Data Requirements for Process Learning

Johny Ghattas (*) is a PhD Student at the University of Haifa, Israel. He obtained his MSc in Telecomunications Engineering in the ETSIT, Valladolid, Spain (1995) and MBA in Heriott-Watt University, England (2002). Johny is the General Manager of Smart Path Telecom consulting company and is a management member of the ITSMF (IT Service management forum) Israel. His research is focused on advanced machine learning algorithms applications to process learning.

Johny Ghattas
Smart Path Ltd, P.O.Box 8168, 61081, Jaffa-Tel-Aviv, Israel
Tel: +972546265267
Fax: +97236816981
Email: John@Smart-Path.com

Dr. Mor Peleg has been a Senior Lecturer at the Dept. of Information Systems at the University of Haifa, Israel, since 2003, and Department Head since 2009. Her BSc (1991) and MSc (1994) in Biology and a Ph.D. (1999) in Information Systems Engineering are from the Technion–Israel Institute of Technology. She spent 6 years at Stanford Medical Informatics. In 2005 she was awarded the New Investigator Award by the American Medical Informatics Association. Her research concerns knowledge representation and decision support systems in biomedicine and appeared in Journal of the American Medical Informatics, Journal of Biomedical Informatics, IEEE Transactions on Software Eng., IEEE T on Knowledge and Data Eng., Bioinformatics.

Mor Peleg
Department of Management Information Systems, University of Haifa, Israel, 31905
Tel: +97248249641
Fax: +9724828-8522
Email: MorPeleg@is.haifa.ac.il

Dr. Pnina Soffer is a Senior Lecturer at the Dept. of Information Systems at the University of Haifa, Israel. She received her BSc in 1991, her MSc in 1994 and PhD in 2002 from the Faculty of Industrial Engineering and Management at the Technion-Israel Institute of Technology. Her research interests are business processes and conceptual modeling, including ontological foundations of both. She has served as guest editor of special issues on business processes in various journals. Her research papers appeared in journals such as Journal of the AIS, Information Systems, European Journal of IS, Requirements Engineering, and more.

Pnina Soffer
Department of Management Information Systems, University of Haifa, Israel, 31905
Tel: +97248288506
Fax: +97248288522
Email: SPnina@is.haifa.ac.il

Dr. Yaron Denekamp is currently the director of the Medical Informatics unit of the Hospital Division at Clalit Health Services corporate headquarters. He is a board certified specialist in Internal Medicine and is a leader in developing and implementing clinical information systems. Dr. Denekamp completed a post-doctoral fellowship and M.Sc. studies in Medical Informatics at the Harvard-MIT division of Health Sciences and Technology in Boston. His research area is clinical decision support systems and he is a member of the Galil center for Medical Informatics at the Faculty of Medicine of the Technion.

Yaron Denekamp
Galil Center for Medical Informatics, Faculty of Medicine, Technion Institute of Technology, Haifa, Israel
Tel: +97236946512
Fax: +97236946512
Email: Yarondp@clalit.org.il

Data Requirements for Process Learning

Johny Ghattas, University of Haifa, Israel
Mor Peleg, University of Haifa, Israel
Pnina Soffer, University of Haifa, Israel
Yaron Denekamp, Technion Institute of Technology, Haifa, Israel

ABSTRACT

Process flexibility and adaptability is essential in environments where the processes are prompt to changes and variations. Process learning is a possible approach for automatically discovering from process log data those process paths that yielded good outcomes and suggesting appropriate process model modifications to enhance future process performance in such environments. We discuss and establish the data requirements for process learning, applicable to clinical process management. Our discussion extends a previously established learning process model (LPM) by providing a formal set of data requirements which enables us to accomplish effective learning. Learning data requirements are illustrated by walking through the application of the LPM framework to a clinical process.

Keywords: Process learning, learning requirements, context, path, outcomes, soft-goals.

INTRODUCTION

Modern medicine, in its tendency towards evidence-based formalization of clinical knowledge and procedures, uses clinical guidelines in order to standardize health-care processes and use the most updated evidence based clinical knowledge. Many illnesses still have no guidelines at all, and whenever guidelines exist, for practical reasons, they do not refer to all possible patient case variations (e.g., the patient's clinical condition, past diagnoses, current medications, etc.) – just the most frequent and important ones. Since guidelines address a limited set of patient groups, it is possible that process support is not optimized for some variants that are not addressed. Our research concerns identification of important groups of patient cases and recommendation of the best process paths for them that would yield best outcomes.

In a previous work we have established the fundamentals for a process learning framework, the Learning Process Model (LPM) (Ghattas, Soffer & Peleg, 2008; Ghattas, Soffer & Peleg, 2010; Ghattas, Peleg, Soffer & Denekamp, 2010). A major component of LPM addresses context learning (Ghattas, Soffer & Peleg, 2008; Ghattas, Soffer & Peleg, 2010; Ghattas, Peleg, Soffer & Denekamp, 2010). Context refers to the set of inputs provided by the environment to the process (e.g., the patient's initial conditions, sudden changes to the patient's state). Particularly, we have developed a lifecycle approach to process learning from historical experience, based upon the premise that it is possible to group the variations in the execution of patient cases into groups – which we refer to as context groups. Each context group should be homogenous in the outcomes achieved for a clinical process execution. We demonstrated (Ghattas, Peleg, Soffer & Denekamp, 2010) how through grouping process instances into context groups, we can predict for each context group a process path resulting in good outcomes. This prediction would lead the execution of similar process instances to the best known outcomes. Repeating this learning cycle, we should obtain a better specified process model, with improved performance for each context group.

As our approach is based on learning from past experience, we need to establish the data requirements for the process learning to be effective. These data requirements need to provide a methodic way for answering the following questions: When should data be collected? How should it be formatted and coded? What should be the frequency of data collection? How do we rank each data item's importance for each context? Finally, given a case study, how can we evaluate a priori the feasibility of the required learning task?

