Chapter 14

# Advances in Workflow Modeling

## *The Modeling & Analysis Toolsuite for Healthcare (MATH)*

Keith Butler
*University of Washington*

Ali Bahrami
*Medico Systems*

Konrad Schroder
*University of Washington*

Melissa Braxton
*University of Washington*

Lawrence Lyon
*University of Washington*

Mark Haselkorn
*University of Washington*

## ABSTRACT (150 WORDS MAX)

Synchrony between information flow and the workflow of clinical care is a key principle for health information technology (HIT) to improve the way healthcare can be performed by clinicians. When HIT design decisions are not based directly on better care the resulting application can rearrange clinical workflow by accident rather than by design. We introduce the MATH method and toolsuite to make measurable improvements to clinical workflow a predictable, integral part of HIT systems. We illustrate new evidence-based techniques to analyze how HIT should be applied by prioritizing HIT functionality for beneficial impact. We demonstrate the feasibility to use formative evaluations of workflow models to predict HIT impact, and present validation data from summative tests conducted in situ on an alpha version of the new system. These capabilities allow a new evidence-based approach to HIT in which healthcare leaders decide and plan the appropriate role of computing for their clinics.

# Introduction

Improving the way healthcare can be performed by clinicians is one of the most important goals of health information technology (HIT) [1, 2]. Based on recent analyses the goal is not being achieved on any widespread basis [2, 3]. Two of the most important challenges to HIT success are problems with workflow and usability [2; 4].

This chapter focuses on the synchrony between information flow and the workflow of clinical care as a key principle for HIT effectiveness, quality, and usability that has not been addressed adequately [5]. If the flow of information does not match the needs of an appropriate workflow, then its users are faced with a dilemma: either perform overhead tasks to compensate by modifying their information environment, or follow a sub-optimal workflow that is constrained by the way their HIT applications currently provide information [6]. Unplanned overhead is more than just extra work that interrupts users in the performance of care; it can disrupt their cognition, inhibit coordination among team members, and even obscure their understanding of tasks [7, 8]. It is a form of usability problem that places ease-of-use in opposition to patient safety [9, 10]. A well-designed HIT application with good usability will make the routine performance of safe, efficient and effective care procedures the easiest course of action for users.

When design decisions are not related directly to the values of better care, HIT design can be dominated by issues of technical feasibility, schedule or cost. The resulting application can have the unfortunate effect of rearranging clinical workflow by accident rather than by design [11, 12]. Conversely, we believe that credible, understandable evidence about measurable benefit to care will result in increased adoption and productive use of HIT.

To close this gap we developed a comprehensive new design paradigm called MATH[1] to integrate three fundamental elements that are currently disjoint: workflow models of how clinical care is actually performed and the areas where improvement is needed; the options for how HIT should improve workflow in measurable ways, and; software specifications for rapid implementation of data structures, information flows, use-cases, and user interface concepts.

The MATH method is supported by a suite of tools.

- MATHflow for capturing, analyzing and integrating workflow and information flow
- An information dictionary that is created while modeling a workflow in the MATHflow tool
- MATHsim for discrete-event simulations and formative evaluation of HIT options

In this chapter we will describe and illustrate how these advances in workflow modeling enable an evidence-based design approach for HIT (EB-HIT), and then demonstrate the feasibility of applying the method to make measurable, predictable workflow improvement integral to the design of HIT systems.

## *Cognitive Science Principles*

1. Modeling & Analysis Toolsuite for Healthcare (MATH)

The term "information system" is something of a misnomer because its work is actually executed not only by computers but also by the cognitive and manual procedures of human users. Further, HIT applications typically play a support role as they are used in a clinical workflow with many important manual tasks of care and administration.

Integrating manual and computer-performed work is critically important for HIT effectiveness and usability [11]. Information resources, like many other types of resources, constrain the way clinicians can use them to perform care. Research from cognitive science [7, 8] and software design [12, 13, 14] consistently demonstrates that the content, organization, and representation of information inherently impose powerful constraints on the way users are able to perform their tasks. The constraints' are widespread and powerful. They affect users' procedures and even their strategies for performing cognitive tasks [8]. Cain and Haque described how HIT implicitly imposes workflows on nursing [15]. White and Miers argue that the software of an information system actually embodies a model of workflow, whether or not that workflow was understood and planned [16]. Further, unplanned tasks may be added to the workflow in an accidental manner when users have to deal with design-induced errors [10, 17] that are caused when the properties of information do not match the needs of a task.

HIT developers may be reluctant to accept responsibility for constraining the way clinicians will be able to work with their applications, but attempts to take a neutral design stance are mistaken. Information resources, like many other types of resource, constrain the way they can be used to perform work. Constraints on information-dependent work are inherent to information resources, including complex HIT applications. The only question is whether the way an HIT application constrains clinical workflows will be understood and planned as part of systems design, or risk that its impact will happen largely by accident after it has been deployed.

## APROACH

Our method for design integration exploits two established software standards: the Business Process Modeling Notation standard for modeling workflow [16]; and the class and state diagrams of the Unified Modeling Language to model the information architecture of the HIT application [18, 19]. Originally popularized for web design, information architecture is a powerful, non-visible dimension of usability in interactive systems on mobile and client platforms, too. Information architecture is a key part of software design that defines an application's body of content or data, and how it is organized for end-users [20].

Previously, the best way to reduce the risk of unpredictable HIT impact was to make very conservative, incremental improvements to existing information systems. While easy to understand, this cautious approach can easily fail to exploit the full potential of HIT or achieve the benefits that justify significant cost. Our use of standards enables a method to develop well-defined models of workflow and information architecture. As the paired models are being developed we can understand the implications of design decisions for one on the other, so their respective designs can converge. MATH's high-level design goal is for workflow and HIT to function as a pair of well-matched, complementary components to improve the performance of a larger care system in a predictable manner.

### *MATHflow*

The core capability is MATHflow, a visual diagramming tool to capture the existing workflow and use of current information resources in such a manner to reveal how it should be improved with better information resources. There is growing interest in workflow models as a tool to analyze and design HIT [21, 22, 23, 24, 25, 26. Carayon, et al., argued that HIT necessarily involves workflow, but little is known about how to integrate the two [25]. MATHflow allows us to analyze how an HIT application will impact workflow by replacing, augmenting, or complementing important manual activities of care, then evaluating the resulting model.

MATHflow is different than earlier tools for diagraming clinical workflows. As we will illustrate, MATHflow integrates modeling for workflow, the use of physical resources, and the use of information resources. MATHflow distinguishes the access properties of different types of information resources (e.g., paper records vs. EHR. These capabilities allow the MATH method to make trade-offs between better information resources for less need of physical resources, such as the labor of highly skilled clinicians. These analyses provide valuable, formative evidence about the HIT impact during the design stage of a project to guide decision making about HIT functionality.

MATHflow is independent of any specific EHR because it is based on the Object Management Group's recent standard for Business Process Modeling Notation (BPMN) [16]. The standard for BPMN has been rapidly adopted in numerous diagramming tools for software requirements of applications to support manual work by teams of people. MATHflow is a partial implementation of BPMN 2.0 with innovative extensions to the standard for integrating information modeling with workflow modeling.

The MATH method enables researchers, working with care stakeholders, to capture workflow improvements and connect them to HIT design. This connection enables conscious design trade-offs between the added value of an HIT system, in terms of its impact on the quality or efficiency of clinical workflow, and the factors around HIT technical implementation, such as feasibility, risk, cost and schedule. MATH addresses three of the Institute of Medicine's dimensions of quality: Patient-centered, Efficiency, and Timeliness [27]. MATH closes the gap in conventional methods by integrating the capture of how clinical care is actually performed, the options for how it could be improved in measurable ways if supported by better HIT applications, and the algorithms, data, and user interface concepts of those applications. An ancillary benefit of MATH's integration capabilities is to close the gap between the design of workflow and the design of supporting HIT software, thereby reducing cost and time for software development.

## *The MATH Method*

The overall objective is to understand how care is performed currently, including the cognitive models of EHR users and the contextual constraints that produce significant problem trends, and how the problems can be mitigated or improved, with focus on better designed HIT. The MATH method is summarized in figure 1. The first step of MATH is similar to popular process improvement methods [28] but the rest of MATH is a variant of "concurrent engineering" in which multidisciplinary teams collaborate on a common design objective [29]. MATH integrates "patient-centered design" with conventional "technology-centered design" in an iterative method. The objective is to produce a pair of matched designs that will operate smoothly together as: 1) a measurably better workflow of care, and 2) a cost-effective, highly usable HIT application whose information flow maps clearly and efficiently to the needs of the better workflow.

