iMAT: Intelligent Medication Administration Tools

P. H. Tsai, C. Y. Yu
Department of Computer Science
National Tsing Hua University
Hsinchu, Taiwan
peipei@iis.sinica.edu.tw, rayswin@gmail.com

W. Y. Wang, J. K. Zao
Senior Fellow, IEEE
Department of Comp. Science
National Chiao Tung University
Hsinchu, Taiwan
chesnut0802i@gmail.com, jkzao@cs.nctu.edu.tw

H. C. Yeh, C. S. Shih,
Member, IEEE
Department of Comp. Science and Information Engineering
National Taiwan University
Taipei, Taiwan
r94048@csie.ntu.edu.tw, cshih@csie.ntu.edu.tw

J. W. S. Liu,
Fellow, IEEE
Institute of Information Science
Academia Sinica
Taipei, Taiwan
janeliu@iis.sinica.edu.tw,

Abstract—**iMAT is a system of automatic medication dispensers and software tools. It is for people who take medications on long term basis at home to stay well and independent. The system helps its users to improve rigor in compliance by preventing misunderstanding of medication directions and making medication schedules more tolerant to tardiness and negligence. This paper presents an overview of the assumptions, models, architecture and implementation of the system.**

Keywords-Medication errors, Medication administration tool, Medication scheduler

I. Introduction

This paper describes a system of smart medication dispensers, medication schedule managers, and supporting software tools. The system, called iMAT (intelligent medication administration tools), targets as users the growing population of elderly individuals and people with chronic conditions who are well enough to maintain active, independent lifestyles. Such a person may take many prescriptions and over the counter (OTC) medications and health supplements at home and work without close professional supervision. In subsequent discussions, by a *user*, we mean such a person.

Nowadays, modern drugs can help people conquer previously fatal diseases, control debilitating conditions, and maintain wellness and independence well into old age, provided that the drugs are taken as directed. Unfortunately, even critically important drugs such as those for treatments of hypertension, diabetes, and heart conditions are often not taken as directed [1]. The fact is that statistics on health care quality continue to show alarmingly rates and serious consequences of preventable medication errors [2-6]. *Administration errors* due to non-compliance to medication directions are known to contribute 25 – 40% of all preventable medication errors and are the cause of approximately 10 % of hospital admissions and 23% of nursing home admissions. The primary goal of iMAT is to prevent administration errors as much as possible and when errors occur despite prevention efforts, reduce the adverse effects caused by them. iMAT can also help to make sure that interactions among all medications of each user and their interactions with food and drink have been properly accounted for by the directions for the user.

A look at causes of non-compliance points out that information technology can help eliminate many common ones, including misunderstanding of medication directions, inability to adhere to complex medication regimens, and inconvenience of rigid schedules. iMAT is designed specifically to eliminate these causes. A user of iMAT medication dispenser and schedule manager has no need to understand the directions of her/his medications. iMAT enables the pharmacist of each user to extract a machine readable *medication schedule specification* (*MSS*) from the user's prescriptions and OTC directions. Once loaded into an iMAT dispenser or schedule manager, the tool automatically generates a medication schedule that meets all the constraints specified by the user's MSS. Based on the schedule, the tool reminds the user at the times when some doses should be taken and provides instructions on how the doses should be taken (e.g., with 8 oz of water, no food within 30 minutes, etc.) In this way, iMAT helps to make complex regimens easy to follow.

Directions of modern medications typically provide some flexibility in choices of dose sizes and times as well as instructions on what to do in case of late or missed doses. By taking advantage of this leeway, the tools make the user's medication schedule easy to adhere and tolerant to user's tardiness. Some users take medications for months and years. Late and miss doses are unavoidable. For this reason, iMAT dispenser and schedule manager monitor user's response to reminders, adjust the medication schedule as instructed by MSS when the user is tardy, and when a non-compliance event becomes unavoidable, sends alert and notification in ways specified by MSS and the user. In this way, they help to reduce the rate and effect of non-compliance.

The remainder of the paper is organized as follows: Section II first describes how iMAT component tools fit in the chain of tools and information systems for medication use process (i.e., the process of ordering, transcription, dispensing and administration [7]). It then compares and contrasts iMAT with other medication usage assistance devices and systems. Section III presents key assumptions that must be valid for iMAT tools to work and discusses rationales with the help of an illustrative scenario. Section IV presents the timing and dosage constraints parameters defined by MSS. The generation of such specifications needs a database containing machine readable directions of commonly used medications. The contents of iMAT database are extractions from comprehensive drug information systems, including PDRHealth [8]. Section V describes a smart medication dispenser prototype and alternative configurations of the schedule manager. Section VI summarizes the paper and discusses future work.

II. Overview of iMAT and Related Works

Figure 1 shows how iMAT fits in a chain of information systems and tools for medication use process: It complements computerized physician order entry (CPOE) systems [9-14] at the top level by supporting the dispensing and administration stages of the process. By far, computerized physician order entry (CPOE) systems are the most well developed tools in the tool chain. Today, CPOE systems are used in a majority of hospitals and clinics in developed countries. Recent data on their effectiveness show that CPOE systems, together with clinical decision support (CDS) and electronic patient health and medication records (ePHR and eMAR) systems [15, 16], can help prevent up to 80% of prescription errors, i.e., 40% of all medication errors.

