**Gate reminder [1]:**

This concept centralizes around providing reminders at the moment a user leaves their house. Knowledge about possibly forgotten items is obtained through the use of RFID tags, focusing on a zero user workload interaction. A crucial part in its working is that it is focused on Korean household, where shoes are generally left at the front door, so there is a clearly defined time slot in which all tags can be analyzed. Focus on the study was mostly the actual prototype rather than any smart algorithm.

*Properties:*

* Physical prototype
* Transparent interaction
* Object detection

**Decision maker [2]:**

This concept intercepts notifications from all sources and processes them in a “decision maker” prior to actually arriving at the user. Instead, it processes information from sensors and IoT devices within user and environment contexts to decide upon the target device, type of notification and time of notification. This is done using a machine learning approach. Rather than analyzing the actual patterns in decisions on whether to and how to notify, the paper continues by focusing mostly on the speed and accuracy of various machine learning algorithms.

*Properties:*

* Machine learning
* Habit analysis

**Smart reminder system [3]:**

This concept creates a smart reminder system through three major components: activity recognition, location recognition and prediction. The activity recognition is done through the use of analysis of the hand movements over time and applying machine learning algorithms and fuzzy logic to map this to activities. Location recognition is done through image recognition by camera and neural networks. These two are then combined to analyze coupled activities, two activities that are strongly related. Alongside, predictions are made regarding pending and forgotten activities. As such reminders can be produced when likely to be forgotten activities should occur.

*Properties:*

* Specific setup

**Olisto/IFTTT/CybreMinder/CAMP [4]–[7]:**

These apps and concepts allow setting reminders based on various aspects of user and environment contexts. Once the current situation satisfies all conditions in all contexts, the user is automatically notified. Information is retrieved from the user’s (IoT) devices and (online) services. No form of pattern recognition or prediction is done, however.

*Properties:*

* Existing (possibly discontinued) apps

**CogKnow [8]:**

This concept actually touches upon user values, but instead uses them to define the required support. A distinct number of support scenarios are handled and rulesets are defined accordingly. Predominantly the user context is considered rather than anything else. The rulesets are aimed at avoiding interruptions of important activities, but don’t do any further analysis.

**Fuzzy linguistics [9]:**

This concept uses fuzzy logic and linguistic variables to analyze the urgency of the reminder and the level of annoyance created by the interruption of the current activity. Resulting from this is a reminder level which determines whether or not the reminder is delayed and/or how the reminder is presented. The urgencies and other levels are all given at design time, however, and are averaged over all users tested prior.

**Goal models [10]:**

This concept does not directly involve itself with reminders, but rather with linking certain activities to achieving certain goals. These activities may have complex relations with one another and may promote or demote a goal. As such, this can be similarly applied to activities aiming to achieve a certain goal where the promotions and demotions are linked to the user values.

*Special properties:*

* Linking activities to goals
* Not related to reminders

**PAIR [11]:**

This is a relatively older paper which describes one of the first, more advance planners. It takes into consideration several rules as prescribed by the user or caregiver and lays them alongside the activities of the user to provide appropriate reminders. However, no dynamic analysis is done, only design time rules are analyzed.

**AHCS/TAFETA [12], [13]:**

These concepts attempt to design a context-aware application which analyses data from various sensors within the user’s house. AHCS makes use of the CASanDRA framework [14] in order to create awareness of the user’s context. The CASanDRA framework is a middleware which provides easily consumable context information and accepts different information inputs which are fused together. The concepts use either the middleware or their own AI to analyze the collected information and compare this with a number of predefined rules to provide detailed information on the user to the caregiver and provide reminders when rules are broken.

*Special properties:*

* Context analysis independent from reminder system
* Levels and types of alerting

**Supporting daily activities [15]:**

Why is this PDF not available anywhere…? It seems very interesting: mailed the author.

**HeadacheCoach [16]:**

While not directly a reminder system, HeadacheCoach does propose a possibly usable system. It uses user and environmental context analysis to identify possible triggers for a headache and consequently provides possible solution. A similar approach may be used to identify moments of lower cognitive ability in order to preempt a reminder being necessary at all.

**What should I do/Action Hierarchies [17], [18]:**

These two papers, while again not a directly related to reminders, do portray several underlaying concepts. The first paper presents a framework which represents hierarchical relationships among actions. This is formalized in the second paper. Secondly, this framework shows how the relationships tie in with promotion and demotion of values. Lastly, a method is shown on how to infer norms from values rather than vice versa. However, this remains a very theoretical paper.

*Special properties:*

* Values → Norms
* Not directly related to reminders
* Action hierarchy

**CIA [19]:**

Although this paper clearly states “smart reminder”, it doesn’t actually do much in regards to reminding. Instead, it uses image recognition to identify people. After this identification it combines information previously gathered through various systems to display information regarding this person and possible events and reminders tied to them.