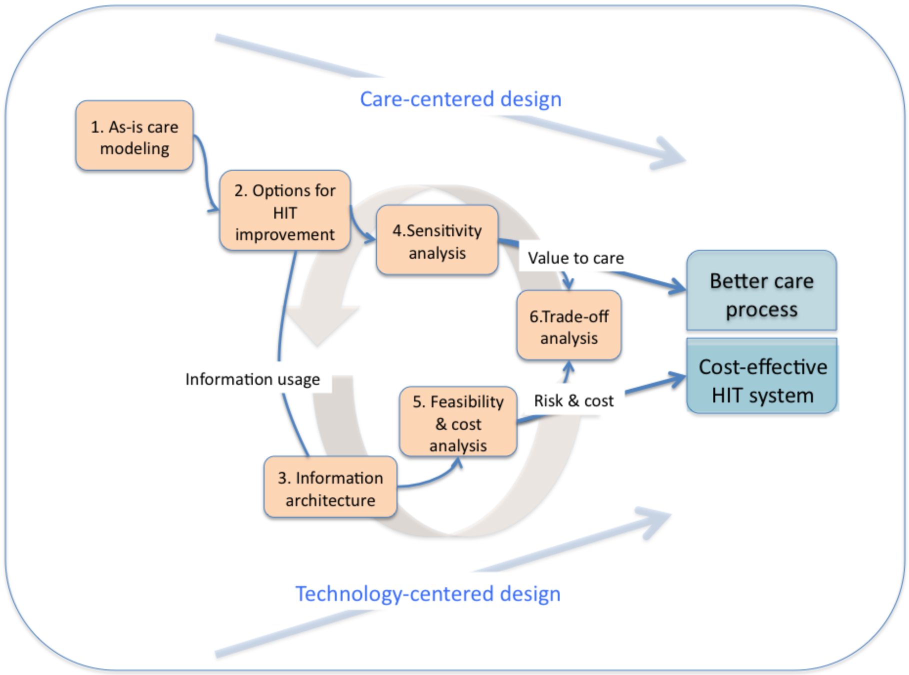


**Figure 1: Flow of steps of the MATH method**

Step 1 is an observational study that applies ethnographic methods to discover, model and understand the way clinical care is currently performed with existing information resources, record a dictionary of the information required to support it, and identify awkward workflows and recurring problems. In the most general terms, if we do not understand how care is performed currently then we cannot reliably analyze and design cost effective HIT improvements.

Step 2 is an analysis of HIT options to address problem areas that appear to be caused by information resources in which the organization or sequence of information does not match workflow, the presentation format does not match a task, the use of information is degraded because diverse resources must be integrated manually, or excessive user attention is required to control existing HIT functionality.

Steps 3, 4, 5 and 6 will progress through several iterations. In step 3 we analyze the information architecture for each option. The information architecture defines the content and organization of a body of information needed to support the workflow. It can be readily translated into key software specifications using the Unified Modeling Language (UML) for software [18, 19]. This step establishes a clear connection between the workflow benefits of HIT and the technical feasibility and cost of software to actually provide it.

Step 4 uses formative evaluation to compare at least two options: “as-is” and a proposed “to-be.” A key principle of evidence-based HIT is that computing functions should be prioritized on the basis of their impact on better care. The evidence of improvement should be the benefits to workflow, such as gains in the quality or efficiency of care.

In our experience, user participation in this step can quickly generate ideas for several "to-be" options. We organize these into several coherent options from steps 2-3 and estimate the dimensions and magnitude of improvement for each. The results provide valuable formative evidence about beneficial impact on workflows of clinical care.

Step 5 will consider each HIT option's technical difficulty, time to availability, and cost. These are the typical factors of decision making in conventional design. They are important, but must be weighed against their impact on the efficiency and quality of care to select the best option.

Step 6 transitions responsibility for prioritizing the options to the clinic's leadership, stakeholders, and IT staff. We facilitate their analysis to rank order the options, based on their values for patient care. The analysis weighs the trade-offs of three factors for each option: 1) the value of care improvement; 2) the technical risk; 3) and a cost estimate for the project to acquire or implement each option.

The principles we have incorporated into MATH provide a methodical, understandable means for stakeholders to direct strategic, cost-effective improvements to workflows [12, 13]. This remainder of this chapter describes a feasibility study of the MATH method and tool suite to illustrate how increased insight into the workflow and information flow of a large primary care center can result in a predictable, measurably better workflow.

## *Feasibility Demonstration*

In collaboration with VHA Medical Informatics, we used the MATH method, tools and techniques in a study at a primary care clinic in the Puget Sound region. Working with clinicians, our team of analysts followed the MATH method to:

- develop an *as-is* workflow model of how care was actually practiced using existing information resources;
- analyze how it should be improved with HIT;
- perform formative evaluations to predict the HIT impact on a *to-be* workflow;
- build and test an alpha version of the software to compare summative empirical results with the formative predictions.

The modeling and analysis part of the study took about sixteen hours of semi-structured interviews and observations with providers and nurses, and thirty hours for model-building, analysis, and design. The design was then implemented in an alpha version of software, and tested in a summative evaluation that took another twenty-two hours of subject interviews, and twenty hours of software testing by subjects.

Figure 2 is a screen shot of a MATHflow model for current practice around patient visits to a primary care clinic in the Puget Sound region. Evidence-based HIT requires an understanding of the way clinicians currently perform care, so improvements can be identified, prioritized, and incorporated into design of a better system. This workflow begins when the patient enters the clinic at upper left. Each patient follows one of the optional paths defined by decision gates (diamonds), where the various paths through the workflow are decided. Decision gates can select outgoing flows based either on probabilities or the logic rules assigned to them.

MATHflow uses the colored rows in figure 2 as swim-lanes to organize the visual layout of various job types in the clinic. In the *Receptionist* lane there is one task activity to *Schedule new appointments* (top right) and one sub-process for *Check-in patient* (top left).

Sub-processes have a small cross at the bottom-middle, indicating they are made up of a lower level flow of activities or sub-processes. For example, in the swim-lane for PhysicianMD there is a sub-process for *Continue assessment-plan*, which waits at the message symbol until test results arrive.

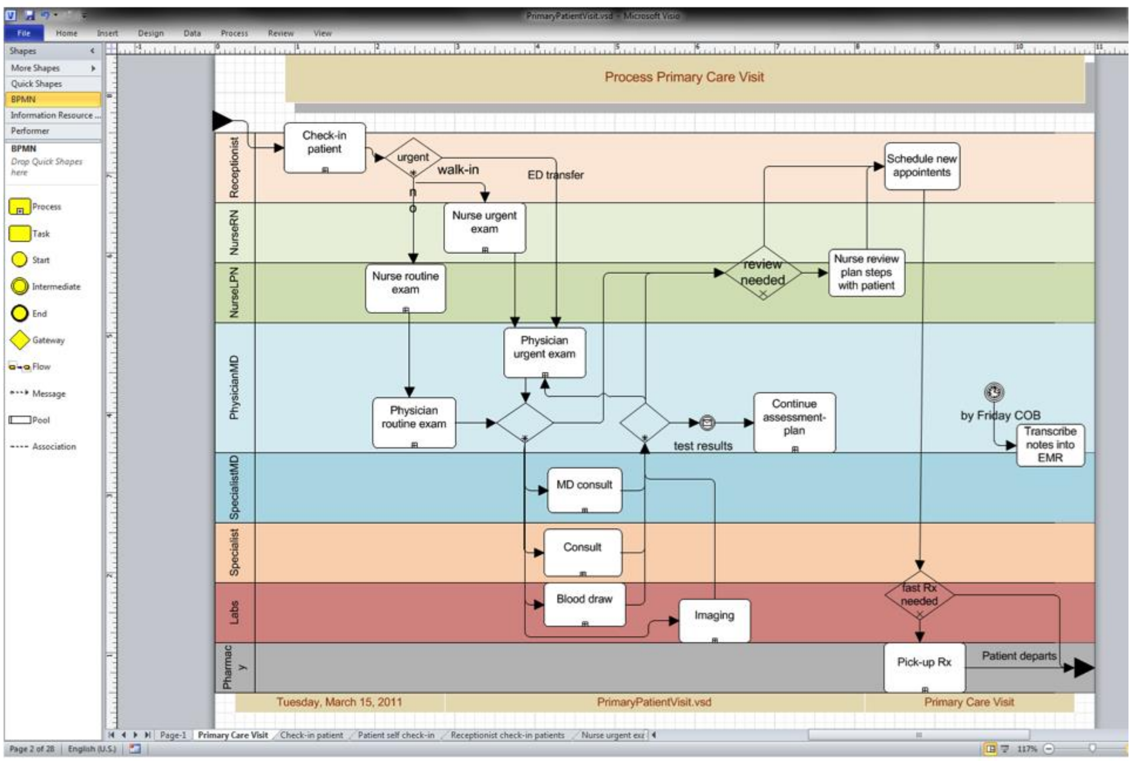


**Figure 2: High-level workflow of primary care**

In figure 3 the sub-process for *Continue assessment-plan* has been opened to display more detail of the workflow it contains. It has several activities at the task level, and also several more sub-processes, which in turn can be opened for more detail. Workflows necessarily reflect many factors, including the type of patients and their care, the nature of the work entity, the personnel and organization of the clinic, facilities and equipment, regulations, and clinic policies. Managing detail with hierarchies of sub-processes is one feature that allows MATHflow to represent large, complex workflows without displaying an overwhelming amount of detail all at once.

