appliances available (especially in the living room, for example, TV, stereo set, other light sources, and luminaries).

Most participants were enthusiastic about the "simple way" of making connections. However, they did indicate that they wanted to be more in control of what would actually happen when the connection was made. Some participants indicated that this lack of control was not crucial, because they figured that the connections could be undone in the same fashion when they did not like the effects of the connection. With regard to the overall functionality, participants also indicated that they would like to see more "practical applications" that would make their daily life easier. These remarks were mostly in the direction of concepts known from the home automation domain.

Several remarks were made concerning the user interaction with the Spotlight Navigation and Connector device. For the Spotlight Navigation, remarks were often made about the icons that were projected. The icon for the Living Color lamp was not always clear to users. Remarks were also made about the (mis)alignment of the icons and the physical devices, and many participants indicated that the icons could be omitted or be replaced by boxes around the physical objects. Additionally, remarks were made about the inaccuracy of the pointing gesture and difficulties in operating the button on the device.

For the Connector device, the low speed of the selection procedure was often mentioned. The effort required to physically select a device was often mentioned as a downside, while others mentioned it as a positive point as it was considered playful. Remarks were also made about the limited pairwise selection—participants indicated that they would want to have the possibility to select and connect more devices at the same time.

10 Discussion

Spotlight Navigation and the Connector are two alternative user interface approaches to configuring ubiquitous computing infrastructure. Although we cannot directly compare the mental models elicited during the user experiment, which would have asked for a more controlled setting (e.g., having the same setup and having an equal number of participants for both treatments), we did make some interesting observations.

The most striking difference between the way users described the setup was the perception of the users that the Connector was part (if not the central part) of the system, while the Spotlight Navigation was often considered outside of the system. We hypothesize that this is due to the different roles that the Spotlight Navigation and the Connector have in the interaction with the connections. The Connector is used to conceptually "carry" the content between the two devices and in itself represents the relationship between these two artifacts. The Spotlight Navigation is, in contrast, perceived as a "remote control" that visualizes the connections in physical space. This might lead the users to conclude that the projected lines are the connections, directly between the devices, and leave the Spotlight Navigation itself outside of this network.

The results show that devices and appliances that automatically act and react to people's behavior are often not considered in the mental models, compared to the devices and relations that users interact with explicitly. However, the results also show that some participants noticed these relations and incorporated them in their mental models. More interestingly, some of the participants expected that they could manipulate these relationships (e.g., between sensor and light) in the same way as they could manipulate the other connections. This result is promising and might indicate that people project their experience with one part of the system to the rest of it. This also became apparent when participants started looking for tags on other devices they though could also be connected.

An interesting observation is the rather direct impact of the interaction device's *design* on the mental models; for instance, the design and interaction of the Spotlight Navigation reminded them of a remote control, and consequently they used and described it as such. One participant even thought it was connected to the speaker set because it controlled the music (i.e., making or breaking a connection between the Living Colors lamp and the speaker set started and stopped the music playback).

It also showed that giving proper feedback (the right feedback, at the right time) is crucial when users are developing their mental model. When feedback is inconsistent or too late, the participants showed to be confused and change their assumptions of the system's internal working accordingly. Only when feedback is understood correctly (in terms of its meaning), is consistent, and is provided at the right time, designers can assume that the users will develop suitable mental models.

A performance measurement that was performed during the user experiments (as was already discussed at the beginning of Sect. 8) showed that the software architecture is fast enough to respond to a user's inquiry request (e.g., checking whether a certain connection is possible or already exists, or other user actions). The average response time was measured to be well within the maximum allowable value of 2 s, as is defined in [33].

Another observation was the complicated conceptions the participants had about the connections and their properties. Although there was no explicit directionality on the interactions or the connections, participants conceived the connections that for instance carried music to have a direction, travelling from its source to a destination. Directionality was also indicated where one device seemed to control the behavior of the other. By allowing users to use this sense of directionality in their interaction to establish the connection, we could easily give them more control over the connections. Transitivity was another, less obvious, concept that emerged from some of the mental models. Transitivity of a connection is a logical property that emerges when a network node A is connected to B, and in turn B is connected to node C. Transitivity then defines A to be connected to C as well. We observed participants to erase connections they indicated to exist before because they "were no longer needed" because of transitivity. And, in another case, worried about (hypothetically) removing a device from the network when it was in a chain of multiple connected devices, because it would lead to removing the transitive connections as well.