*Special properties:*

* Linking information
* Not directly related to reminders

**Attelia [20]:**

Attelia is a middleware concept which intercepts any notifications. It analyses breakpoints in the user’s mobile interactions and adaptively delivers the notification to minimize interruptions and the user’s attentional overload. As such, it lowers the user’s frustration caused by receiving too many notifications.

*Special properties:*

* Focuses on mobile screen use to analyze activity

**TEREDA [21]:**

Another concept not directly related to reminders. It works by gathering simple data from many sensors around the house and feeding that into the middleware. From this, distributions for the start time and duration are analyzed and used to help recognize activities and cluster them by starting time. For example, there might be 4 clusters of starting times in which the user may generally start to watch TV (with corresponding durations). Each of these clusters may have different subsequent activities, each with different likelihoods. As such, this temporal analysis may be used to predict the likely following activity.

*Special properties:*

* Activity clustering
* Prediction of next activity without machine learning

**MLCARS [22]:**

This dissertation discusses a concept which uses machine learning to analyze shopping items and where they were bought (or cleared off the to-do list) to predict similar available items or similar stores. This data is collected among all users and combined with information from companies and stores and ultimately stored in a database which is continuously updated. Combining this with the data of the user’s shopping list as well as their location allows to provide appropriately timed reminders not to forget items from their shopping list. These reminders are not just when near their usual supermarket (like is already possible with location-based reminders) but also when close to any store that is expected to have the desired item.

*Special properties:*

* Activity clustering
* Prediction of next activity without machine learning

**MagHive [23]:**

This honeycomb shaped magnetic smart surface is attached to the wall and allows devices and other objects to be placed on them. Aside from the useful functionalities such as wireless phone charging, it uses NFC and QI technologies to detect the presence and identity of the objects. As such it is able to remind the user when he or she forgets to take or put back an item.

*Special properties:*

* Actual product
* Provides a great base for further development

**Long term evaluation of smart homes [24]:**

Another one not related to reminders per se. This dissertation reviews the users values over long time use of smart home appliances. Their conclusions span generally across all types of smart home appliances. In order for the appliances to provide usefulness it is important that the values of accessibility and trust are upheld. Any appliance which does promote accessibility immediately diminishes any usefulness for the user. Trust generally boils down to the reliability of the provided functionality. If the product still has function impairing bugs, users are likely not to use the product. Even if the producer manages to fix the flaws, the lost trust takes vast time to recover. Another drawn conclusion is that whatever solution implemented, users are initially curious and excited and are willing to try most ideas, but ultimately go back to their routine behavior. As such, the smart appliance should blend into this rather than interrupting it.

| Concept | RP | RI | SS | Tim | Loc | Act | Env | Dyn | UV | IA | Ref. |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Gate reminder | x |  | x |  |  |  | x |  |  |  | [1] |
| Decision maker |  | x |  | x | x | x | x | x |  |  | [2] |
| SRS | x |  | x |  | x | x |  | x |  |  | [3] |
| Olisto | x |  |  | x | x | x | x |  |  |  | [4] |
| IFTTT | x |  |  | x | x | x | x |  |  |  | [5] |
| CybreMinder | x |  |  | x | x | x |  |  |  |  | [6] |
| CAMP | x |  |  |  |  |  |  |  |  |  | [7] |
| CogKnow | x |  | x | x | x | x |  |  | x\* | x | [8] |
| Fuzzy lingustics |  | x |  |  |  | x |  |  |  | x | [9] |
| Goal models |  |  |  |  |  | x |  | x |  |  | [10] |
| PAIR |  | x |  |  |  | x |  |  |  |  | [11] |
| AHCS | x |  |  | x | x | x | x |  |  |  | [12] |
| TAFETA | x |  |  | x | x | x | x |  |  |  | [13] |
| SDA |  |  |  |  |  |  |  |  |  |  | [15] |
| HeadacheCoach |  |  |  | x | x | x | x |  |  |  | [16] |
| WSID/AH |  |  |  | x | x | x | x | x | x |  | [17], [18] |
| CIA |  |  |  |  |  |  |  |  |  |  | [19] |
| Attelia |  | x |  |  |  |  |  |  |  | x | [20] |
| TEREDA |  |  |  | x |  | x |  | x |  |  | [21] |
| MLCARS | x |  |  |  | x |  |  | x |  |  | [22] |
| MagHive | x |  | x |  |  |  | x | x |  |  | [23] |
| LTE SH |  |  |  |  |  |  |  |  | x |  | [24] |

\* Only at design time

**Legend:**

RP: Reminder Producing

RI: Reminder Intercepting

SS: Specific setup

Tim: Time

Loc: Location

Act: Activity

Env: Environment

Dyn: Dynamic

UV: User Values

IA: Interrupt Analysis

**Discussion**

Overall the smart reminder concepts can be divided based on two distinct features. Firstly, those requiring a specific set-up of hardware. And secondly, those that use any sort of dynamic AI over simply hardcoding rules at design time.