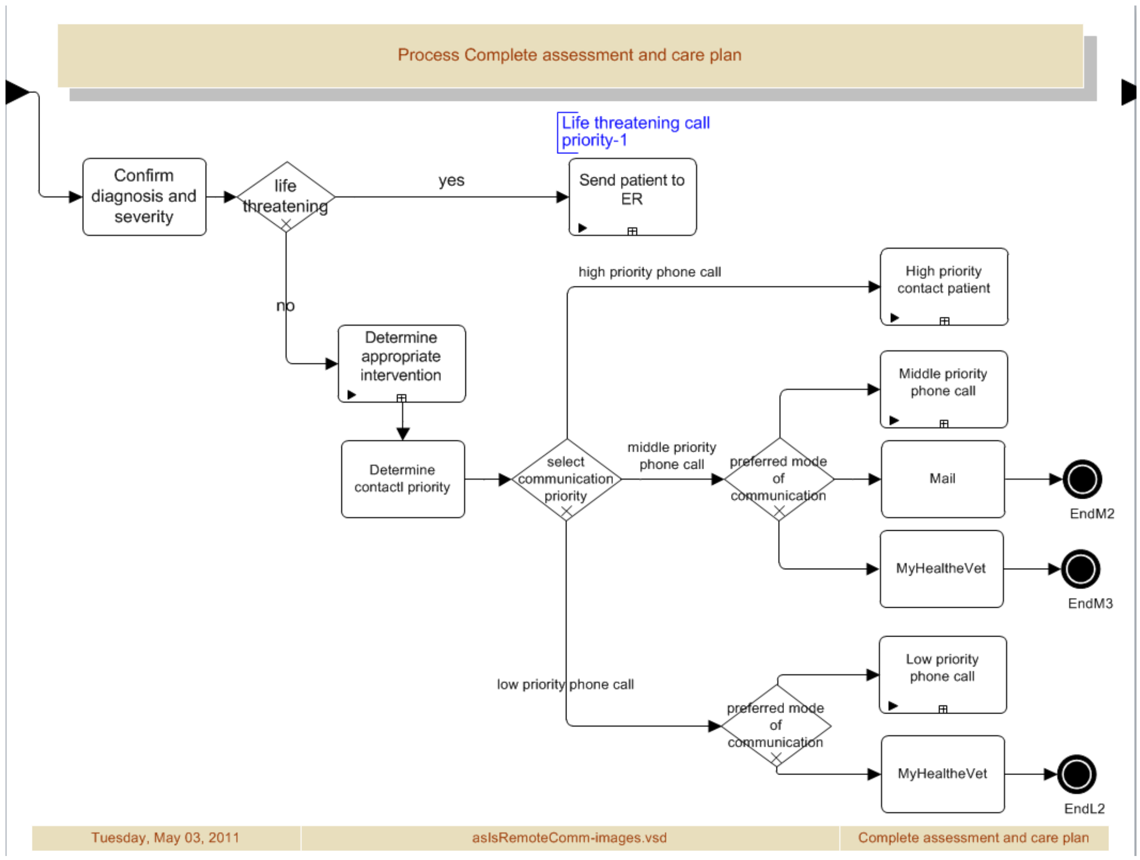


**Figure 3: Detail for the sub-process for Continue Assessment -Plan**

Figure 3 shows a sub-process where a provider has received the test results, and is beginning the tasks of the sub-process by confirming the diagnosis and severity. The workflow reflects clinic policy for contacting the patient depending on the severity. The diamond shaped life threatening is a decision. In the rare event that the diagnosis is life threatening, policy requires the doctor to immediately contact the patient to go to nearest emergency room. Otherwise the flow continues to *Determine appropriate intervention*, where treatment is planned or further tests ordered. VHA policy requires that patients must be informed of test results. The workflow in the right half of figure 3 reflects the three non-critical priorities and the permissible means within each level for contacting them. This workflow was currently carried out manually at the cost of about forty hours a month per provider. According to interviews only about 33% of phone calls actually reach the patient directly. The resulting churn and phone-tag made this workflow a candidate for efficiency improvement.

### Scoping Sub-processes

A new sub-process is needed when the entity of clinical work changes. For example, in figure 2, upon the patient's arrival the work entity was a visit registration. Other sub-processes in figure 1 have entities for patient exams, treatment plans, and lab tests. These are distinctly different and the scope of each sub-process must account for the transformation of the work entity to its goal state. For example, the sub-process for a lab must account for how the entity of a blood draw is transformed to the goal state, which is a lab test report. Importantly, MATH can also model and analyze HIT support for work entities that are conceptual, such a diagnosis, or a treatment plan [30].

A workflow of care is also constrained by resources, including information resources. The purposes of the initial *as-is* MATHflow model of the clinic were to capture how care is actually performed and to understand how it is constrained by the context and available resources, including both physical resources and information resources, as shown in figure 4. MATHflow can capture not only workflow but also the information and its flow needed to support care. Our focus is to understand current workflow to determine how HIT should improve care. MATHflow can model physical resources, such as the highly skilled labor of clinicians, and the way their availability constrains workflow. But we are primarily on workflow at the level of detail where information is accessed, used, changed and recorded. We then apply that insight to achieve better synchrony between workflow and information flow that will result in the improved performance of care. One important type of improvement is more efficient use of the time of clinicians, another is better quality.

**Capturing Information Flow**

The flow of information is not identical to the workflow of clinical care. The entity that flows through much of primary care is the patient. In contrast, information resources contain information about the patient, tests, diagnoses, treatment plans, etc. Their information contents flow in and out of care activities, which can change their values. Further adding to the complexity, the patient may act in multiple roles: the entity of care; an actor in his own care; or an information resource on other occasions.

Most diagramming tools for software are aimed at creating elegant design solutions, as opposed the complex, and often informal way that healthcare is actually performed. MATHflow has a representation for information that is distinct from, but related to clinical workflow, to provide the flexibility needed to capture the way care is actually performed with information resources. It is also important to model all the information resources that are used in a clinic, whether they are physical or electronic. In a clinical environment the information resources may include media that are paper, digital, mechanical equipment or instruments. Information resources also often include people, adding complexity that may overwhelm conventional modeling languages. Doctors and staff may play the multiple roles of labor resource or information resource. Further adding to the complexity, the patient may have multiple roles as the entity of care, actors in their own care; or an information resource for clinicians. This principle requires a representation for information that is distinct from, but related to clinical workflow.

The information modeling capability in MATHflow allows it to represent large, complex workflows without displaying an overwhelming amount of detail. By treating information as a resource (instead of a task) the models are visually simpler, while capturing complexities of the interaction between computing functions and manually performed functions. Figure 4 shows how information resources needed for *High priority contact patient* are captured with the MATHflow properties editor.