11 Future work

Currently, we are investigating the graphical design of the connections and icons projected by the Spotlight Navigation, to be more appealing and communicative. We are also investigating how to present different types of connections to the user. Currently, we only deal with audio connections, but we plan to distinguish between several different types of connections like: audio, visual (e.g., images and videos), textual (e.g., SMS and text) but also more abstract connections like control connections to UI elements (e.g., text-to-speech, buttons, and media controls), time-dependent events, and projecting GUI's for displayless devices. We are also busy generalizing the ontology to enable different types of connections and to integrate multimodal interaction data in order to enable semantic interaction in a smart environment.

For our experiments, we also had to bypass or simulate some of the system's functionality that can have an impact on the overall usability. Most importantly, our first Spotlight Navigation prototype has no capability of knowing where a specific device is located in the real world. We were assuming that a detection or setup step had already been performed, from which Spotlight Navigation learned in which direction it would find the respective devices. For the next iteration of the prototype, it is planned to use markerless visual object recognition technology that is expected to give sufficiently reliable results for the comparably few objects we have. We also expect that the recognition process can be aided by the IMU in the device, such that the angle information (from which angle the device is seen) can be utilized in the pattern matching process. Alternative approaches we are considering are visual markers (such as QR codes or any of the other available fiducials in the AR community), or actively generated systems such as infrared beacons and visually perceivable, time varying codes such as C-Blink [38, 39].

The SOFIA project tries to solve the interoperability problem by means of a blackboard-based approach. Some of the problems associated with current blackboard-based platforms are scalability and access rights. There is currently work being performed in the project to keep KPs from overloading the SIB, and KPI-based secure extensions to SSAP (used to communicate with the SIB) are being developed to enable access control.

12 Conclusion

The SOFIA project provides a platform and therewith the possibilities to improve the interoperability among devices. In this context, two prototypes were developed to experiment with tangible and augmented reality approaches to manage semantic connections. Both show their potential in moving the interaction with devices from a device-oriented paradigm toward a more task-oriented paradigm with increased interoperability. Although we are still exploring the possibilities of our approach, promising results and insights have been achieved already. The results obtained during this evaluation will be used to further define our semantic connections interaction model and may hopefully help other interaction designers to deal with design opportunities and challenges that emerge when designing for interoperability.

Acknowledgments We would like to thank Aly Syed, Riccardo Trevisan, Sriram Srinivasan, Hans van Amstel, Jettie Hoonhout, and Jolijn Teunisse for their contributions to the smart home pilot. This work is supported by Smart Objects For Intelligent Applications (SOFIA), funded by the European Artemis programme under the subprogramme SP3 Smart environments and scalable digital service.

References

1. Weiser M (1991) The computer for the 21st century. Scientific American
2. OWL web ontology language use cases and requirements. http://www.w3.org/TR/webont-req/ (2004)
3. Niezen G, van der Vlist B, Hu J, Feijs L (2010) From events to goals: Supporting semantic interaction in smart environments. In: 2010 IEEE symposium on computers and communications (ISCC), pp 1029–1034. doi:10.1109/ISCC.2010.5546634
4. Rapp S, Michelitsch G, Osen M, Williams J, Barbisch M, Bohan R, Valsan Z, Emele M (2004) Spotlight navigation: interaction with a handheld projection device. In: Advances in pervasive computing: a collection of contributions presented at PERVASIVE 2004. Oesterreichische Computer Gesellschaft, pp 397–400
5. Rapp S (2010) Spotlight navigation: a pioneering user interface for mobile projection. In: Proceedings of ubiprojection, Helsinki
6. van der Vlist B, Niezen G, Rapp S, Hu J, Feijs L (2012) Controlling smart home environments with semantic connections: a tangible and an AR approach. In: Design and semantics of form and movement (DeSForM) 2012. Wellington, New Zealand, pp 160–169