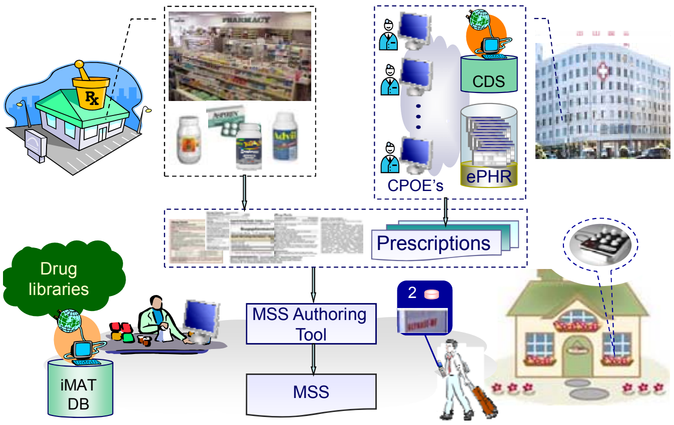


Figure 1 iMAT in medication use tool chain

The MSS authoring tool and iMAT database, shown at the lower left half of Figure 1, are for pharmacists. An essential function of the authoring tool is to merge the directions of all medications of each user and generate from the merged direction a machine readable medication schedule specification (MSS) for the user. As stated earlier, the MSS is needed to guide the operations of user's medication dispenser and schedule manager. We will describe the operations of a version [17] of the tool in the next section.

Another function of the authoring tool is to improve rigor in medication dispensing, which is often a weak link for people on medications outside care-providing institutions. A typical user may be cared by multiple physicians and given prescriptions ordered via independent CPOE systems. While each of the user's prescriptions is error free, it may fail to account for interactions between medications ordered by different prescriptions. An elderly individual is also likely to take OTC and herbal medicines that may also interact with her/his prescription drugs. A function of the MSS authoring tool is to alert the pharmacist about possible *conflicts* (i.e., drug interactions that have not been properly taken into account by some of user's prescriptions and directions). The tool assists the pharmacist to work with responsible physicians to resolve the conflicts if any and generates a MMS when all possible conflicts have been resolved.

A user may have an iMAT medication dispenser for use at home, as shown in the bottom right corner of Figure 1. The medication scheduler that runs on the dispenser can serve as a schedule manager. It delivers reminders to a cell phone and other mobile devices, also shown in the figure. A user may choose to have only a schedule manager and have the tool run on a PC, laptop or a smart phone that can hold the MSS and has network access. These devices have the same purpose as numerous pillboxes and programmable medicine dispensers (e.g., [18-20]) for home use and mobile medication administration tools (e.g., [21, 22]) for use in hospitals and long-term care facilities. Existing pillboxes and dispensers require the user to load the individual doses of medications into the device, understand their directions and program the device to send reminders accordingly. This error-prone manual process and rigid medication schedules are serious disadvantages for users targeted by iMAT.

Intelligent medication advisory tools and services such as MEDICATE Tele-assistance System, Magic Medicine Cabinet, and other medication advice services [23-26] can check directions for drug interactions for users at home. Like schedules used by our dispenser and schedule manager, medication schedules used by these automatic devices and scheduling tools can also be adjusted to compensate for user tardiness and condition changes. The advices and adjustments are provided by care takers who monitor and supervise the user via Internet, however. Those devices are better suited for users who need close professional supervision and fully integrated health care services. In contrast, our medication dispenser and schedule manager are capable of making schedule adjustments permitted by existing prescriptions without requiring their users to incur the costs in fees and privacy loss of close monitoring and care.

III. Assumptions and Rationales

Hereafter, when there is no need to be specific, we use the term *medications* to mean prescription and OTC drugs, as well as health supplements that have non-negligible interactions with some drugs. When food and drink interact with some of the user's medications, we also call them medications in the context of MSS and medication schedules. Although iMAT does not handle food, it must schedule meals and snacks along with medications when they interact.

A. *Illustrative Scenario*

We use here a real-life scenario to motivate the objectives and justify the assumptions of iMAT: 65-year old Mr. Li of Taiwan has three prescriptions [27]. Despite being an avid biker, he had to start taking medications for hypertension in his 50s. The medications prescribed by his doctor for this purpose are listed in the left half of Figure 2(a). In the schedule shown in Figure 2(b), hearts labeled by the first one or two initials of the medications marks the times for these medications: He takes the medications critical for controlling blood pressure at meal and bed times and skips or postpone Acetaminophen (Tylenol) except when he needs it for aches and pains. This schedule is easy to follow even during the around-the-island biking tours which he takes frequently with friends since his recent retirement and move to the scenic east coast. When his latest physical exam at a near-by health management center revealed the on-set of diabetes and elevated cholesterol level, physicians at the center ordered for him the medications listed in the right side of Figure 2(a). Mr. Li's medication schedule now includes the times for the new drugs, which are marked by circles and a teardrop labeled by initials of the drugs in part (b).