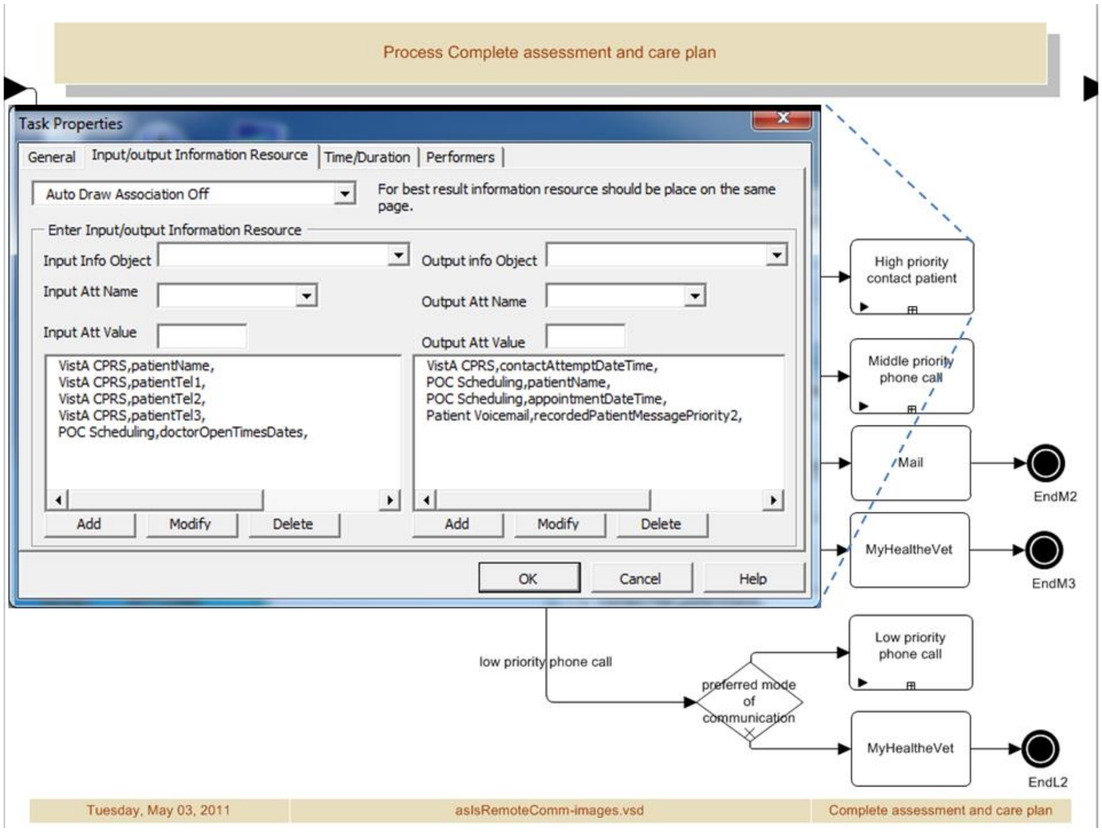


**Figure 4: The information use editor**

In Figure 4, the *Task Properties* editor tab has been opened for Input/output Information Resource. On the left panel the information needed to contact patients with high priority have been entered in terms of the attribute name and the resource object where it was accessed. Similarly, the right panel shows the outgoing information attributes for logging the call into CPRS, leaving a priority-2 message on the patient's voicemail, or scheduling an appointment. The editor captures information use regardless of the medium of the resource.

There are several important reasons to capture all the types of information resources that are used in a clinical workflow, whether digital or not. Some physical information resources may be good candidates for an improved HIT system, there may be overlapping records in different media that need better configuration management, and the integration of physical and digital resources can be a source of inefficiencies to correct in a new system.

Until recently workflow models captured information requirements in an ad-hoc manner. For very small incremental effort, however, MATHflow captures the use and flow of information in a manner that integrates it with activities and their flow. Through the use of the information property editor, as the analyst builds the workflow model the editor also captures a dictionary of all the information attributes that the workflow tasks need (and only the information that they need) and their flow as well. MATHflow automatically builds the dictionary from the entries in the properties editor. Table 2 shows the concept of the information dictionary.

Information attributes

| | 1 | 2 | 3 | . . . | N |
|---|---|---|---|---|---|
| Task activity | | | | | |
| A | 0 | 1 | 1 | 0 | 1 |
| B | 1 | 0 | 0 | 1 | 0 |
| C | 1 | 0 | 0 | 1 | 0 |
| . . . | 0 | 1 | 1 | 0 | 1 |
| Z | 1 | 1 | 1 | 0 | 0 |

**Table 1: Concept of the MATHflow information dictionary**

The information dictionary depicted in table 2 lists each task in the workflow down the left column and the information attributes needed by tasks across the top. The resulting matrix captures information usage patterns needed by the workflow. For example, task-A does not need attribute #1, but it does need attributes #2 and #3, and so on.

The information dictionary also enables another technique for managing the complexity of models. MATH's implementation of Complex gateways differs from the BPMN standard, since the standard does not share MATH's concept of information resources. MATHflow has complex decision gateways that can check the value of an information variable in the dictionary, and then select which outgoing flow to take based on its value. This technique allows activity in one sub-process to determine the behavior in another with having to draw long flows across pages. It provides an technique to capture computer-mediated communication.

As we will show next, the way the dictionary captures the body of required information, along with where in the workflow each attribute is needed, establishes a key connection between HIT improvement of a workflow and the design of the software needed to enable it. This connection allows us to analyze and understand how an as-is model of workflow can be improved by changing the resources that contain the information. This innovation has the added benefit of representing both manual tasks and HIT tasks in the same notation, which is key for integrating HIT into workflow improvements. Rather than focusing on the features of HIT the application is treated as a resource that supports or performs tasks.

**Analyzing Options for Improvement**

An important result of our preliminary study of the clinic was an understanding of the *as-is* workflow that identified a significant problem area and lead to the design of measurable improvement. An additional important understanding of the demographics and the context in which care takes place revealed important factors that affect HIT requirements. The VA already offers internet accounts to MyHealthEvet for web-based access to veteran health benefits and services. But clinic personnel estimated that only about 10% of their patients use it. In addition, the patients of this clinic typically have comorbidities and complex treatment plans. Consequently, providers want to talk with patients to check how well they will be able to carry out new orders. Another factor is patient preference: recent surveys show a strong preference for real-time phone conversations with providers when new orders are issued [31, 32].

In figure 5 the new workflow that is enabled by the new HIT application has replaced the unproductive activity (right half of figure 3) with an software product named *Priority Contact*, which won an award in the national challenge competition SMART Apps for Health [33]. *Priority Contact* was initially designed to interact with EHRs by reading and writing data via the new SMART Connect interoperability standard, but runs on its own, separate web server to allow maximum functional flexibility.

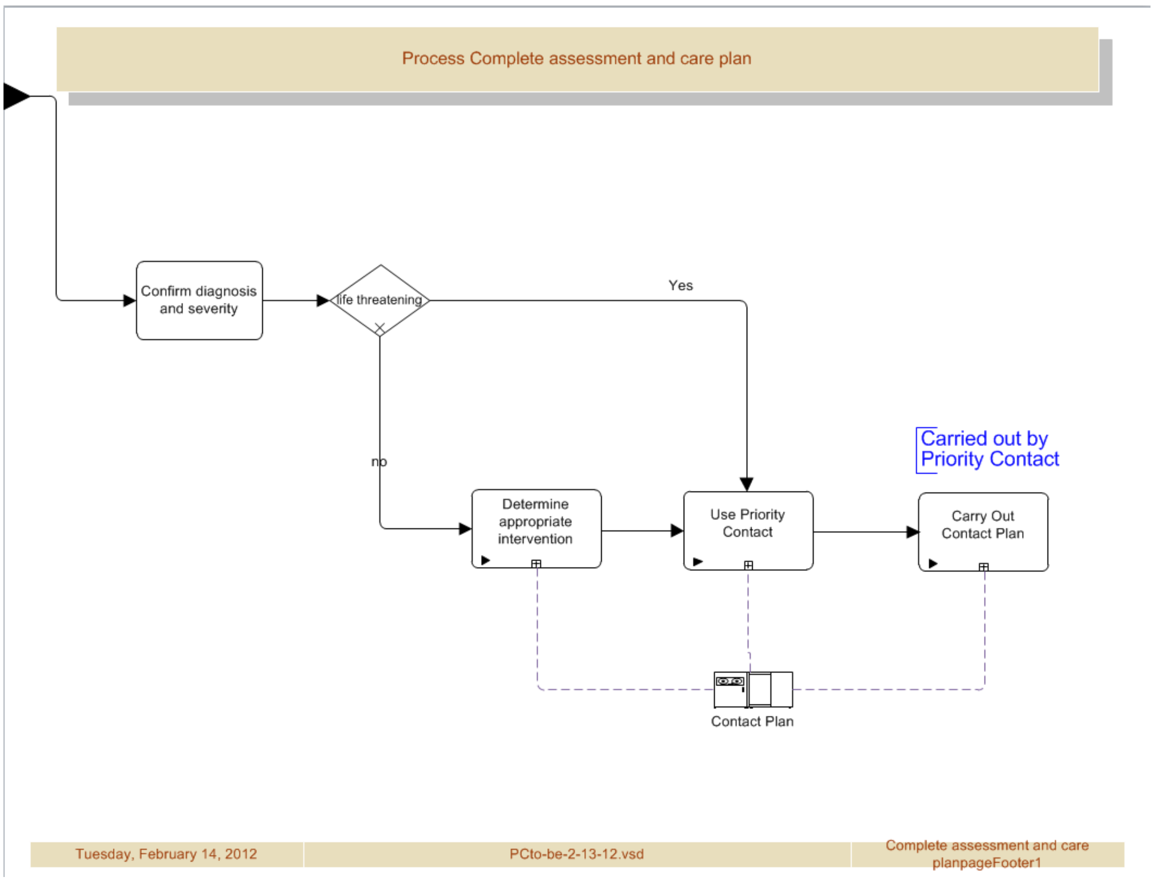


**Figure 5: More efficient workflow with new HIT resource**

A key tradeoff enabled by MATH is better information resources for less use of physical resources, such as the work time of highly skilled clinicians. The streamlined workflow in figure 5 has replaced the large amount of unproductive activity for contacting patients manually. In exchange for the large reduction in effort the clinician must only perform a new small task to set up a *Contact Plan* for each patient, which is then carried out by *Priority Contact* in a manner that integrates it with patient activity and the remaining manual clinician activity.