The specific set-up concepts are quite apt and able for those scenarios, but quickly fall short when applied to other scenarios or when generalizing the solution.

The majority of the concepts use (or can use) information about the user’s activity to some extent. This activity information can either be used to instigate the notification, or to analyze However, there are a few which take it further and use machine learning or other methods in order to predict the subsequent activity.

Generally, user values are not something linked to timing smart reminders. To less or more extent, however, they are being used at design time to shape the model.

So, what is useful? There is no existing implementation that can immediately be extended with user values. However, there are several implementations that contain interesting ideas that can be combined. Most notably [4], [9], [10], [12], [13], [17], [21].

**Feasibility of base papers**

If the desire is to combine the concept of the aforementioned papers, just like with any store-bought product, it is important to check whether they actually deliver. Additionally, the question is to what extent these concepts can be used to create a value based smart reminder system.

Firstly, the data collection concepts. Thank to close ties with the company behind Olisto [4], access is granted to all services and code behind. As such, a simple middleware can easily be built and integrated into their existing infrastructure. Using their information provides direct insight into events (such as device alarms) and states of devices (such as door open or closed) and services (such as weather). This is already an up and running platform, so lots of data is readily available.

Aside from gathering and analyzing data ourselves, there are numerous existing data sets. Two of which, are directly provided [25], [26] and probably more are available. These datasets have a range of activities and other data recorded over time. Note that in [2], the first dataset was used, but synthetically enhanced to add several properties such as the user activity other than call information and mobile phone usage. The second dataset has a limited but clear number of activities which are recognized and as such more readily usable. These, and similar, datasets can be used both for design and for testing. The range of activities recorded in these datasets and platforms limits the applicable scenarios. As such the data source should be chosen before the initial designs are done.

Secondly, how to analyze the incoming data. The part of the program analyzing this, or the middleware, is dependent on the type of incoming data. If fully detailed activity information is incoming, the middleware is not necessary since the any further analysis or manipulation can be directly performed on the data. However, when taking information from sensors, such a middleware has to be used to filter any interesting information.

The first solution is writing such a middleware from scratch. This is the most labor intense solution. However, if the other middleware are not easily implementable or require extensive rewrites, starting from scratch may actually require less work. In [13], they did just that; they designed their own middleware. However, it cannot be used since it remains exactly that, a design. In [12], however, they used an existing middleware [14]. In combination with a context toolkit [27], also used in CybreMinder [6]. The CASanDRA framework [14], however shows great promise since it’s actively used. However, up to this moment, 03-08, I have not been able to find the actual implementation. As such I have contacted the authors of the original paper and those of papers which used/referenced it.

Lastly, the most important aspect is the actual analysis. A wonderful starting point from within this research group is that of Tielman [17]. Combining its ideas of action hierarchies and values with that of goal reasoning [10] and possibly that of temporal analysis [21] could lead to very interesting results. The goal reasoning will allow for analyzing the possibilities of the moments of reminding (i.e. before it’s too late). The temporal analysis will allow for better predictions. The authors have been contacted for the actual code behind. However, the description is clear enough to incorporate it without it as well.

**Initial design idea:**

* Do not use any existing middleware
* Use existing datasets as a basis for design
* Construct a middleware for data obtained through Olisto for later testing
* Describe several basic scenarios to test
* Optional: Design basic software code based on [21]
* Design and create basic software based on [10], [17]
* Create analysis of maximizing value (with and without prediction)

**References**

[1] S. W. Kim, M. C. Kim, S. H. Park, Y. K. Jin, and W. S. Choi, “Gate Reminder: A Design Case of a Smart Reminder,” in *Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, New York, NY, USA, 2004, pp. 81–90.

[2] F. Corno, L. D. Russis, and T. Montanaro, “A context and user aware smart notification system,” in *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, 2015, pp. 645–651.

[3] H. T. Chaminda, V. Klyuev, and K. Naruse, “A smart reminder system for complex human activities,” in *2012 14th International Conference on Advanced Communication Technology (ICACT)*, 2012, pp. 235–240.

[4] “Olisto makes smart thing smarter, according to your rules.,” *Olisto*. [Online]. Available: https://olisto.com/. [Accessed: 19-Apr-2018].

[5] IFTTT, “IFTTT helps your apps and devices work together.” [Online]. Available: https://ifttt.com. [Accessed: 19-Apr-2018].