In this paper, we provide insights to these questions by extending the LPM framework through the establishment of process learning data requirements.

The paper is structured as follows. We start by briefly describing the LPM framework. Next, we use the clinical urinary tract infection (UTI) disease management process to walk through LPM data requirements for the different components of the model. Later on, we discuss feasibility assessment of a specific learning task. Next, we review the literature and compare our model requirements to previous works. We conclude by summarizing the results of our research and presenting possible future lines of research.

THE LEARNING PROCESS MODEL (LPM)

Let us consider a clinical process; while diagnosing a patient, the clinical expert needs to consider the available data about the patient, including his current state, medical history, and any inputs that may be important for making decisions throughout the clinical process, including diagnosis and treatment.

The data required for the clinical expert to accomplish this task is provided in two different time periods: (a) initially available data from the patient records and from the initial examination of the patient; (b) data generated by external events during process execution, such as sudden changes in the state of the patient. External events, which are out of the clinical team's control , may provide additional inputs, which may require some change to the patient treatment decided up to that moment. Together, the initial inputs and the external events data determine the overall path to be adopted and constitute what we call the process context.

We assume that patients who have similar values of contextual data (i.e., belong to the same context group) should go through similar treatment paths, and in principal would be expected to reach similar outcomes. In contrast, patients with different values of contextual data may be treated similarly but the treatment would not necessarily attain the same outcomes. There might be a certain grade of variation between different executions in a context group. When different process paths are followed they might lead to different levels of performance. We may learn from historical executions the different possible paths, and adopt for each context group the path that potentially provides the best outcomes for that group.

Based on this intuitive discussion, a generic LPM would basically include three major steps, schematized in Figure 1: (1) Context learning stage, whose objective is to identify ranges of contextual data items that would predict the outcomes of an applied process path; (2) Path learning stage: Once we identify the context groups, we learn the different variations of the paths in our historical database and select for each context group the path that provides best outcomes; (3) Process model adaptation based on the learned context groups and associated best paths. Once the process model has been adapted, another cycle of learning begins, gathering more process instances and once again adapting the process model based on LPM stages.

FIGURE 1 TO BE PLACED HERE

Figure 1. Proposed LPM architecture. Legend: I=initial context data, X=external event data received during runtime, G=goal-state data, SG=soft-goal data, P=path data.

As an example for the application of LPM, consider the case of urinary tract infection disease management process. As a first step, the context learning should yield context categories such as "young women without recurring UTI" or "elder men with catheter who live in nursing homes". For the context category "young women without recurring UTI", stage 2 of LPM (path learning) would identify the paths used for this category and would recommend a path where the patient is administered a specific kind of antibiotics with a high probability of good outcome (patient is released with no further complications). For the context category "elder men with catheter", the path learning stage would learn and recommend a path where the patient is administered a more costly antibiotic in order to target antibiotic resistant bacteria and increase the probability of achieving a good outcome.

We have established a formal process model supporting the different concepts needed for LPM and we proposed an overall architecture for process learning (Ghattas, Soffer & Peleg, 2008; Ghattas, Soffer & Peleg, 2010; Ghattas, Peleg, Soffer & Denekamp, 2010). The established process model is built upon a formal state-based process framework, the Generic Process Model (GPM) (Soffer & Wand, 2004; Soffer & Wand, 2005), which was extended with the necessary concepts to support the process runtime view and context modeling. The main concepts of LPM are summarized in Table 1.

A detailed analysis and formulation of these concepts are provided in (Ghattas, Soffer & Peleg, 2008).

Table 1. Main Concepts of LPM

LPM concept	**Definition**
Context (C)	The set of all inputs provided to the process by the external environment. These can be of two types: (1) *Initial input (I)* provided to the process at t= 0; (2) *External events (X),* events providing the process with new data generated during its execution.
Process state (S)	The vector of state variable values at a given moment of the process execution.
Termination state (t)	The end state of the process instance. It can be either a *goal state* (G) or *exception (E).*
Goal (G)	The set of end states of the process, in which the process is assumed to already comply with its objective. The process goal is represented as a set of states as it may have different variations, all of which are assumed to comply with the process objectives. Different goal states are differentiated by how well they accomplish the process objectives. This may depend on several soft-goals (SG's) such as performance and quality measures.
Exception (E)	A process instance may terminate in an unexpected state that is not one of the goal states. We call such state an *exception* (E).
Process path (P)	The sequence of states which the process goes through during its execution. A process state change may be caused either by an internal event (i.e., an activity) or by an external event, triggered by the environment and not controlled by the process.
*Process instanc*e (PI)	A PI is modeled as a tuple <P, C, t>, that is the instance path, context, and termination state. Hence, process instances are characterized basically by their behavior (defined by their path and termination state), and their context, which is imposed by the process environment and constrains the process behaviors that can be adopted for any given process instance.
Context Group (CG)	A group of instances that share similar paths, outcomes, and contexts.

Illustration of Process Learning Data Requirements through UTI Case Study

In this section, we illustrate LPM data requirements by walking through the LPM application to the urinary tract infection (UTI) clinical process. We start by providing a brief overview of the UTI clinical process. Although we illustrate the data requirements using a specific case study, the presented requirements were derived from our experience with several case studies from different domains, such as the medical domain, specifically, ear infection diagnosis and treatment (Peleg, Soffer & Ghattas, 2007) and UTI infection process (Ghattas, Peleg, Soffer & Denekamp, 2010), manufacturing processes (Soffer, Ghattas & Peleg, 2010), service provisioning processes (Ghattas, Soffer & Peleg, 2010), "daily morning preparation" process (Ghattas, Soffer & Peleg, 2008), Customer care processes, among others. Within this paper, we provide data requirements that were validated through all the case studies examined during our research.