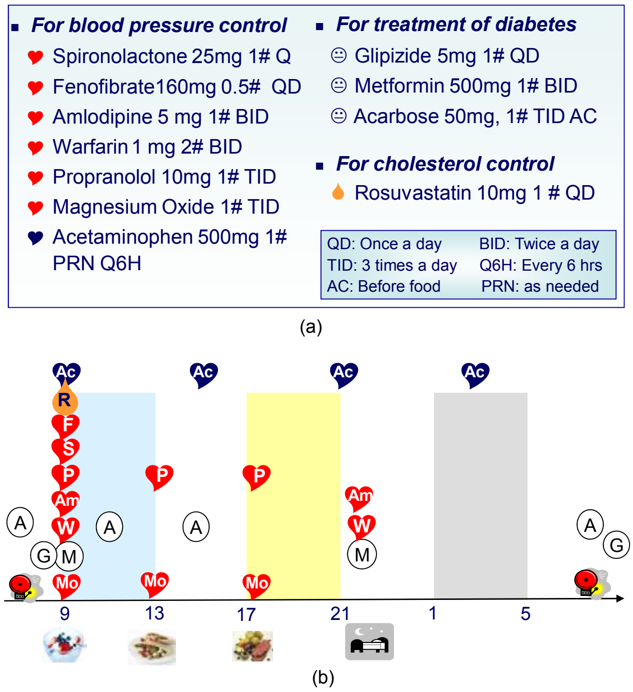


Figure 2 An illustrative scenario

It is easy to understand Mr. Li's frustration when he sets up his new daily medication schedule as instructed. According to the new schedule, he has to get up 2 hours before breakfast for a dose of Acarbose and must, even on his bike tours, take some medications every 2-3 hours. Else, he risks non-compliance. If Mr. Li were an iMAT user, he would not have this dilemma: His MSS would let him take the doses of Acarbose immediately before meals and snacks. The reason is that PDRHealth [8], hence iMAT, requires only that Acarbose be taken "before the first bite of food". He schedule manager would remind him of this rule.

Unknown to Mr. Li, a much more serious problem with his new schedule is that it is not correct. Mr. Li's new prescriptions were ordered without full knowledge of the old prescription. Consequently, no one instructed Mr. Li to keep the daily dose of Rosuvastatin (R) separated from doses of Magnesium Oxide (Mo) by at least two hours. In contrast, Mr. Li's MSS would contain this separation constraint and the schedules generated by his medication scheduler would not have doses of both drugs at breakfast time.

More seriously, no one warns Mr. Li of the fact that because he is on Fenoflibrate and Rosuvastatin, he has a much higher chance of getting a serious muscle-wasting disease. If Mr. Li were an iMAT user, his pharmacist would be alerted of this conflict and would discuss this matter with one or both physicians who ordered the drugs and suggest a replacement. It is possible that Rosuvastatin was prescribed to keep medication direction simple. Knowing that Mr. Li has an intelligent schedule manager, the physician might have ordered an alternative that is safer but imposes constraints deemed too complicated to follow without help. In the case when both physicians decide to use the drugs anyway, Mr. Li would be instructed to alert his physicians of any muscle problem, rather than taking Tylenol to mask the problem.

B. Generation of MSS

From the illustrative scenario, it is easy to see that iMAT cannot be effective for a user unless the user let the system manage all her/his prescription and OTC medications and has access to all his/her prescriptions and directions. We assume that this is the case.

It is also clear that the work of processing the user's prescriptions and generating a schedule specification should be done by a care provider. It is impractical to require individual users to have access to up-to-date drug library and processing tools needed to do work or to be able to resolve conflicts when they arise. For sake of concreteness, we assume here that the care provider is the user's pharmacist. Other possibilities include pharmacies designated by hospitals that use iMAT dispensers to care for patients on medication regimens long after their releases, or as in Taiwan, pharmacies that fill prescriptions covered by national health insurance.

Figure 3 depicts the parts of iMAT responsible for the generation of MSS. The MSS authoring tool [17] consists of the parts encircled by the dotted box in the middle of the figure. The tool accepts as input user's medication directions and generates as output a MSS for the user.

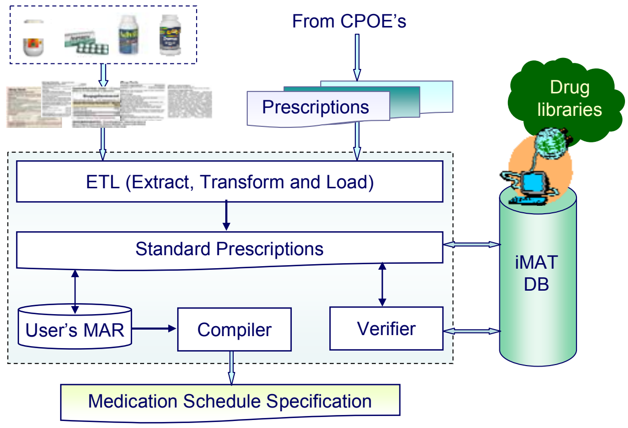


Figure 3 Structure of MMS authoring tool

Specifically, ETL (Extract, Transform and Load) module is responsible for processing the input, consisting of user's prescriptions and directions of his/her OTC drugs, and transforming them into an internal standard form, referred to as *standard prescriptions* here. The need for this component to process natural language directions of OTC drugs with the help of the pharmacist is gradually eliminated by preprocessing directions of more and more commonly used OTC drugs into standard prescriptions and storing them in iMAT database. Increasingly wider adoption of standards and de facto standards for prescriptions from CPOE's also contributes to the significant simplification of ETL from the preliminary version described in [17].