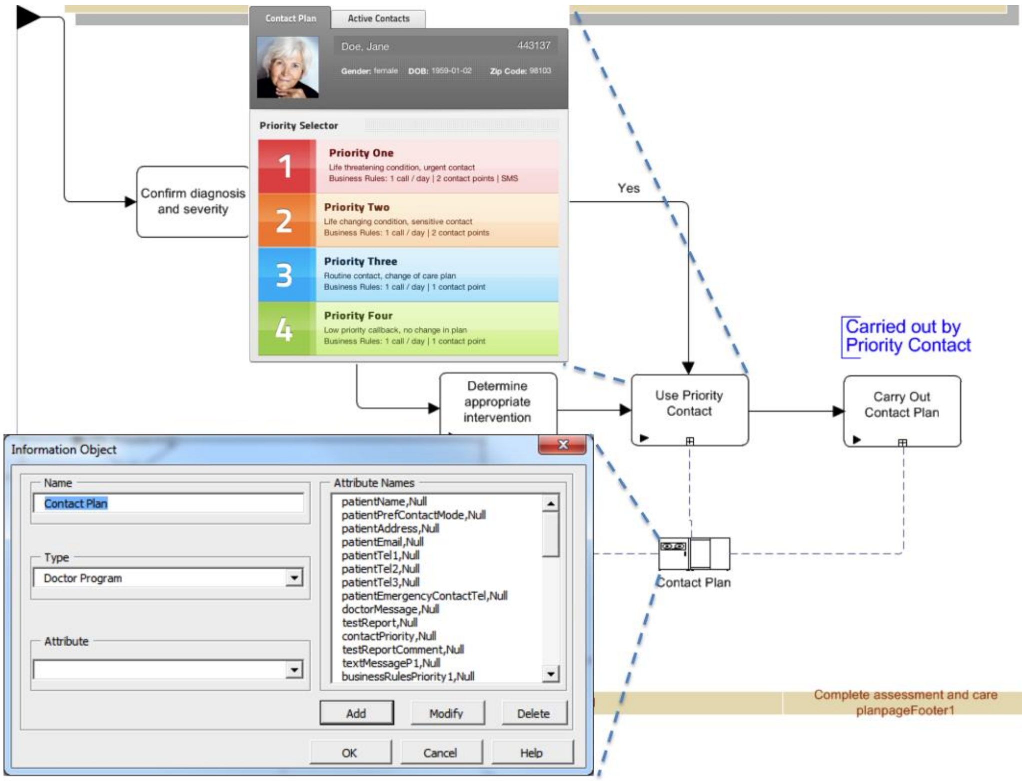


**Figure 6: User interface and contents of the contact plan**

The *Contact Plan* waits in the background until the doctor enters a patient's identity and contact priority via the user interface depicted in figure 6. The user interface for setting up a contact plan is shown above. Another benefit of design integration is that the user interface can be simplified greatly when it is based on a specification of the desired workflow and information flow. To start the new patient contact workflow a clinician simply reviews, edits, and launches the plan, which reads the EHR for the remaining information it needs and then provides the data needed to carry out the algorithm shown in figure 7.

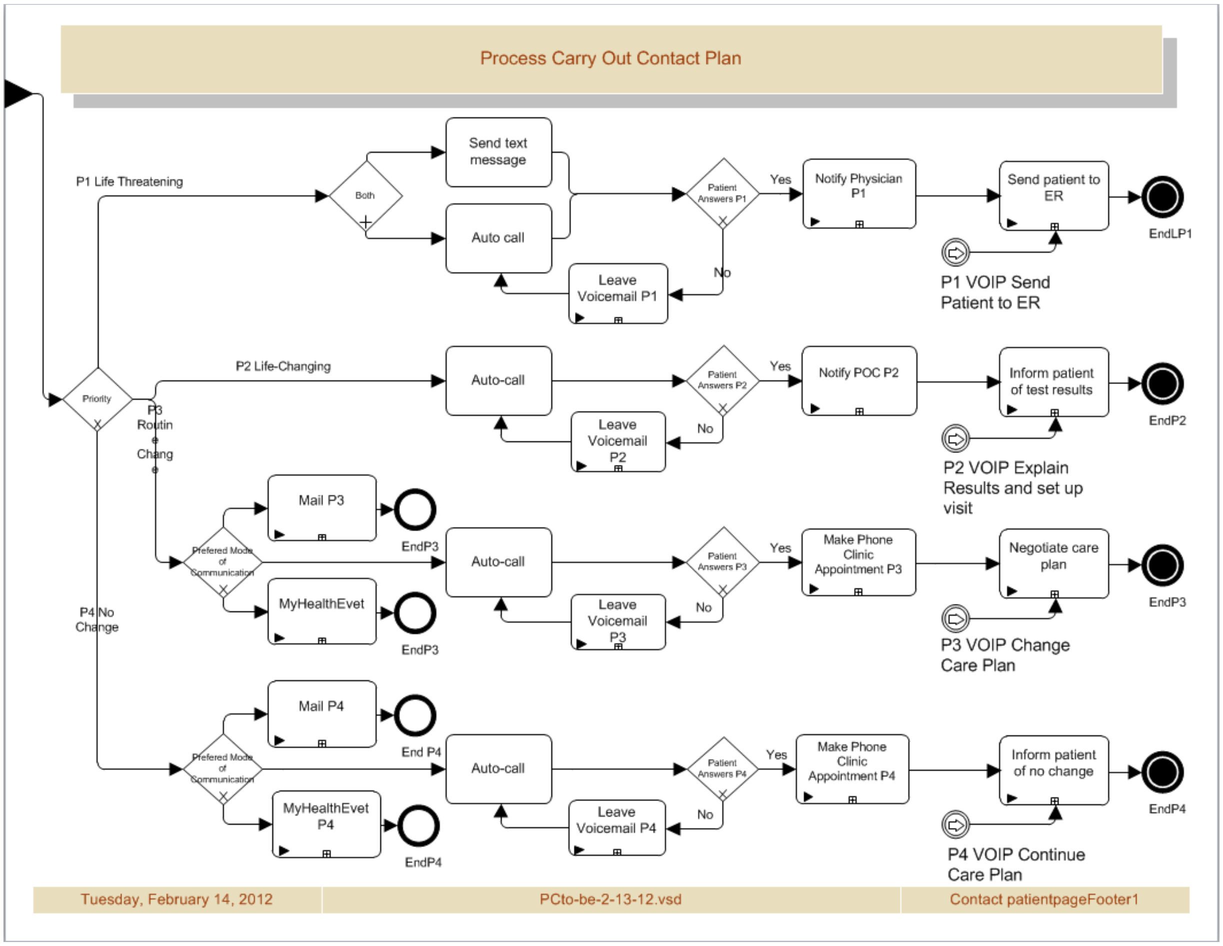


**Figure 7: Partial Model of the *To-be* Workflow**

In figure 7 the as-is workflow of manual activities has been translated into an algorithm that is now carried out by the patient, clinicians, and the software agent. *Priority Contact* does not replace important real-time conversations between clinicians and patients (represented by the four sub-processes on the far right). It enables them to happen more quickly, with less frustration and wasted effort. An additional benefit is that a single system can handle urgent, priority-1 contacts by calling multiple phones simultaneously and repeatedly until the physician is notified that one of the specified phones answers or calls back.