UTI Overview

We consider the implementation of the UTI management process based on available clinical guidelines and medical literature (NGC, 2010; Wein, 2007). We focus on patients that reach the hospital's emergency room (Figure 2-activity: “Receive Patient at emergency Room”). The clinical expert interviews the patient and reviews his electronic medical record (EMR) (Figure 2-“EMR review”), his medical history, medication listings (Figure 2-“Current drugs listing & review”), and performs a physical examination (Figure 2-“Physical and vital signs check”). In addition, the clinical expert may require the patient to undergo several tests (mainly blood and urine tests, CT, etc.) (Figure 2-“Urine tests”, “Blood tests”, “Other procedures & tests”).

Following the initial diagnosis (Figure 2-“First Diagnosis (UTI Y/N/Inconclusive)”), the patient may be provided with an initial antibiotic, and other medications (e.g. fever/pain reduction medications, fluids, etc.) (Figure 2-“ER treatments”). The clinical team decides whether the patient needs to be hospitalized or sent to home care (Figure 2-“Decide Hospitalization”), based on the overall clinical evaluation. During the entire process, if unexpected complications occur (e.g., hematuria, kidney failure, heart failure, etc.), the patient may be submitted to emergency treatment and procedures (Figure 2-“Emergency procedures and treatments”). Once hospitalization is decided, the patient may undergo additional tests to further diagnose his condition; tests may include ultrasound, CT, etc. (Figure 2-activity: “Further lab and other tests”). Some test results, mainly urine and blood cultures, may arrive several days later; based on these test results, the treatment, such as antibiotic type may be changed (Figure 2-“Further/Change treatments & drugs”) and additional tests may be ordered (Figure 2-“Further lab and other tests”). In addition, in some cases such as urinary obstruction, patients may go major procedures (Figure 2-“Procedures (minor/major procedures)”).

FIGURE 2 TO BE PLACED HERE

Figure 2. Overall UTI process modeled using BPMN notation.

LPM Data Requirements

In order to illustrate the LPM requirements, we will go through the data specification for each building block of the model (context, path, and outcome data), highlighting issues we faced and requirements we needed to impose on the data for the learning task to be effective. In our requirements presentation we will use a well known framework for data requirements, provided by Wang, Strong & Guarascio (1996). Wang, Strong & Guarascio (1996) established four categories for data quality requirements: (1) intrinsic data requirements, which are requirements related to creating correct and true data values; (2) contextual data requirements, whose objective is to ensure the data collected is pertinent to the task to be accomplished; (3) representational data requirements which relate to supplying intelligible and clear data; (4) accessibility data requirements which relate to providing readily available and obtainable data. Each category is further detailed in dimensions (Table 2).

For each LPM data requirement that we present, we will highlight the relevant Wang category(ies) and associated dimensions.

Table 2. Data quality categories and dimensions following Wang, Strong & Guarascio (1996).

ID	Category type	Category meaning	Dimensions
W1	Intrinsic	Creating correct and true data values	Accuracy, Objectivity, Believability, Reputation
W2	Contextual	Data pertinent to the tasks of the user	Value addition, Relevance, Timeliness, Completeness, Appropriate Amount
W3	Representational	Supplying intelligible and clear data	Interpretability, Ease of Understanding, Representational Consistency, Concise Representation
W4	Accessibility	Providing readily available and obtainable data	Accessibility, Security

Identification of the initial context data (I).

Some of the initial context data is known from the medical record of the patient (either electronic (EMR) or paper-based); further data is collected during the patient interview by the medical expert (anamnesis), during which the physician questions the patient to identify chronic illnesses, active prescriptions, symptoms, UTI recurrence, UTI related historical illnesses (e.g., calculi, reflux problems, kidney problems, etc.), general test results (urinalysis); physical examination and general tests provide additional context data. We relied on the clinical expert and on clinical guidelines' review to identify relevant context data. A partial list of context data is provided in Table 3.

A basic validation of data completeness was done by verifying that we had all data necessary for determining the branching points (XOR joins in Figure 2) in the clinical process execution.

Requirement 1: Data completeness (W1/Accuracy, W2/Completeness,W3/Interpretability): Once the process model is specified, the availability, accuracy and interpretability of all necessary data items need to be assessed. This can be accomplish by (1) checking the data required at each branching point of the process; (2) checking inputs required by each activity; (3) using domain knowledge and looking for domain expert advice.

Table 3. Context data (partial list).

Context Item	Context Item
Demographics	Chronic illnesses
Age	Diabetes mellitus
Gender	Hypertension
Vital Signs	Coronary arterial disease
Temperature	Congestive heart failure
Blood pressure	Cancer
Heart Rate	Chronic pulmonary disease
Symptoms	Chronic renal failure
Physical examination	Cerebro-vascular disease
UTI history	Medications
Permanent catheter	Hospital acquired UTI
Mental state	Residence Type
Functional state	Past tests results

The context data (Table 3) presented several issues we needed to address, before being able to use it for our learning purposes.

First, some of the process instances were missing essential context data such as active prescriptions or essential path data (initial and/or final treatment types mainly related to antibiotics type). When the instances could not be completed with the help of the clinical expert, they had to be ignored.

Requirement 2: Context data missing data handling (W1/Accuracy,W2/Completeness,W3/Interpretability): It is necessary to decide whether an instance with partial data is to be ignored or completed using estimation, depending on how accurately the missing data can be filled in. Several methods of automatic data completion have been proposed in (Bowerman, O'Connell & Hand, 2001): fill with a constant value (e.g., "Unknown") or statistic value derived from similar instances.

Chronic illnesses data (Table 3) are accompanied by a history of patient's events, treatments, and procedures; however without the severity assessment of the illness, this data is meaningless for clinical decision-making that determines process paths selected and outcome reached; hence we had to code each chronic illness based on its severity, with the help of the clinical expert.

Requirement 3: Data item meaning (W1/Accuracy and believability,W2/Completeness): Data items coding needs to take into account complementary data taken from domain knowledge and/or other data items to ensure the data is meaningful and represents faithfully the necessary facts it is meant to convey.