The *verifier*, also called *conflict resolver*, processes one at a time the new standard prescriptions produced by ETL and existing ones from the user's medication administration record (MAR). It checks for conflicts while integrating user specific directions given by user's prescriptions with general directions the tool extracts from iMAT database and on-line drug libraries. If the verifier finds no conflict at the end of the integration process, the processed and merged standard prescriptions are stored in the user' MAR for later use. As the last step of its work, the verifier uses algorithms [28, 29] that the user's schedule manager will use to make sure that the constraints specified by the user's merged directions are *feasible,* i.e., there is a schedule meeting all the constraints. It then invokes the compiler to translate the merged directions into a MSS for the user's dispenser and schedule manager.

When the verifier detects a conflict, it alerts the pharmacist of the conflict and provides the pharmacist with information on the conflicts. We assume here that conflicts are resolved manually by the care providers involved. If some of user's prescriptions are changed as a result, the tool repeats the above described process on the revised prescription(s).

IV. MEDICATION SCHEDULE SPECIFICATION

The medication schedule specification that defines the constraints to be met by user's medication schedule is based on the model described in [30]. In general, for each medication M taken by the user, the specification contains a section consisting of parts extracted by the MSS authoring tool from an XML file on the medication stored in the iMAT database. The section provides general information (e.g., name(s), granularity and picture(s) of the medication and the duration the user is supposed to be on it). The dispenser needs this information to manage and schedule the medication. The section has a *dosage parameters* (*DP*) part that defines firm and hard size and timing constraints for doses of M when the medication does not interact with other medications of the user. If some of the user's medications interact with M, the section also contains a *special instructions* (*SI*) part; this part specifies changes in dosage parameters and additional timing constraints to account for the interactions.

A. Firm and Hard Constraints

We use the terms firm and hard in the same sense as they are used in real-time systems literature. Firm constraints are typically more stringent. The scheduler tries to meet all of these constraints whenever possible. Violations of firm constraints can occur nevertheless, usually due to tardiness or forgetfulness on the user's part. They degrade the quality of the schedule but may be acceptable.

Hard constraints, being less stringent, limit the degree to which medication directions are allowed to be relaxed and schedule quality to degrade. A violation of a hard constraint is treated as non-compliance event and warrants an action (e.g., warn the user, call a designated family member, alert the user's doctor, and so on.) Clearly, the action depends on the medication, the user, and the severity of the violation, and is specified as a DP parameter of M. The scheduler treats user input on preferred times and frequencies for taking medications as soft constraints to be met on a best effort basis. Due to space limitation, we will not discuss soft constraints and non-compliance event handling hereafter.

B. Dosage Parameters

Specifically, firm constraints are defined by the following four sets of parameters. The scheduler computes the normal schedule of M based on these parameters.

- *Nominal dose size range* $[d_{min}, d_{max}]$ bounds the sizes, in term of multiples of granularity of M, of individual doses of M.
- *Nominal separation range* $[s_{min}, s_{max}]$ bounds the length of time between two consecutive doses of the medication.
- *Nominal maximum rate* (B, R) of M constrains the total size of all doses within any time interval of length R to be no more than B. We sometime call R and B the replenishment time and budget, respectively, and the rate the supply rate.
- *Nominal minimum rate* (L, P) constrains the total size of all doses within any interval of length P to be at least equal to L.

Take Mr. Li's Acetaminophen (Tylenol) as an example. Its direction reads "Take one (500 mg) tablet every 4 to 6 hours. If pain does not respond to one tablet, two tablets may be used. Do not exceed 8 tablets in 24 hours." The DP part of this medication has $[d_{min}, d_{max}] = [1, 2]$, $[s_{min}, s_{max}] = [4, 6]$, $(B, R) = (8, 24)$; granularity of time is one hour. The values of these parameters follow literally from the direction. Since the drug is to be taken as needed, there is no required minimum total dose size for this drug; hence $(L, P) = (0, 24)$.

In contrast, Mr. Li takes Propranolol for hypertension. His physician ordered for him one tablet 3 times a day. With such small dosage, it is best that he does not skip any dose, or at most a dose occasionally. This is specified as $(L, P) = (3, 24)$, or more relaxed $(L, P) = (2, 24)$ or $(20, 168)$ (i.e., skip one dose per day or per week).

Hard constraints of each medication M is specified in the DP part by the following two sets of parameters:

- *Absolute dose size range* $[D_{min}, D_{max}]$ says that the size of every dose must be in this range.
- A*bsolute separation range* $[S_{min}, S_{max}]$ says that the time separation between consecutive doses must be within this range.
- *Rate tolerances* β and λ specify the allowable deviations from rate constraints. In other words, $(B+\beta, R)$ and $(L-\lambda, P)$ are the absolute maximum and minimum rates.

Hard constraints being less stringent means that absolute dosage parameter ranges contain the corresponding nominal ranges and β and λ are larger than zero. Indeed, directions of almost all medications provide instructions in case "if you miss a dose". This instruction invariably leads to a wider absolute separation range $[S_{min}, S_{max}]$. As an example, the nominal and absolute separation ranges of a once a day medication are [24, 24] and [12, 48] or [8, 48], respectively, when its missed dose instruction reads "If you miss a dose, take it is when you remember. If it is close to the time for the next dose, skip the one you miss and go back to regular schedule." Our experiments with scheduling real-life and synthetic sample prescriptions demonstrate that the more relaxed separation constraint can make schedules of most medications more tolerant to user tardiness and hence friendlier to the user [28].