7. Siio I, Masui T, Fukuchi K (1999) Real-world interaction using the FieldMouse. In: Proceedings of the 12th annual ACM symposium on user interface software and technology—UIST'99, vol 1. ACM Press, pp 113–119. doi:10.1145/320719.322592. http://portal.acm.org/citation.cfm?doid=320719.322592
8. Ayatsuka Y, Rekimoto J (2005) tranSticks: physically manipulatable virtual connections. In: CHI'05: proceedings of the SIGCHI conference on human factors in computing systems. ACM, Portland, OR, USA, pp 251–260. doi:10.1145/1054972.1055008
9. Want R, Fishkin KP, Gujar A, Harrison BL (1999) Bridging physical and virtual worlds with electronic tags. In: CHI'99: proceedings of the SIGCHI conference on human factors in computing systems. ACM, New York, NY, USA, pp 370–377. doi:10.1145/302979.303111
10. Holmquist LE, Redström J, Ljungstrand P (1999) Token-based acces to digital information. In: HUC'99: proceedings of the 1st international symposium on handheld and ubiquitous computing. Springer, London, UK, pp 234–245
11. Rekimoto J (1997) Pick-and-drop: a direct manipulation technique for multiple computer environments. In: Proceedings of the 10th annual ACM symposium on user interface software and technology. ACM, Banff, AB, Canada, pp 31–39. doi:10.1145/263407.263505
12. Lee H, Jeong H, Lee J, Yeom K, Shin H, Park J (2008) Select-and-point: a novel interface for multi-device connection and control based on simple hand gestures. In: CHI'08: extended abstracts on Human factors in computing systems. ACM, Florence, Italy, pp 3357–3362. doi:10.1145/1358628.1358857
13. Rekimoto J, Ayatsuka Y, Kohno M, Oba H (2003) Proximal interactions: a direct manipulation technique for wireless networking. In: Proceedings of human–computer interaction; INTERACT'03. IOS Press, Amsterdam, The Netherlands, pp 511–518
14. Fitzmaurice GW (1993) Situated information spaces and spatially aware palmtop computers. Commun ACM 36:39–49. doi:10.1145/159544.159566.
15. Rukzio E, Holleis P (2010) Projector phone interactions: design space and survey. In: Workshop on coupled display visual interfaces at AVI 2010. Rome, Italy
16. Ballagas R, Borchers J, Rohs M, Sheridan JG (2006) The smart phone: a ubiquitous input device. IEEE Pervasive Comput 5(1):70–77. doi:10.1109/MPRV.2006.18
17. Foley JD, Wallace VL, Chan P (1984) The human factors of computer graphics interaction techniques. IEEE Comput Graph Appl 4(11):13–48
18. Rukzio E, Leichtenstern K, Callaghan V, Holleis P, Schmidt A, Chin J (2006) An experimental comparison of physical mobile interaction techniques: touching, pointing and scanning. In: 8th international conference on ubiquitous computing (UbiComp'06). Springer, pp 87–104. http://www.springerlink.com/index/h6jut558g604x083.pdf
19. Hinckley K, Ramos G, Guimbretiere F, Baudisch P, Smith M (2004) Stitching: pen gestures that span multiple displays. In: Proceedings of the working conference on advanced visual interfaces. ACM, pp 23–31. http://portal.acm.org/citation.cfm?id=989866
20. Molyneaux D, Gellersen H, Kortuem G, Schiele B (2007) Cooperative augmentation of smart objects with projector–camera systems. In: Proceedings of the 9th international conference on ubiquitous computing. Springer, pp 501–518. http://portal.acm.org/citation.cfm?id=1771592.1771621
21. Beardsley P, van Baar J, Raskar R, Forlines C (2005) Interaction using a handheld projector. IEEE Comput Graph Appl 25(1): 39–43. http://www.ncbi.nlm.nih.gov/pubmed/15691170
22. Heider T, Kirste T (2002) Supporting goal-based interaction with dynamic intelligent environments. In: Proceedings of the 15th European conference on artificial intelligence. Lyon, France
23. Torge S, Rapp S, Kompe R (2002) The planning component of an intelligent human–machine interface in changing environments. In: Multi-modal dialogue in mobile environments (IDS-02), ISCA Tutorial and Research Workshop (ITRW)