In some cases, we added new data variables, derived from other variables. Such was the case of the pulse pressure, derived as the difference between the systolic and diastolic heartbeat rates, an important variable which may indicate shock, low stroke volume, or low cardiac output (Simon & Boring, 1990). In addition, medications (Table 3) are sometimes associated with the patient's chronic illnesses, e.g., insulin is mostly associated with diabetes mellitus, etc. In some cases, we could infer missing chronic illnesses from the medications given to the patient. As a result, we added a new iron deficiency anemia field.

Requirement 4: Data derivation (W2/Value Addition): Data items need to be derived from existing data. Derivation may use simple mathematical expressions, a procedure involving multiple variables as independent variables, or rely on more complex methods involving feature extraction from exiting data (Dasu & Johnson, 2003).

Some medications (Table 3) are associated with the patient chronic illnesses, while others (e.g., antibiotics) are given to treat UTI – the acute condition that is the main focus of the process studied. As long as a medication is not treating the main medical problem addressed (UTI), we can select whether to keep the medication or the chronic illness specification, as both data items are not needed. In most cases we decided to keep the chronic illness as we need to know in addition to the illness presence, its severity and history, as they are important for path selection and outcomes reached.

Requirement 5: Data redundancy reduction (W2/Value Addition and relevance): Data item dependencies need to be assessed; data items that are already represented by other items should be filtered out.

The patient's medical record is rich in data, part of which is irrelevant for UTI. Such was the case of historical data and chronic illnesses we filtered out, leaving only UTI relevant complications.

Requirement 6: Data item relevance assessment (W2/Relevance): Context data relevant for each context group need to be identified out of the available data for each specific learning task. This may be done through domain knowledge and/or using feature selection methods (Akaike, 1974), as we did in (Ghattas, Peleg, Soffer & Denekamp, 2010).

One major problem in UTI cases is the appearance of "extended-spectrum beta lactamase" (ESBL) bacteria types, which exhibit resistance to different kinds of antibiotics (Wein, 2007). Knowing whether ESBL is present influences the process path (i.e., the selection of appropriate antibiotic treatment). The issue we faced wass that ESBL existence or inexistence cannot be determined when the patient is admitted to the hospital; Due to its importance, we used a decision tree to learn and predict the existence of ESBL based on existing initial context data (e.g., partial results of blood and urine test, patient state, medical history, catheter presence, an indication whether it was a hospital acquired infection).

Requirement 7: Hidden context variables prediction (W2/Value addition and relevance): Predicting variables through other context data by inference should be considered when such variables are known a-priori to affect the process path and outcomes and/or render the learning task easier to accomplish.

The variables "Past test results" (set of different test results including blood, urine, blood pressure, heartbeat rate, cholesterol, etc.) and "UTI History" (set of treatments, events and procedures the patient has gone through in the past) contained sets of unrelated and different types of data. In order to simplify the learning task, the UTI Past test results variable was replaced by a set of single-type variables, such as "iron deficiency anemia" and "Cholesterol", which reports trends of aggravation of anemia and Cholesterol levels,.

Requirement 8: Reducing sets of multiple data types into sets of single-type data (W2/Value Addition, W3/Interpretability and Concise representation): Complex data items containing multiple types of data should be reduced into sets of single data type variables.

Considering the continuous variable "age", the exact age of the person is not important; hence we discretized it in ranges of ten years, assigning each instance the midst value of the range (e.g. value 60 represented the age range [55-65]).

Requirement 9: Continuous data discretization (W2/Relevance, W3/Ease of understandability and representational Consistency): Continuous Data items should be discretized to simplify processing. This can be done by dividing the range of values of the variable into sub-intervals that can be either of equal length, equal depth, or equal frequency ranges (Dasu & Johnson, 2003).

Patients with chronic illnesses such as diabetes are required to measure their blood glucose level several times a day. These sequences of time-stamped measures are mainly used for monitoring the evolution of the relevant vital signs of the patient, seeking for symptoms and/or trends. Sequences are harder to process than simple variables; hence we replaced them by their clinical interpretation (trend and/or relevant severity).

Requirement 10: Sequence data processing (W2/Value addition, W3/Concise representation): Sequence data needs to be substituted by its domain interpretation (e.g., trends, average value, median value, etc).

For most of the UTI context data, rather than the numerical values of measures, such as body temperature, heartbeat rate, blood and urine test results, we needed their clinical interpretation as abstractions (e.g., fever/hypothermia, hypotension/hypertension, anemia, etc.). We recoded these values using clinically meaningful categories in order to highlight the clinical situation.

Requirement 11: Data coding (W3/Ease of understanding): Data needs to be coded to reflect required domain knowledge. This may reduce considerably the complexity of the learning task.

Identification of external events data (X).

UTI external events are occurrences of unexpected complications (e.g., high fever, abdominal pain, hematuria, renal failure), which modifies the clinical path. For example, whether the fever continues to increase before or long after the antibiotic treatment is applied, may mean different things; if it occurs after the patient is administered with antibiotics this may require a change of the antibiotic type.

Requirement 12: External events association with process state (W3/Interpretability): Each external event needs to be associated with the process state in which it occurs. One possible solution is time-stamping external events and states.

Identification of path data.

An example of path data in the UTI management process path is provided in Table 4.

In LPM, the process path is modeled as the evolution of the process state from one set of states to another. An example of the UTI sets of states is provided in Figure 3.

Each specific state, which is represented by a vector of state variable values, should be assigned to a state set. Note that although timestamps may be used for sequencing, it is not always mandatory, as in some cases it would be sufficient to establish the order of the states.

Much of the UTI process is spent on patient follow up, monitoring his current state (e.g., vital signs, fever, blood pressure, etc.) and making therapy changes (treatments, medications, minor or major procedures) whenever necessary. Follow-ups collect a considerable amount of data, whose relevance we need to assess. Based on UTI guidelines review, we coded only those follow-up results that affected the clinical path, e.g., resulted in antibiotics modifications or major procedures.