C. *Special Instructions*

We refer to a medication (or food) that interacts with *M* to the extent to require some changes in how *M* is to be administered as an *interferer* of *M*. The SI part of *M* has an entry for each of its interferers. The dose size and separation ranges of *M* may need to be changed to take into account of their interactions. Such changes are specified by the *change list* in the entry. The dosage parameters in the change list are in effect as long as the user is on both *M* and *N*.

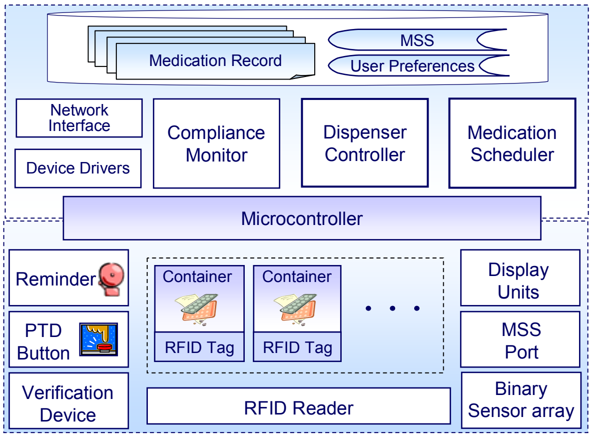
The entry for an interferer *N* may also define additional separation constraints: The time separation between each dose of *M* and any dose of the interferer *N* must be within the specified range: The *minimum separation* $\sigma_{min}(M, N)$ *from M to N* specifies a lower bound to the length of time from each dose *M* to any later dose of *N,* and $\sigma_{min}(N, M)$ *from M to N* is a lower bound to the time from each dose to *N* to any later dose of *M.* Earlier, we mentioned that Rosuvastatin (R) and Magnesium Oxide (Mo) should be taken at least 2 hours apart. In other words, $\sigma_{min}(R, Mo) = 2$, and $\sigma_{min}(Mo, R) = 2$. Another example is $\sigma_{min}(Antibiotic, Food) = 1$, and $\sigma_{min}(Food, Antibiotic) = 2$. These constraints ensure that antibiotic is taken on empty stomach.

V. SMART DISPENSER AND SCHEDULE MANAGER

Figure 4(a) and (b) show a picture of a stand-alone smart dispenser for home use and its internal structure, respectively. As we can see, this dispenser has a memory card reader (i.e., a MSS port) for reading the MSS provided by the pharmacist. The sockets on the base hold medication containers. Each socket is encircled by an indicator light. Other parts that interact with the user include a LED display, PTD (Push-To-Dispense) button, verification boxes, and dispensing cup, as well as an alarm device inside the base for delivering reminders locally. Also inside the base are an RFID reader and an array of switches (i.e., binary sensor array). There is a switch at the bottom of each socket for sensing whether the socket is empty or not. These hardware components are also shown in the dotted box in the bottom half of Figure 4(b).

The top half of Figure 4(b) shows the major software components. The interface to a dial up or broadband connection enables the dispenser to deliver reminders to the user remotely via a phone or a PDA and to send notifications of non-compliance events. The important work done by the device is divided between the (medication) scheduler and (dispenser) controller. The scheduler has full knowledge of what medication administration related actions should be done at what times. It does not keep track of time, however. That important task is done by the controller: The controller keeps track of time, informs the scheduler the arrival of the time instant for each action requested by the scheduler, and enables and monitors the execution of the action. Indeed, the controller monitors and controls the state of the dispenser. In addition to the health of individual components and the device as a whole, the controller is responsible for monitoring user actions, contents of medication containers, etc. and handles events such as user tardiness, insufficient medication supplies and so on that require it to take actions. Hereafter, we sometimes refer to the scheduler, or the controller or both as the dispenser when there is no need to be specific.

(a)



(b)

Figure 4 Exterior and structure of iMAT dispenser

A. *Dispenser Operations*

When a user comes to get new medication supplies, the pharmacist gives him/her an updated MSS in a memory card together with medication container(s). Each container holds one kind of medication, and the medication is identified by the RFID in the tag on the container. The user makes the dispenser ready to manage the new medications along with existing ones by plugging in the memory card into the MSS port and the new containers in empty sockets, one at a time, in any sequence. When the dispenser senses that the open switch for an empty socket (say socket number *k*) becomes close, indicating that a container is just plugged in the socket, it commands the RFID reader to read the tags on all containers in sockets. Upon discovering a new id (say *M*), it creates and starts to maintain the id-location mapping (*M, k*) for the new medication; it will need the mapping to locate the medication. It then locks the container in the socket to prevent unintentional removal.

Set up completes when the dispenser base holds at least a container and the controller has the id-location mapping for every medication listed in the updated MSS. Based on the information extracted from the MSS, The scheduler then computes an initial schedule specifying the dose times and sizes of the medications based on the directions extracted from MSS. Algorithms described in [28] are for this purpose.

Shortly before each dose time, the dispenser sends a reminder locally and remotely to tell the user it is time for some medication. In response, the user reports to the dispenser by pushing the PTD button.

The user may or may not response promptly. When the user is late, the schedule may need to be adjusted. This is why the dispenser updates the dose size of each medication due to be taken when the user pushes the PTD button. For each medication remaining to be due, it turns on the indicator light around the socket holding the container of the medication and unlocks the socket so that the user can pick up the container. When the user picks up a container, the LED display shows him/her the dose size to retrieve from the container. After the user retrieves the dose and puts the container back into the socket, the dispenser locks the container in place again. Thus the dispenser works with the user to retrieve a dose of every medication due to be taken at the time.