## Principles of Formative Evaluation

In order for HIT to benefit clinical care reliably developers need guidance during the design stage of HIT projects. The importance of evaluating HIT applications for their impact is well established [34, 35] but conventional methods require some version of the software to exist in order to evaluate its impact. Until recently it often meant conducting evaluations after the major design decisions of an application were already decided, and the results of could not be applied until work on the next version of the software began. Little could be done to improve any fundamental design problems during in the intervening months or years while clinicians dealt with unplanned overhead and risk of errors.

From a practical standpoint evaluation needs to provide feedback on HIT design before a project has reached the point where it is too expensive or difficult to make major changes. The growing use of usability evaluation like TURF is an example of how formative evaluation can provide timely feedback to improve user interfaces (5). MATH's focus on achieving synchrony of workflow and information flow for clinicians working teams complements TURF's deep focus on individual user interface designs.

The integration of workflow with information resources and physical resources enables MATH to perform formative evaluation of the impact of HIT on workflow to support major design decisions. By moving the evaluation of workflow impact much earlier in the HIT life-cycle MATH can "build-in" the needed benefit through model-based design iterations.

**Formative Evaluation for Quality**

In the current study clinicians were able to understand and critique MATHflow diagrams with minimal training. The algorithms in figure 7 were derived from the as-is workflow. They accomplish the same purpose but far more efficiently by integrating HIT functions and automation with manual tasks. Exhaustive, qualitative evaluations of options for the *to-be* workflows were conductive using the cognitive walk-through technique [36]. Clinicians and developers worked in joint sessions to review every path and drill-down into sub-processes for correct conformance with appropriate care, efficiency, and feasibility. As shown earlier in figure 6, user interface images could be linked to tasks in MATHflow to make the evaluation more tangible.

An important patient safety criterion was that the workflow must allow clinicians to maintain positive situation awareness until the need to contact the patient is resolved, e.g., "What should happen if the patient does not have voicemail, or nobody checks it?" When either a provider or a developer recognized a problem they negotiated by suggesting other options for solving it, then checking that it would work from both perspectives.

Another key part of the walk-through was the flow of information, as depicted in figure 8. The integration of human tasks and those performed by *Priority Contact* was mediated by the common use of information objects: i.e., both HIT and the human users operate on the same information objects, but with a more appropriate allocation of responsibility.

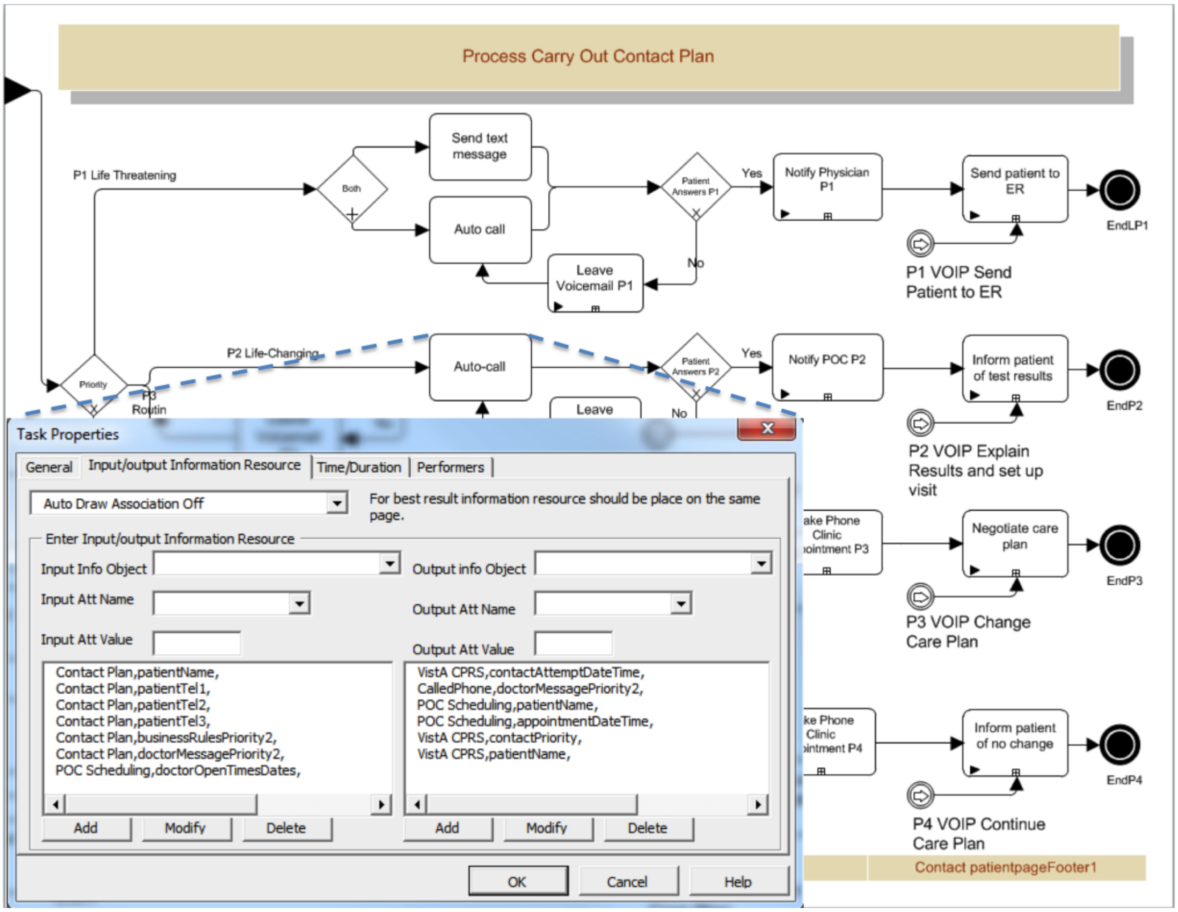


**Figure 8: Information Used by Tasks in the *To-be* Workflow with Priority Contact**

Figure 8 illustrates how MATHflow makes the information requirements explicit for both clinicians and developers to review how the design of *Priority Contact* will work from their

### Formative Evaluation for Efficiency

In addition to *Input/Output Information Resource*, the property sheets of tasks also include tabs for *Time/Duration* and for the *Performers* of a task. When combined with the information needed for each task they allow MATHsim to analyze an important trade-off: less use of labor resources for access to better information resources.

MATHsim is a discrete-event simulation [37] that is integrated with MATHflow. It reads models from a MATHflow database, and performs Monte Carlo simulation to measure the performance of the models under user-supplied workloads, allowing multiple models to be compared against one another. The results are quantitative distributions of task times and resource usage. MATHsim results provided a baseline for the *as-is* model, then evaluated the impact of different options of *Priority Contact*.

MATHSim implements a Monte Carlo simulation, running several independent trials, each of which contains several process instances that may interfere with one another's operation. The random numbers needed for Monte Carlo simulation are provided by a strong pseudo-random number generator, and the process state is maintained by a discrete event queue.

MATHsim executes a computer-based simulation [36] by generating all possible workflow and information resource combinations to estimate important performance statistics, such as the distribution for how long a workflow will take or how much time it will require from a given type of resource, such as a doctor. One of the key innovations of MATHsim is that it can make appropriate distinctions between the resource requirements for using digital information resources vs. manual information resources.

Figure 9 shows MATHsim's formative evaluation that compares the number of clinician hours required for the workflow to contact 100 patients about test results with thirty replications to approximate a normal distribution around the modes.

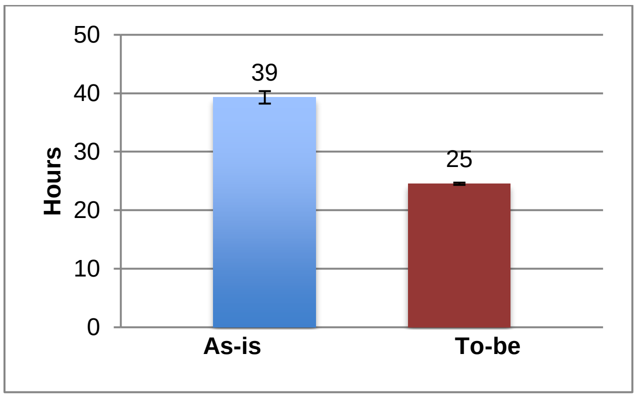


**Figure 9: Formative comparison of monthly hours spent contacting patients**

The comparison in figure 9 provided evidence of a promising advantage of about 26% time savings if the workflow used *Priority Contact*. The results were also consistent with available clinic historical data and also with test data from user interface prototypes. Based on the formative evaluation we developed the software for alpha test of *Priority Contact*. The *to-be* model and its information dictionary provided detailed specifications of how the application should work in the context of the workflow in a manner than complements the procedures of clinicians and patients, as illustrated in figures 8 and 9.