FIGURE 3 TO BE PLACED HERE

Figure 3. An example of a UTI process state evolution. Legend: DGx= Diagnosis # x, TRx = Treatment #x; HTN= Hypertension, R= Tests Results, BP= Blood pressure, HR= Heartbeat-rate.

Table 4. Path data structure.

Data Item Name	Data Item Value
Process instance ID	253467
Partial tests outcomes	<blood tests partial results>, <Urine tests partial results>
ER Initial Diagnosis	<UTI>, <Estimate ESBL+ = N>
ER Initial Treatment	< Zinacef>, <Fluids>
Hospitalization decision	Y
Follow ups	{Temperatures, Blood pressure, sugar levels, treatments, procedures.}
Urine Culture test results	<…>(1 field/measure), <ESBL+= Y>
Blood test results	<…> (1 field/measure)
Additional tests:	<ULTRASOUND Results>, <CT results>
Modified treatment	< INVANZ>, <Fluids>, <Inotropes>
Procedures	<Catheterization>, <surgery …>
Final Patient status	< Released>

Requirement 13: Relevant path data identification (W2/Relevance). For each activity of the process, relevant data needs to be identified, based on (1) domain knowledge and (2) whether this data serves to determine that the process has moved to a different set of states.

Some of the process' sets of states are hard to identify, e.g., how do we identify the process set of states “Patient state improved” and “Patient state worsened”? In Figure 3, process state “Patient state improved” is based on the criteria “BP decreased to normal value (135/95)” AND “Temperature is normal”; in other instances the same state may be based on different criteria depending on the specific initial context.

Requirement 14: State identification criterion (W3/Interpretability and ease of understanding): Each possible set of states needs to be identifiable. Identifying the set of states per context group may reduce the state variability and thus reduce the complexity of this task. Accomplishing state identification may be automated by defining sets of predicates over state variables values. As an example, in Figure 3, we may establish that state “Patient diagnosed” occurs once variables “Treatment X” (See “TRx” in Figure 3) values are set, and state “Antibiotic change” occurs once state variable “Antibiotic Type” is changed. Finally, note that mapping the process model expressed in Figure 2 in BPMN to a state based model as expressed in Figure 3 requires detailing the process model further (e.g. drilling down to tasks inputs and outputs variables) and is out of the scope of this paper.

Vital signs values such as temperature and blood pressure are important inputs for supporting the clinical expert while assessing and diagnosing the patient's illness. In LPM we would like to recommend to the clinical expert the steps to be taken based on the values of such decision parameters. For example, we should have at least 3 ranges for temperature measure – hypothermia (T<36.5^{o}C), normal (36.5^{o}C<T<38^{o}C), fever (T> 38^{o}C).

Requirement 15: Path data granularity (W3/Interpretability, Representational consistency and Concise Representation). Path data coding should take into account the needed level of detail of each data item involved with path decision making.

Identification of termination state data (t).

As a next step, we need to distinguish sets of UTI termination states, both goal states (G) or exceptions (E). We identified five termination states, including three goal states (G_1-G_3), one exception state (E_1) and a generic exception state (E_*) that would capture exceptions that are termination states that we did not identify as goal states. To do so, we first established a criterion for identifying each goal state {G_i}.

Requirement 16: Goal state identification criterion (W3/Interpretability, ease of understanding): A clear goal state identification criterion must be established as a predicate over the relevant state variables.

We need to identify the occurrence of undesired termination states (e.g., death of the patient, defined as E_1 in Table 5). Each exception state needs also to be labeled in the same way as we do for goal states, using a logical expression over relevant state variable, as shown in Table 5.

Requirement 17: Exception identification criterion (W3/Interpretability, ease of understanding): Desired and undesired termination states should be distinguished. Exceptions should be identified as any non goal state from which the process does not continue. This would be expressed as any state that has not been identified as one of the known termination states (either goal or exceptions known states).

Table 5. Termination state identification criteria

Termination state name	Termination state identification criteria
G_1- Patient Cured	<PatientCured==‘Y’>
G_2- Patient sent to Home Care	<PatientSentToHomeCare =‘Y’>
G_3- Require other specialists examination	<RequireOtherSpecialists= ‘Y’>
E_1- Patient death in Hospital	<DeathInHospital=‘Y’>
E_*- Exception	<Termination states != {G_i},*i=1..3*>

Some of the UTI instances were of terminal cancer patients that were transferred to a different clinic. While this state is a legal one, we did not account for this as a termination state in our initial process model. This required a modification of our termination state criteria to include a fourth goal state.

Requirement 18: Exception terminated instances analysis (W1/Accuracy,W2/Completeness): Process instances that get stuck in non-goal states need to be analyzed to assess whether they are unaccounted-for goal states.

Evaluation of soft-goals (SG).

Each goal state of the process may have associated soft-goals. We considered three major UTI soft-goals: (1) the length of stay in the hospital (LOS); (2) patient state severity upon release (PRS); and (3) total cost of stay (TCS).

Requirement 19: Support multiple soft-goals (W2/Value Addition and Relevance): Each goal state needs to be associated with at least one soft-goal.

Soft-goals may exhibit negative, positive or unclear correlation with other soft-goals. For example, we note that LOS and PRS are negatively correlated; hence they cannot be simultaneously improved.

Requirement 20: Assessment of soft-goals cross-correlations (W3/Interpretability, ease of understanding): Soft-goals cross-correlations need to be assessed in order to check the feasibility of simultaneously improving several soft-goals. Negatively correlated soft-goals cannot be optimized simultaneously.

Learning task feasibility assessment.

In addition to the challenges associated with coding the data for different LPM components, we need to assess the feasibility of the learning task we have at hand, considering the data which is available.

For the UTI case study, we have been dealing with two different learning tasks: (1) Learning the context groups of UTI elderly patients reaching the emergency room; (2) Learning positive ESBL context groups for UTI elderly patients.