A dispenser with the verification capability is equipped with a camera to capture the image of objects placed in verification boxes. The user puts each retrieved dose in a verification box. Once there, the dispenser checks visually whether the retrieved dose size is correct. It uses the display to instruct the user when correction is necessary, and when there is no error, locks the returned container in place and drops the dose in the verification box into the dispensing cup.

B. *Action-Oriented Collaboration*

Limitation in space prevents us from describing the software architecture of the prototype dispenser [31]. It suffices to note here that the dispenser controller is event-driven. The controller uses a pool of worker threads to execute event and action handlers. For this purpose, it maintains several prioritized FIFO queues. When it is instructed by the scheduler to take a specified action or alerted by an event requiring its attention, its wraps the handler in a work item and inserts the work item into the queue according to the priority of the handler. Once queued, the work item is executed by a worker thread at the priority of the queue.

The work to ensure that the right doses of right medications are given to the user at the right times is done collaborative by the medication scheduler and the dispenser controller according to an action-oriented model. In this collaboration, the scheduler is the decision maker, while controller is the one and only action executor. In particular, the controller is purely passive. While it is aware of the time, it relies on the scheduler to specify the time instants for it to query for actions, the actions to be executed at the time instants and so on.

We use the scenario in Figure 5 to explain communication exchanges between the scheduler (decision maker) and the controller (executor) in general and at the same time, illustrate the operations of the dispenser. In the prototype [31], the controller initiates each round of exchange with the scheduler by calling one of the scheduler API functions GetNextAction() and ActionComplete(), passing as an input parameter the current time. As the function names indicate, the controller calls the former to query the scheduler for action(s) it should take at the current time. It calls the latter to report the completion of an action and query for new action(s). Both functions return an action list, which is empty if no action is required, and a future time NHST (next handshake time). The value of NHST set by the scheduler tells the controller the time to query for action again. In response, the controller sets timer at NHST and returns to wait for timer expiration, together with other events it monitors.

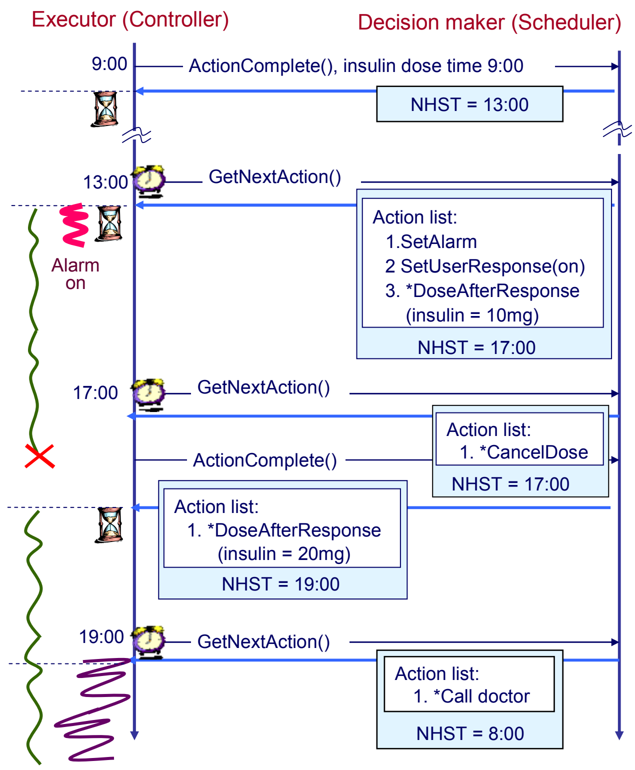


Figure 5 Scheduler and controller communication

In this scenario, the user is supposed to take a 10 mg dose of insulin every 4 hours. If the user is tardy for more than 4 hours, the pending dose is cancelled and a double-size dose is scheduled. Furthermore, MSS specifies that the user's physician is to be notified if the user has not taken any dose for 10 hours or more. The figure shows what have taken place during part of the day:

- At 9:00, the controller reports that a dose of insulin was dispensed to the user and queries for action. No action is required at the time; the scheduler sends NHST = 13:00 only, which is the next dose time. So, the controller sets timer to expire at that time.

- At 13:00, the timer expires and controller queries for action. The action list returned by the scheduler includes turn on the local alarm (i.e., deliver reminder), start to monitor the PTD button and prepare to help the user retrieve a 10 mg dose when the user pushes the button. After it queues the work items for these actions, the controller sets the NHST timer to expire at NHST = 17:00 and returns to wait, while the worker threads process the work items. The threads are represented by wiggly lines in the left side of the figure.

- When the controller wakes up at 17:00 and calls GetNextAction(), the user still has not responded. When processing the query, the scheduler can conclude that the dose scheduled at the current time is still pending from the fact that the controller has not yet reported the completion of DoseAfterResponse action requested at 13:00. Since more than 4 hours has elapsed, the scheduler tells the controller to cancel the pending dose, while it adjusts the schedule according to the instruction from MSS.

- When the controller reports the completion of CancelDose, the scheduler requests that a 20 mg dose be given to the user when the user responds. There is no need for turning on the alarm because it is still on, and the controller is still monitoring the PTD button. The value of NHST returned by the scheduler this time is 19:00. By then, 10 hours will have been elapsed since the user took the latest dose of insulin.