[6] A. K. Dey and G. D. Abowd, “CybreMinder: A Context-Aware System for Supporting Reminders,” in *Handheld and Ubiquitous Computing*, 2000, pp. 172–186.

[7] S. Vurgun, M. Philipose, and M. Pavel, “A Statistical Reasoning System for Medication Prompting,” in *UbiComp 2007: Ubiquitous Computing*, 2007, pp. 1–18.

[8] D. Zhang, M. Hariz, and M. Mokhtari, “Assisting Elders with Mild Dementia Staying at Home,” in *2008 Sixth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 2008, pp. 692–697.

[9] S. Zhou, C.-H. Chu, Z. Yu, and J. Kim, “A context-aware reminder system for elders based on fuzzy linguistic approach,” *Expert Syst. Appl.*, vol. 39, no. 10, pp. 9411–9419, Aug. 2012.

[10] P. Giorgini, J. Mylopoulos, E. Nicchiarelli, and R. Sebastiani, “Reasoning with Goal Models,” in *Conceptual Modeling — ER 2002*, 2002, pp. 167–181.

[11] L. S. Shafti, P. A. Haya, M. García-Herranz, and X. Alamán, “Personal Ambient Intelligent Reminder for People with Cognitive Disabilities,” in *Ambient Assisted Living and Home Care*, 2012, pp. 383–390.

[12] A. Hristova, A. M. Bernardos, and J. R. Casar, “Context-aware services for ambient assisted living: A case-study,” in *2008 First International Symposium on Applied Sciences on Biomedical and Communication Technologies*, 2008, pp. 1–5.

[13] A. Arcelus, M. H. Jones, R. Goubran, and F. Knoefel, “Integration of Smart Home Technologies in a Health Monitoring System for the Elderly,” in *21st International Conference on Advanced Information Networking and Applications Workshops, 2007, AINAW ’07*, 2007, vol. 2, pp. 820–825.

[14] A. M. Bernardos, P. Tarrío, and J. R. Casar, “CASanDRA: A Framework to Provide Context Acquisition Services ANd Reasoning Algorithms for Ambient Intelligence Applications,” in *2009 International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2009, pp. 372–377.

[15] H. Eon Byun and K. Cheverst, “Exploiting user models and context-awareness to support personal daily activities,” Jan. 2001.

[16] A. Fritzen, N. Leipold, N. Terzimehic, M. Böhm, and H. Krcmar, “HeadacheCoach: Towards Headache Prevention by Sensing and Making Sense of Personal Lifestyle Data,” 2017.

[17] M. L. Tielman and C. M. Jonker, “What should I do? Deriving norms from actions,values and context,” p. 5.

[18] P. Pasotti, C. M. Jonker, and M. B. van Riemsdijk, “Towards a formalisation of Action Identiﬁcation Hierarchies∗.”

[19] F. Kargl, B. Dong, T. Illmann, and M. Weber, *Smart Reminder - Personal Assistance in a Mobile Computing Environment*. 2002.

[20] T. Okoshi, H. Nozaki, J. Nakazawa, H. Tokuda, J. Ramos, and A. K. Dey, “Towards attention-aware adaptive notification on smart phones,” *Pervasive Mob. Comput.*, vol. 26, pp. 17–34, Feb. 2016.

[21] E. Nazerfard, P. Rashidi, and D. J. Cook, “Using Association Rule Mining to Discover Temporal Relations of Daily Activities,” in *Toward Useful Services for Elderly and People with Disabilities*, 2011, pp. 49–56.

[22] P. Shanahan, “Machine Learning for Context-aware Reminders and Suggestions,” PhD Thesis, University of California at San Diego, La Jolla, CA, USA, 2009.

[23] “MagHive - World’s First Modular Smart Reminder (Canceled),” *Kickstarter*. [Online]. Available: https://www.kickstarter.com/projects/2034560442/maghive-worlds-first-modular-smart-reminder. [Accessed: 24-Jul-2018].

[24] G. Sandström and Kungliga tekniska högskolan (Stockholm), *Smart homes and user values: long-term evaluation of IT-services in residential and single family dwellings*. Stockholm: Royal Institute of Technology, 2009.

[25] N. Eagle and A. (Sandy) Pentland, “Reality mining: sensing complex social systems,” *Pers. Ubiquitous Comput.*, vol. 10, no. 4, pp. 255–268, May 2006.

[26] “Activity Recognition Challenge Dataset Download | Opportunity.” [Online]. Available: http://www.opportunity-project.eu/challengedatasetdownload. [Accessed: 03-Aug-2018].

[27] D. Salber, A. K. Dey, and G. D. Abowd, “The Context Toolkit: Aiding the Development of Context-enabled Applications,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 1999, pp. 434–441.