In the first learning task, described in Ghattas, Peleg, Soffer & Denekamp (2010), we obtained satisfactory results. By clustering the paths and outcomes into five path instance categories (PIC's) and building context groups using a decision tree algorithm, we could define predicates over patient data items and their values for each context group. These predicates predict with 92.2% specificity and 45.5% sensitivity the outcome achieved when a certain process path is followed.

However, for the second task, we faced some issues that prevented us from reaching conclusive results. ESBL cases are currently identified through blood and urine cultures, whose results are obtained after 2-3 days. The objective was to try to establish a way of identifying ESBL contexts with relatively high accuracy in much shorter time using LPM. The set of data instances we collected had 22% ESBL cases that where cured and 4% of ESBL cases which resulted in death,

LPM builds on establishing the path similarity and outcome similarity and grouping cases that simultaneously have path and outcome similarities in what we call process instance categories (PIC).

We distinguished 16 different PICs based on path data (different diagnosis, treatments type (antibiotics types, which were classified as either covering ESBL for specific bacteria types or not covering any ESBL case), procedures and outcome data). This partitioning of the data reduced the number of instances in some of the PICs obtained so unacceptable sensitivity and/or specificity values of the learning algorithm were obtained (Simon & Boring, 1990). For example PIC's derived from the group of ESBL patients that died in the hospital (4%) had insufficient instances quantity for learning.

Requirement 21: Appropriate number of instances (W2/Appropriate amount). Getting enough instances for each targeted context category is required in order to ensure our data sample is valid for learning. The number of required instances depends upon the total number of instances as well as on the distribution of instances over the context groups. In addition, we need to ensure that the set of instances really represents the different business scenarios and the relevant process paths. However, it may be hard to get large enough sets of instances of infrequent context categories (Bowerman, O'Connell & Hand, 2001).

Clinical knowledge, processes, technology, procedures and treatments evolve constantly. Hence we need to choose data sample that reflect timely process instances, corresponding to current medical practice. As the learning process is continuous, we can ignore past process instances that reflect outdated practices by periodically identifying approved changes that were made in the processes and consulting with the business process experts to decide whether instances collected before these changes are still relevant.

Requirement 22: Depth of relevant historical database (W2/Relevance, W3/timeliness): Learning requires considering the relevance of the instances based on their timeliness, as the process evolves based on technology, knowledge, standards, etc.

We provide a requirement summary in Table 6, where in addition we prioritized each one of them in order to differentiate mandatory from optional ones (see column "Requirement priority") . By "priority", we refer to how much that requirement is essential for our learning approach. Critical requirements are those requirements which are key for the success of our learning task. Such are requirements 1, 2, 6, 12, 14 and 16-20 which relate to correctly identifying and representing the components of LPM (context, path, goal, states, external events, exceptions and soft-goals); We divided non-critical requirements into two sub-categories: (1) high priority requirements, and low priority requirements. While both categories do not affect the viability of the learning task, high priority requirements are those related to deriving data from the collected data, such as hidden context data (requirement 7), feature selection and redundant data items elimination (requirements 5, 13, 15) or adding domain knowledge to the data through coding or representation modification (requirement 11), all of which require business process domain specific knowledge. Low priority requirements (requirements 4, 8, 9 and 10) are more related to technical and statistical processing issues and depend exclusively on statistical knowledge.

RELATED WORK

Data Specification Requirements

Data specification requirements have been largely discussed in the literature (Martin, 1976; Wang, Strong & Guarascio, 1996; Zmud, 1978), and in particular, Wang, Strong & Guarascio (1996) data quality framework which has been used throughout this paper. Our requirements cover three out of four

Wang categories, as the accessibility category is less relevant for our objective.

Table 6. Data requirements' summary. Legend: DS/CL/PL=Data specification/Context learning/Path learning. W1/W2/W3 are the Wang Data quality dimensions as provided in Table 2;M/H/L= Mandatory/High priority/ Low priority.

#	Requirement	Wang Data Quality dimensions/Categories (see Table 2)	Req. Priority	Relevant LPM stage		
				DS	CL	PL
1	Data completeness	W1/Accuracy,W2/Completeness,W3/Interpretability	M	√		√
2	Context data missing data handling	W1/Accuracy,W2/Completeness,W3/Interpretability	M	√	√	√
3	Data item meaning	W1/Accuracy and believability,W2/Completeness	M		√	√
4	Data derivation	W2/Value addition	L		√	√
5	Data redundancy reduction	W2/Value addition and relevance	H		√	√
6	Data item relevance assessment	W2/Relevance	M		√	√
7	Hidden context variables prediction	W2/Value addition and relevance	H		√	
8	Reducing sets of multiple data types into sets of single-type data	W2/Value Addition,W3/Interpretability and concise representation	L		√	√
9	Continuous data discretization	W2/Relevance,W3/Ease of understandability and representational consistency	L		√	√
10	Sequence data processing	W2/Value addition,W3/Concise representation	L		√	√
11	Data coding	W3/Ease of understanding	H			√
12	External events association with process state	W3/Interpretability	M	√		
13	Relevant path data identification	W2/Relevance	H	√		√
14	State identification criterion	W3/Interpretability and ease of understanding	M	√		√
15	Path data granularity	W3/Interpretability, representational consistency and conciseness	H	√		√
16	Goal state identification criterion	W3/Interpretability, ease of understanding	M	√		
17	Exception identification criterion	W3/Interpretability, ease of understanding	M	√	√	
18	Exception terminated instances analysis	W1/Accuracy,W2/Completeness	M	√	√	
19	Support multiple soft-goals	W2/Value Addition and relevance	M	√	√	
20	Assessment of soft-goals cross-correlations	W3/Interpretability, ease of understanding	M	√	√	
21	Appropriate number of instances	W2/Appropriate amount	M	√	√	√
22	Depth of relevant historical database	W2/Relevance,W3/timeliness	H	√	√	√

Approaches in Medical Informatics with Some Similarities to LPM Context Modeling

In the medical informatics literature, Tu et al. (2007) considered usage scenarios in order to identify opportunities for providing decision support. Usage scenarios have some similarities to our context concept as they model all the necessary process inputs: "who is doing what, where and when". Though in our UTI case study we focused more on what and when rather than on the actors (who) of the process,

The definition of Act classes in Health Level 7's Reference Information Model (RIM) (Russler, Schadow, Mead, Snyder, Quade & McDonald, 1999) takes an action-centered view, where Act classes identify the kind of action, the actors, the objects or targets which the action influences, all of which may be considered as part of LPM path, although LPM does not provide for a specific concept for actors. RIM adverbs of location, time , manner, and other information about circumstances, such as reasons or motives, would be considered part of LPM context definition, though in LPM we do not model explicitly these adverbs; rather our approach advocates assessing the relevance of each data type (location, time, etc.), based on its impact on the process outcomes.