- At 19:00, the user still has not come to push the PTD button. The scheduler requests that the controller calls the designated care taker to report the non-compliance event. The value of NHST is 8:00, the time for the start of the next day. In the meantime the 20 mg dose of insulin is still pending. The wide wiggly line on the left side of the figure represents the thread that logs the event and calls the care taker.

C. *Schedule Manager Configurations*

An advantage of the iMAT dispenser is that it can help, as much as an IT device can, to make sure that user retrieves the right dose of each medication from the right container when the user responds to reminder and comes to retrieve the medication. When the user is away from home and carries the medications with him/her on the road, the dispenser can provide reminders to the user by sending text and voice messages, with or without pictures of the medications, to his/her phone. This is what the picture in the lower right corner of Figure 1 tries to show. In case where the dispenser is connected to Internet, the user can acknowledge the receipt of each reminder and report his/her action taken as response to the reminder. With the user's permission, the compliance monitor can log all compliance related events. In this way, the dispenser can still carry out compliance monitoring function, though to a limited extent.

Some users do not need or want to have RFID tagged containers and the associated hardware. Widjat, the mobile-phone based schedule manager and monitor described in [29], offers them most of the scheduling management and compliance monitoring functions of the iMAT dispenser. Figure 6 shows two configurations of the portable schedule manager. The dotted box below the devices encircles the software components. Here, the user interface manager plays the role of the dispenser controller. It interacts with the scheduler in the manner illustrated by the scenario in Figure 5 on the one hand. On the other hand, it manages the interface to facilitate the interactions with the user according to the user's preference. Like a dispenser, the manager also maintains locally the user's current MSS and medication record.

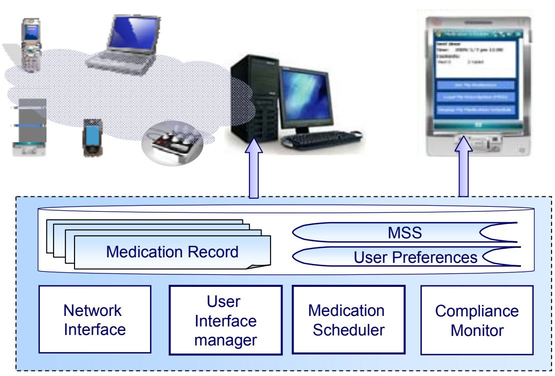


Figure 6 Alternative schedule manager configurations

The smart phone shown in the top right corner is the mobile device used in the Wedjat prototype. It is an essentially stand-alone device that does all the work of scheduling, delivering reminders, monitoring user response, etc. A user may choose to have his/her medication supplies delivered and use the pharmacy service of the supplier for downloading the MSS via Internet. A user who chooses to pick up supplies in person can have his/her MSS loaded to Wedjat by the pharmacist.

The computers and devices in the top left corner illustrate the most flexible of available configurations. In essence, the schedule manager software runs on a PC or a laptop and uses one or more mobile devices for its interaction with the user. A user may start with only these parts. As he/she starts to spend more time at home and take more and more medications, the user can get a dispenser, less the software components, and connect it to the computer as a peripheral device. We have not yet implemented this configuration of iMAT.

VI. Summary and Future Work

We present here an overview of iMAT. Its goal is to help people who take medications without close professional attention to maintain rigor in medication compliance. iMAT medication dispenser and schedule manager are end-user devices. They remove some of the major causes of medication administration errors by computing automatically the user's medication schedule based on a machine readable medication schedule specification (MSS) using algorithms that can take advantage of the flexibility provided by user's medication directions to make the schedule tolerant to tardiness and easy to adhere. They monitor user's response to reminders and adjust dose times and sizes when the user is tardy. In this way, they try to help the user stay compliant whenever possible. In the event non-compliance occurs despite of prevention effort, the tools can deliver alerts and notifications as specified by the user's medication schedule specification.

In addition to the end-user tools, iMAT also provides the MSS authoring tool for use by user's pharmacist. The tool helps the pharmacist process all of the user's medication directions to make sure that all drug interactions have been correctly accounted for before merging the directions and translate them into user's MSS. This tool requires the support of iMAT database. For sake of proof of concept, iMAT database now contains XML specifications of around 150 commonly used drugs. The specification of each drug provides the values of constraint parameters described in Section IV and [30]. For iMAT to be adopted and used in practice, we will need to expand the database to include at least all the medications available in Taiwan. To accomplish this initial goal and to maintain the database in the future, we need techniques and supporting tool(s) to automate the process of translating human readable directions in available drug libraries into XML specifications.

The source code of a dispenser controller prototype [31] on Microsoft Windows XP and medication scheduling algorithms described in [28] have been released under BSD and GPL licenses on http://www.openfoundry.org/en/. They are listed as projects dispenser2 and medscheduler, respectively. We also have a Wedjat prototype. The next step is to redesign and architect the dispenser software so that a user can choose to have any one of the configurations shown in Figures 4 and 6.

Thus far, we have not yet let the tools be used on trial basis by targeted users. This, clearly, is an important step on the way of transition iMAT from research prototypes into real-life tools.

Acknowledgment

This work is partially supported by Taiwan Academia Sinica thematic project SISARL. The authors wish to thank Yu-Chuan (Jack) Li and Frank Kuo for the prescriptions used in Section III and Dr. D. H. Burkhardt for his critiques on the general model of medication scheduling.

References

[1] Medication regimens, causes of non-compliance, Office of Inspector General, US Department of Health and Human Resources, June 1990.