# Results of Alpha Testing

The purpose of this summative evaluation was to test the impact of the system in situ on key workflows of medical users, to collect subjective questionnaire data, and then compare results with the formative evaluation to assess how well it predicted impact during the design stage.

The summative evaluation included alpha software testing by participants their own clinical work setting. Each trial had a pre-intervention interview, software testing sessions, post-test interview, and an optional observation session. The revised models based on the summative evaluation results for contacting 100 patients are shown in figure 10.

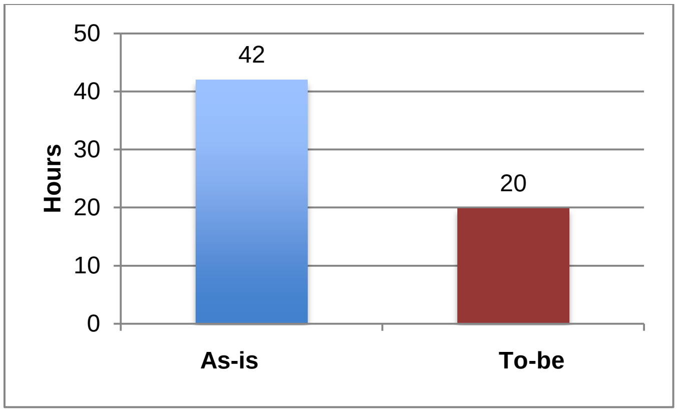


**Figure 10: Revised Models Based on Summative Evaluation Results**

The data are based on a convenience sample of ten medical workers from a variety of clinical roles. Participants were not compensated but were incentivized by the opportunity to influence the design of their future work environment. The study setting was an outpatient primary care clinic at a VA healthcare facility in the Puget Sound area. Interviews were conducted onsite in a conference room, office, or exam room. Participants tested the software on their desktop PCs at their normal places of work. Complete detail of the alpha study is published separately in Braxton, et al [38].

The *Priority Contact* software prototype was web-based and used TWeb-voice services46 to call and send text messages to mobile phones belonging to members of the research team (playing the role of patients); a set of 20 test lab results designed to represent four levels of patient contact priority (Table 3) from urgent (Priority 1) to routine follow-up (Priority 4). The fictitious test patients, (created by SME; validated by the principal investigator) contained (1) minimal clinical history (name, age, gender, presence of diabetes), whether the test was for a new or existing problem and (2) lab values that represented an increase, decrease, or new change from previous value. The percentage of cases in each priority level were weighted to represent an average week.

The test users performed patient contact tasks with the prototype at their normal workstations while test administrators recorded the same variables as in the formative evaluation. Participants tested the application on their workstation computers using simulated patient records. No patients were contacted and no identifiable personal health information were used. Because the alpha version of the prototype is not linked to the EHR, study subjects were delivered realistic, mock "alerts" regarding test results via email (1-5 alert emails per day at various times of day throughout the 7-14 day study period. Total = 20 cases). Participants viewed the attached lab report to interpret the result, and decided on their desired workflow for contacting the patient. Subjects then used their Web browser to log into their password protected, *Priority Contact* user account to (1) identify the test patient they wished to contact from a pre-populated list, (2) select the desired workflow and the message for contacting the test patient, and (3) launch a *Contact Plan* and monitor it. *Priority Contact* then initiated the workflow to reach the test patient. For the purpose of this trial, the *Priority Contact* test patients were assigned phone numbers of members of the research team. Thus the research team was able to test the functionality of the prototype without recruiting actual patients. All contact plan data were stored to a Postgres database housed on a UW server. Figure 3 shows the functional architecture of the systems used in the software trial. Additional objective data on time spent resolving test cases using Priority Contact were gathered for five participants.

Figure 10 shows the results for the summative test results on the work time for the sample of medical professional users. It produced a highly similar pattern of results to those predicted by the model-based, formative evaluation. Both were highly significant, but the formative prediction was more conservative then the benefit measured empirically in the summative test. The results of the summative evaluation showed a larger 53% savings for the quantitative impact on workflow, but also revealed policy changes that would be required to realize the benefits of the new system.

# DISCUSSION

The concordance of figures 9 and 10 demonstrates the feasibility of MATH's modeling techniques to make the impact of HIT on clinical workflow predictable and measurably beneficial. The results of methodology research need to be replicated in a range of project situations. But the patient contact problem does demonstrate a considerable scope because it involved integrating HIT functions with automated functions of communications and manual activities of clinicians and patients into a coherent, efficient workflow with error handing for patient safety.

Although we cannot make claims about causality on the basis of one study, the accuracy of the predictions was certainly based in large part on the level of detail in the workflow models. The *to-be* workflow was planned as part of the design, and task time estimates for clinicians included data from testing user interface prototypes, three-point estimates from subjects, and estimates based on similar tasks. Other factors, such as the skills of the team and the enthusiastic participation by clinicians were factors in the success of the study. For ethical reasons the alpha study used graduate students acting as patients instead of actual patients. We believe that the current feasibility demonstration should justify additional investigations leading towards widespread adoption.

MATH stops short of deciding how the beneficial impact of HIT should be used. Instead, MATH makes the benefits explicit to allow healthcare leaders to decide their value in an informed manner. For example, time-savings could be used to see more patients, to spend more time with current patients, or to alleviate overworked personnel.

The patient contact project is currently at the step for Feasibility and Cost Analysis (step #5 in figure 1). An earlier prototype won an award using the common SMART Apps API to access test patient records [38]. The alpha version of Priority Contact is made up of web services that allow for flexibility and customization. Data interoperability, however, is not only a technical issue. Policies about security and permission have to be negotiated. So, the value of the benefit will be weighed against the difficulty of changing policy or getting exception to it in the Trade-off Analysis (step #6).

MATH brings innovations in five important capabilities of workflow modeling for HIT design:
1) integrating the representation of workflow of care and information flow;
2) identifying information problems and solution options;

3) synchronizing the design of HIT functions with manual tasks to form a coherent overall workflow;
4) prioritizing options for HIT functions on the basis of evidence of benefit to workflow;
5) providing a clear connection between improvements of clinical workflow and the design of the HIT software.

These capabilities enable a new, evidence-based approach to HIT design that can be rapidly translated into software specifications. By filling the strategic gap of conventional approaches we can move towards a vision where HIT serves as a methodical means to reduce health care cost while improving its quality. Physicians, executives, and health care leaders must select and direct HIT projects, but in conventional approaches they do not have sufficient information to answer such fundamental questions as:

| | |
|---|---|
| A. How will a new EHR change current clinical care activities and decision-making? | D. How large are the reductions in cost of care? |
| B. What benefits to care will the new arrangement of activities and decisions bring? | E. Is there a range of options available for HIT functionality ? |
| C. Are there undesirable impacts on care or cost? | F. How favorable is the return on investment for each option? |

**Table 2: Some fundamental questions about EHR**

Failure to answer such common sense questions as part of software engineering for HIT creates risk of unpredictable, negative impacts on care. Until HIT can be understood in terms of added-value to care and applied reliably to realize those benefits, its potential to improve the quality of care while reducing cost will remain elusive and give providers pause to adopt it. Conversely, by supporting health care leaders with the answers will enable them to them to: Plan and compare HIT projects; provide the visibility needed to direct the execution of projects in a manner that reliably achieves planned improvement to care and its cost. Most importantly, it will allow them to participate in concept design by deciding the appropriate role of computing for their professional responsibilities.

# FUTURE DIRECTION

Patient contact is intended as just one example of how MATH can capture current care with existing information resources, identify options for measurable improvement with HIT, and provide evidence to select an HIT solution.  Current MATH projects include a multiple sclerosis clinic, a referral clinic for chronic pain treatment, and hospital admissions workflow is planned to start in September, 2014.

These projects will take advantage of more powerful features that have recently been added to MATH to take advantage of information modeling. In particular, the decision logic for gateways can now be governed by information values. This feature, in turn, allows us to model more complex interactions between information systems and workflow when the information from one task can govern the behavior of another part of the workflow. MATH also now has the capability to calculate the information architecture needed by the information flow, and export them as Java classes. In addition, a new web-based version of the MATH toolsuite has begun development to make it easier to access the tools. These capabilities make a closer connection between design of HIT and our ability to understand and analyze options for beneficial impact.