Data Modeling in the Process Mining Research

Van Dongen & van der Aalst (2005) proposed a meta model for process mining event logs (MXML) which describes process instance logs as a sequence of audit trails and data elements; each audit trail describes an activity and contains a workflow model element, an event type, a timestamp and originator elements. Though some similarities exist with LPM path data requirements, MXML does not provide for context and outcome data support, while LPM does not provide a specific element for describing actors.

The IEEE taskforce on process mining proposed XES (Extensible Event Stream) for formatting generic event log data (Günther, 2010). Compared to LPM, XES focuses on providing a generic framework onto which all event log meta-models found in practice can be mapped with relative ease, with almost no requirements imposed on the data, while LPM goes further in establishing the semantics of the process runtime model. Finally, compared with both previously mentioned frameworks (MXML and

XES), LPM requires a higher data granularity for the context and path learning stages to be feasible.

Learning from process adaptations might also be fostered by approaches like process mining (Jagadeesh, Bose & van der Aalst, 2009; van der Aalst, Weijters & Maruster, 2004), whose objective is to analyze process event logs and path variants. . However, most of these approaches focus on mining the sequences of activities the process goes through, while the data flow perspective is not addressed.

Learning from past experience has been addressed in ProCycle (Weber, Reichert, Rinderle & Wild, 2009), which allows reusimg process model adaptations in similar problem context by capturing contextual information and mapping relations of process instance changes to cases that can be retrieved later when faced with a similar problem context.

Finally, other context-aware frameworks have been suggested to facilitate the implementation of application services, which can somehow adapt their behavior to changing circumstances. Most of these frameworks (Mikalsen & Kofod-Petersen, 2004; Bikakis & Antoniou, 2010; Fahy & Clarke, 2004) provide support for gathering and processing context data from the real world. However, they leave the reaction to context changes to the application or use hard-to-maintain rule-based approaches for dealing with changes.

CONCLUSIONS AND FUTURE WORK

We established the data requirements necessary for effective process learning. Based on our previously formal learning process model (Ghattas, Soffer & Peleg, 2008; Ghattas, Soffer & Peleg , 2010; Ghattas, Peleg, Soffer & Denekamp, 2010), which includes a three stage learning algorithm, mainly context learning, path learning and process model adaptation, we established the set of requirements for each one of the three data components: the context, path and outcome data. To ensure that these requirements are generic and applicable to a wide variety of domains, we analyzed a variety of case studies, including processes from the clinical domain, the manufacturing domain, service provisioning, etc.

Without reducing the generality of these requirements and their applicability to other case studies mentioned in the Introduction, we illustrated the requirements by walking through a clinical process case study. Finally, we discussed the priority of the requirements based on their impact on the feasibility of learning, as described in Table 6.

We intend to use these requirements as a learning feasibility assessment framework, which would enable us (1) to assess whether a specific learning task is feasible or not and (2) to establish a formal approach for assessing and coding process data for new process learning tasks. Further work needs to be done in order to consider the necessity of introducing more extensive time handling techniques in learning tasks. Finally, in order to further explore the applicability of the presented requirements a process simulator has been designed to allow us to further experiment with different levels of complexity and variability of process models, the context relation to the paths and outcomes, and the capacity of our LPM to learn and improve the process logic. We also intend to explore cases where interaction and synchronization exists between different process instances running in parallel and sharing resources, where we need to consider the dependence between the contexts, paths and outcomes of different instances.

REFERENCES

Akaike H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*; 19(6):716-723, 1974.

Van der Aalst W. M. P, Weijters T. & Maruster L. (2004). Workflow mining: Discovering process models from event logs, *IEEE TKDE,* 16(9), 1128–1142, 2004.

Bikakis A. & Antoniou G. (2010). Defeasible Contextual Reasoning with Arguments in Ambient Intelligence. *IEEE TKDE*, 22(11), November,2010.

Bowerman B.L., O'Connell R.T. & Hand M.L. (2001). *Business Statistics in Practice (2nded.),* McGraw Hill, 2001.

Dasu T. & Johnson T. (2003). Exploratory Data Mining and Data Cleaning. *John Wiley & Sons*, 2003.

Fahy P. & Clarke S. (2004). CASS – a Middleware for Mobile Context-aware Applications. *Proceedings of the Workshop on Context Awareness (held in connection with MobiSys'04)*, 2004.

Ghattas J., Peleg M., Soffer P. & Denekamp Y. (2010). Learning the Context of a Clinical Process. *Third International Workshop on Process-Oriented Information Systems in Healthcare (ProHealth 2009),* Lecture Notes in Business Information Processing 43(1):545-556, Springer, 2010.

Ghattas J., Soffer P. & Peleg M. (2010). A Formal model for Process Context Learning. *The Fifth International Workshop on Business Process Intelligence (BPI),* in conjunction with BPM 2009, Lecture Notes in Business Information Processing, Vol. 43, 140-157, Springer, 2010.

Ghattas J., Soffer P. & Peleg M. (2008). A Goal-based Approach for Business Process Learning. *Workshop on Business Process Modeling, Development, and Support (BPMDS'08),* in conjunction with CAISE'08;Montpellier,France, 2008.