[2] Drug Safety and Availability: Medication Errors, June 2009, http://www.fda.gov/Drugs/DrugSafety/MedicationErrors/default.htm.

[3] Medication reconciliation National Patient Safety Goal, National Patient Safety Goals, Joint Commission of Accreditation Program, March 2009, http://www.jointcommission.org/PatientSafety/NationalPatientSafetyGoals/npsg8_review.htm

[4] Veacez, P. J., "An individual based framework for a study on medical error," *International Journal for Quality in Health Care*, Vol. 18, No. 4, May 2006.

[5] Lisby, M., L. P. Nielsen and J. Mainz, "Errors in the medication process: frequency, type, and potential clinical consequences," International Journal for Quality in Health Care, Vol. 17, No. 1, 2005.

[6] Wertheimer, A. I, and T. M. Santella, "Medication compliance research," *Jr. of App. Res. in Clinical and Experimental Therapeutics*, 2003.

[7] Medication use process, http://www.usp.org/pdf/EN/patientSafety/medicationUseProcess.pdf.

[8] PDRHealth, Drug Information, http://www.pdrhealth.com/drug_info/

[9] CPOE.org, http://www.cpoe.org/

[10] Kuperman, G. J., A. Bobb, T. H. Payne *et al.* "Medication related clinical decision support in computerized provider order entry systems: A Review" *J. Am. Med. Inform. Assoc.* 2007.

[11] D. M. Cutler, N. E. Feldman, and J. R. Horwitz, "U. S. Adoption of Computerized Physician Order Entry Systems," *Health Affairs*, Vol. 24, No. 6, 2005.

[12] R. L. Davis, "Computerized physician order entry systems: the coming of age for outpatient medicine," *PLoS Medicine*, September 2005.

[13] B. Koppel, *et al*., "Role of computerized physician order entry systems in facilitating medication errors," *Journal of AMA*, Vol. 293, No. 10, 2005.

[14] King W. J., N. Paice, *et al.* "The effect of computerized physician order entry on medication errors and adverse drug events in pediatric inpatients." *Pediatrics*, September 2003.

[15] Electronic Medication Administration Record (eMAR). 2009; Available from: http://www.dh.org/body.cfm?id=434&oTopID=434

[16] Health Information Systems: http://www.hhs.gov/healthit/ahic.html, and Baron, R. J., E. L. Fabens, M. Schiffman and E. Wolf, "Electronic health records: just around the corner? Or over the cliff?," *Annuals of Internal Medicine,* 2005.

[17] H. C. Yeh, P. C. Hsiu, C. S. Shih, P. H. Tsai and J. W. S. Liu, "APAMAT: a prescription algebra for medication authoring tool," *Proceedings of IEEE International Conference on Systems, Man and Cybernetics,* October 2006.

[18] e-pill, Pill Dispenser: http://www.epill.com/dispenser.html and MD 2: http://www.epill.com/md2.html

[19] Pill boxes, http://www.dynamic-living.com/automated_medication_dispenser.htm

[20] My Pill Box, http://www.mypillbox.org/mypillbox.php

[21] Rx Showcase, http://www.rxinsider.com/prescription_dispensing_automation.htm

[22] P. H. Tsai, Y. T. Chuang, T. S. Chou, J. W. S. Liu, C. S. Shih, "iNuC: an intelligent mobile nursing cart," *The Second International Conference on BioMedical Engineering and Informatics,* October 2009.

[23] C. Nugent, *et. al*, "The next generation of mobile medication management solutions," *International Journal of Electronic Heathcare,*" vol. 3, no. 1, 2007.

[24] M.Governo, V. Riva, P. Fiorini, and C. Nugent "MEDICATE teleassistance system" The 11th International Conference on Advance Robotics. June 2003.

[25] Murray, M. D., "Automated medication dispensing devices," Chapter 11 in Making health care safer: a critical analysis of patient safety, 01-E58, Agent for Healthcare Research and Quality, 2001.

[26] Wan, D., "Magic Medicaine Cabinet: A situated portal for consumer healthcare," *Proceedings of First International Symposium on Handheld and Ubiquitous Computing* (*HUC '99*), September 1999/

[27] Mobile Intelligent Medication Management Platform Project plan, Section 12, National Yang Ming University, Taiwan, 2009.

[28] P. H. Tsai, C. S. Shih, and J. W. S. Liu, "Algorithms for scheduling multiple interacting medications," *Foundations of Computing and Decision Sciences,* Vol. 34, No. 4, 2009.

[29] Wang, W. Y., J. K. Zao, P. H. Tsai, and J. W. S. Liu, "Wedjat: a Mobile phone based medication reminder and monitor," *Proceedings of the 9th IEEE International Conference on Bioinformatics and Bioengineering*, June 2009.

[30] Hsiu, P. C., H. C. Yeh, P. H. Tsai, C. S. Shih, D. H. Burkhardt, T. W. Kuo, J. W. S. Liu, T. Y. Huang, "A general model for medication scheduling," Institute of Information Science, Academia Sinica, Taiwan, Technical Report TR-IIS-05-008, July 2005.

[31] P. H. Tsai, C. Y. Yu, C. S. Shih and J. W. S. Liu, "Smart Medication Dispensers; Design, Architecture and Implementation," Technical Report TR-IIS-08-010, Institute of Information Science, Academia Sinica, October 2008.