24. Newman MW, Sedivy JZ, Neuwirth CM, Edwards WK, Hong JI, Izadi S, Marcelo K, Smith TF (2002) Designing for serendipity: supporting end-user configuration of ubiquitous computing environments. In: Proceedings of the 4th conference on designing interactive systems: processes, practices, methods, and techniques, DIS'02. ACM, New York, NY, USA, pp 147–156. doi:10.1145/778712.778736
25. Oliver I, Honkola J (2008) Personal semantic web through a space based computing environment. In: Middleware for the semantic web, second IEEE international conference on semantic computing. Santa Clara, CA, USA. http://arxiv.org/pdf/0808.1455
26. van der Vlist B, Niezen G, Hu J, Feijs L (2010) Semantic connections: exploring and manipulating connections in smart spaces. In: 2010 IEEE symposium on computers and communications (ISCC), pp 1–4. doi:10.1109/ISCC.2010.5546636
27. Poole ES, Chetty M, Grinter RE, Edwards WK (2008) More than meets the eye: transforming the user experience of home network management. In: Proceedings of the 7th ACM conference on designing interactive systems, DIS'08. ACM, New York, NY, USA, pp 455–464. doi:10.1145/1394445.1394494
28. Wensveen SAG, Djajadiningrat JP, Overbeeke CJ (2004) Interaction frogger: a design framework to couple action and function through feedback and feedforward. In: Proceedings of the 5th conference on designing interactive systems: processes, practices, methods, and techniques, pp 177–184
29. van der Vlist B, Niezen G, Hu J, Feijs L (2010) Design semantics of connections in a smart home environment. In: Chen LL, Djajadiningrat T, Feijs L, Kyffin S, Steffen D, Young B (eds) Proceedings of design and semantics of form and movement (DeSForM) 2010. Koninklijke Philips Electronics N.V., Lucerne, Switzerland, pp 48–56
30. Kwak M, Niezen G, van der Vlist B, Hu J, Feijs L (2011) Tangible interfaces to digital connections, centralized versus decentralized. In: Pan Z, Cheok A, Müller W, Yang X (eds) Transactions on edutainment V, Lecture Notes in Computer Science, vol 6530. Springer, Berlin, pp 132–146
31. Sutherland IE (1968) A head-mounted three dimensional display. In: AFIPS'68 (fall, part I): proceedings of the December 9–11, 1968, fall joint computer conference, part I. ACM, New York, NY, USA, pp 757–764
32. Honkola J, Laine H, Brown R, Tyrkko O (2010) Smart-M3 information sharing platform. In: 2010 IEEE symposium on computers and communications (ISCC). Riccione, Italy, pp 1041–1046
33. Miller RB (1968) Response time in man–computer conversational transactions. In: Proceedings of the fall joint computer conference. ACM Press, p 267. doi:10.1145/1476589.1476628
34. Niezen G, van der Vlist BJJ, Bhardwaj S, Ozcelebi T (2012) Performance evaluation of a semantic smart space deployment. In: IEEE international conference on pervasive computing and communications workshops, pp 835–841. doi:10.1109/PerComW.2012.6197628
35. van der Heijden H, Sorensen LS (2003) Measuring attitudes towards mobile information services: an emperical validation of the hed/ut scale. In: European conference on information systems (ECIS). Napels, Italy
36. Hinds P (1998) User control and its many facets: a study of perceived control in human-computer interaction. Technical report, Hewlett Packard Laboratories. http://www.hpl.hp.com/techreports/98/HPL-98-154.html
37. van der Veer GC, del Carmen Puerta Melguizo M (2003) The human–computer interaction handbook. In: Mental models. Lawrence Erlbaum Associates Inc., Hillsdale, NJ, USA, pp 52–80

38. Want R, Hopper A, Falcão V, Gibbons J (1992) The active badge location system. ACM Trans Inf Syst 10(1):91–102
39. Miyaoku K, Higashino S, Tonomura Y (2004) C-blink: a hue-difference-based light signal marker for large screen interaction via any mobile terminal. In: UIST'04: proceedings of the 17th annual ACM symposium on user interface software and technology. ACM, New York, NY, USA, pp 147–156