## IMPLICATIONS

The workflow of a clinic reflects a variety of factors, such as the type of patient care, organizational structure of responsibilities, personnel capabilities, billing requirements, equipment and facilities, local policy, and regulations. HIT applications, like other types of resources, constrain the way clinician can do the work of care with them. HIT must play a support role to better healthcare. MATH can design customization to maximize benefit and avoid unwanted impacts due to factors that are unique to a clinic but too important to its success to sacrifice to conform to a new, one-size-fits-all HIT application.

## SUGGESTED READING

Keith A. Butler, Chris Esposito, & Ron Hebron. (1999). Connecting the design of software to the design of work. Communications of the ACM, Jan. 1999, 42, No. 1, pp. 38 – 46.

Yunan Chen (2010). Documenting transitional information in EMR, CHI 2010, April, Atlanta, Georgia, USA.

Andre W. Kushniruk, Marc M. Triola, Elizabeth M. Borycki, Ben Stein, Joseph L. Kannry (2005). Technology induced error and usability: The relationship between usability problems and prescription errors when using a handheld application. International Journal of Medical Informatics 74, 519—526.

Jiajie Zhang & Muhammad Walji. (2011) TURF: Toward a unified framework of EHR usability. J. Biomedical Informatics, 44, 6, pp. 1056-1067.

## REFERENCES

1. Friedman, C. (2009) A 'fundamental theorem' of biomedical informatics. Journal of the American Medical Informatics Association, 16, 169-170
2. National Research Council. (2012) Health IT and Patient Safety: Building Safer Systems for Better Care. Washington, DC: The National Academies Press.
3. American Medical Association (2013). Implementation and Usability of Certified Electronic Health Records. Testimony to Health IT Policy Committee’s Workgroup on Certification/Adoption and Implementation. July 23, 2013.
4. HIMSS EHR Usability Task Force – User Pain Points Group (2010) EHR Usability Pain Points Survey Q4-2009. Available online at: http://www.himss.org/files/HIMSSorg/content/files/Usability_Pain_PointsHIMSS10.pdf
5. Zhang, J. & Walji, M. (2011) TURF: Toward a unified framework of EHR usability. J. Biomedical Informatics, 44, 6, pp. 1056-1067.
6. Chen, Y. (2010). Documenting transitional information in EMR, CHI 2010, April, Atlanta, Georgia, USA.
7. Zhang, J. & Norman, D. A. (1994). Representations in distributed cognitive tasks. Cognitive Science, 18, 87-122.
8. Zhang, J. (1997). The nature of external representations in problem solving. Cognitive Science, 21(2), 179-217.

9. Ash JS, Sittig DF, Poon EG, et al. The extent and importance of unintended consequences related to computerized provider order entry. J Am Med Inform Assoc 2007 Jul-Aug;14(4):415-23.
10. Andre W. Kushniruk, Marc M. Triola, Elizabeth M. Borycki, Ben Stein, Joseph L. Kannry (2005) Technology induced error and usability: The relationship between usability problems and prescription errors when using a handheld application. International Journal of Medical Informatics 74, 519—526.
11. Butler, Keith A. (1995): Usability Engineering. In: Kent, A. and Williams, J. "Encyclopedia of Computer Science & Technology, v.33". Marcel Dekker
12. Butler, Keith A., C. Esposito, & R. Hebron. Connecting the design of software to the design of work. Communications of the ACM, January 1999/Vol. 42, No. 1, pp. 38 – 46.
13. Butler, Keith A., Jiajie Zhang, Chris Esposito, Ali Bahrami, Ron Hebron & David E. Kieras: Work-centered design: a case study of a mixed-initiative scheduler. CHI 2007: 747-756.
14. D.D. Woods and E.M. Roth. Cognitive Systems Engineering. (1988) In M. Helander, editor, Handbook of Human-Computer Interaction, North-Holland, New York.
15. Cain, C. & Haque, S. (2008). Organizational Workflow and Its Impact on Work Quality. In: R. Hughes (Ed.) Patient Safety and Quality: An Evidence-Based Handbook for Nurses. AHRQ Publication No. 08-0043. Agency for Healthcare Research and Quality, Rockville, MD.
16. White, S. & Miers, D. (2008) BPMN Modeling and Reference Guide. Future Strategies Inc.
17. Fitts, P. M. & Jones, R. E. (1947) Analysis of Factors Contributing to 460 "Pilot-Error" Experiences in Operating Aircraft Controls. Memorandum Report TSEAA-694-12. Aero Medical Laboratory. Air Material Command. Wright-Patterson Air Force Base, Dayton, Ohio, July 1, 1947.
18. Rumbaugh, J., Jacobson, I., & Booch, G. (1998) The Unified Modeling Language Reference Manual. Addison-Wesley.
19. Bahrami, A. (1999) Object Oriented Systems Development. McGraw-Hill.
20. Rosenfield, L., & Morville, P. (2002) . Information Architecture for the World Wide Web: Designing Large-Scale Web Sites. O'Reilly Media.
21. MASSPRO, A systems approach to operational redesign. 3rd Edition. DOQ-IT, 2010.
22. Zafar, A.; Doebbeling, B.; Burton, M.; Ekbia, H.; & Lehto, M. Multidisciplainary perspectives on best practices for understanding and evaluating clinical workflows. Panel at AMIA 2010.
23. Unertl, K. Novak, L. Johnson, K., & Lorenzi, N. (2010) Traversing the many paths of workflow research: developing a conceptual framework of workflow terminology through a systematic literature review. J Am Med Inform Assoc 17, 265-273.
24. Alina M. Chircu, Janis L. Gogan, Ryan J. Baxter, Scott R. Boss, "Handoffs and Medication Errors: A Community Hospital Case Study," hicss, pp.1-10, 2011 44th Hawaii International Conference on System Sciences, 2011
25. Carayon, R., Karsh, B.T., Cartmill, R., et al. (2010) Incorporating health IT into workflow redesign: request for information summary report. AHRQ Publication No. 10-0074-EF. Rockvile, MD: Agency for Healthcare Research and Quality, July, 2010.
26. Lowry, S., Ramaiah, M., Patterson, E., Brick, D., Gurses, A.. Ozok, A., Simmons, D., & Gibbons, M. Integrating Electronic Health Records into Clinical Workflow: An Application of Human Factors Modeling Methods to Ambulatory Care. NISTIR 7798. Available at: http://dx.doi.org/10.6028/NIST.IR.7988

27. Institute of Medicine. Crossing the Quality Chasm: A New Health System for the21st Century. Washington, D.C.: National Academy Press, 2001.
28. Womack JP, Jones DT. Lean thinking: banish waste and create wealth in your corporation. New York: Free Press, 1998.
29. Y.-S. Ma, G. Chen and G. Thimm (2008) "Paradigm shift: unified and associative feature-based concurrent and collaborative engineering", Journal of Intelligent Manufacturing, special issue on Advanced Technologies for Collaborative Manufacturing, Vol.19, No.6, 2008, pp. 626-641.
30. Keith A. Butler, Jiajie Zhang: Design models for interactive problem-solving: context & ontology, representation & routines. CHI Extended Abstracts 2009: 4315-4320
31. Leekha S, Thomas KG, Chaudhry R, Thomas MR. Patient preferences for and satisfaction with methods of communicating test results in a primary care practice. Jt Comm J Qual Patient Saf. Oct 2009;35(10):497-501.
32. Baldwin DM, Quintela J, Duclos C, Staton EW, Pace WD. Patient preferences for notification of normal laboratory test results: a report from the ASIPS Collaborative. BMC Fam Pract. Mar 8 2005;6(1):11.
33. SMART-Apps Competition: http://smartapps.challengepost.com/
34. Friedman, C. & Wyatt, J. (2006) Evaluation Methods in Biomedical Informatics. Springer.
35. Eisenstein EL, Juzwishin D, Kushniruk AW, Meredith Nahm M. Defining a framework for health information technology evaluation. Stud Health Technol Inform 2011;164:94-9.
36. Wharton, C., Rieman, J., Lewis, C., and Polson, P. (1994). The cognitive walkthrough method: A practitioner's guide. In Nielsen, J., and Mack, R. (Eds.), Usability inspection methods. New York, NY: John Wiley & Sons, Inc.
37. Banks, J. and Carson, John S. II (1984) Discrete-Event System Simuation. Prentice-Hall.
38. Braxton MO, Butler KA, Haselkorn MP, et al. EB-HIT: A Workflow Approach to Achieving Evidence-based Design of Health Information Technology. American Medical Informatics Association (AMIA) Conference, 2014. Submitted.