Günther W.C. (2010). Open XES Developer Guide. *Webpage: http://www.xes-standard.org/_media/openxes/developerguide7.pdf.* Accessed on March 2d, 2011.

Jagadeesh R.P., Bose C. & van der Aalst W. M. P. (2009). Context-aware Trace Clustering: Towards Improving Process Mining Results, in *Proceedings SDM'09*, 2009, pp. 401–412.

Mikalsen M. & Kofod-Petersen A. (2004). Representing and Reasoning about Context in a Mobile Environment. Schulz S, Roth-Berghofer T, editors. *Proceedings of the First International Workshop on Modeling and Retrieval of Context*. CEUR Workshop Proceedings; Ulm, Germany; 2004. p. 25-35.

NGC, (2010). UTI Guideline. National Guidelines Clearinghouse, Agency for healthcare research and quality. *Webpage:http://www.guideline.gov/content.aspx?id=7407.* Accessed on March,2d,2011.

Peleg M., Soffer P. & Ghattas J. (2007). Mining Process Execution and Outcomes. *Business Process Modeling Conference Workshop: 1st International Workshop on Process-oriented Information Systems in Healthcare*, Brisbane, Australia. September 2007.

Sobek J. (2010). What Does a Small Margin Between Systolic & Diastolic Blood Pressure Mean?W*ebpage: http://www.ehow.com/facts_5998451_small-diastolic-blood-pressure-mean_.html.* Accessed on the March,2d,2011.

Russler D.C., Schadow G., Mead C., Snyder T., Quade L.M. & McDonald C.J. (1999). Influences of the Unified Service Action Model on the HL7 RIM. *Proceedings AMIA Annual Symposium*; 1999. p. 930-4.

Soffer P., Ghattas J. & Peleg M. (2010). A Goal-based Approach for Learning in Business Processes. In: *Intentional Perspectives on Information Systems Engineering*, Camille Salinesi, Carine Souveyet, Jolita Ralyte editors. Springer. February 2010.

Soffer P. & Wand Y. (2004). Goal-driven Analysis of Process Model Validity. *Advanced Information Systems Engineering (CAiSE'04)* (LNCS 3084); 2004. p. 521-535.

Soffer P. & Wand Y. (2005). On the Notion of Soft Goals in Business Process Modeling. *Business Process Management Journal* 2005;***11***(6):663-679.

Simon D. & Boring, J.R. III (1990). Sensitivity, Specificity, and Predictive Value. In: *Clinical Methods: The History, Physical, and Laboratory Examinations*, Walker HK, Hall WD, Hurst JW, eds. Butterworths, 3rd edition,1990.

Tu S.W., Campbell J.R., Glasgow J., Nyman M.A., McClure R.J., McClay J. P.C., Hrabak K.M., Berg D., Weida T., Mansfield J.G., Musen M.A. & Abarbanel R.M. (2007). The SAGE Guideline Model: achievements and overview. *Journal of the American Medical Informatics Association* 2007; 14(5):589-98.

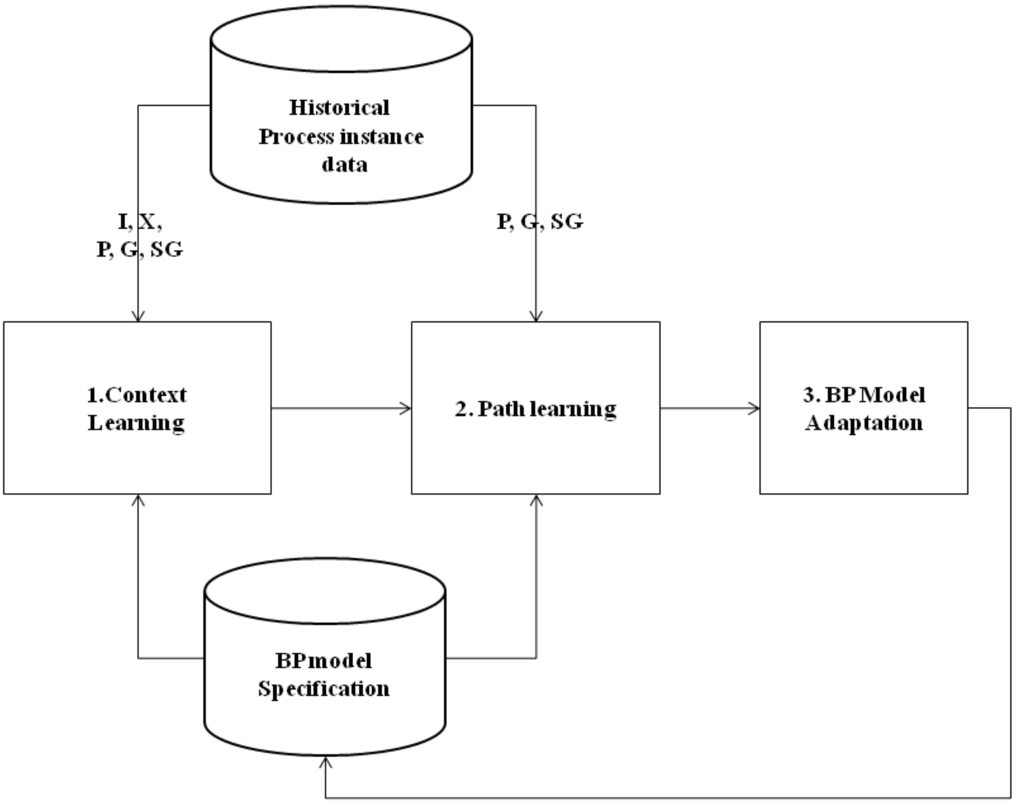
Van Dongen B.F. & van der Aalst, W.M.P (2005). A Meta Model for Process Mining Data. In J. Casto and E. Teniente, editors, *Proceedings of the CAiSE'05 workshops (EMOI-INTEROP Workshop)*, volume 2, pages 309-320. FEUP, Porto, Portugal, 2005.

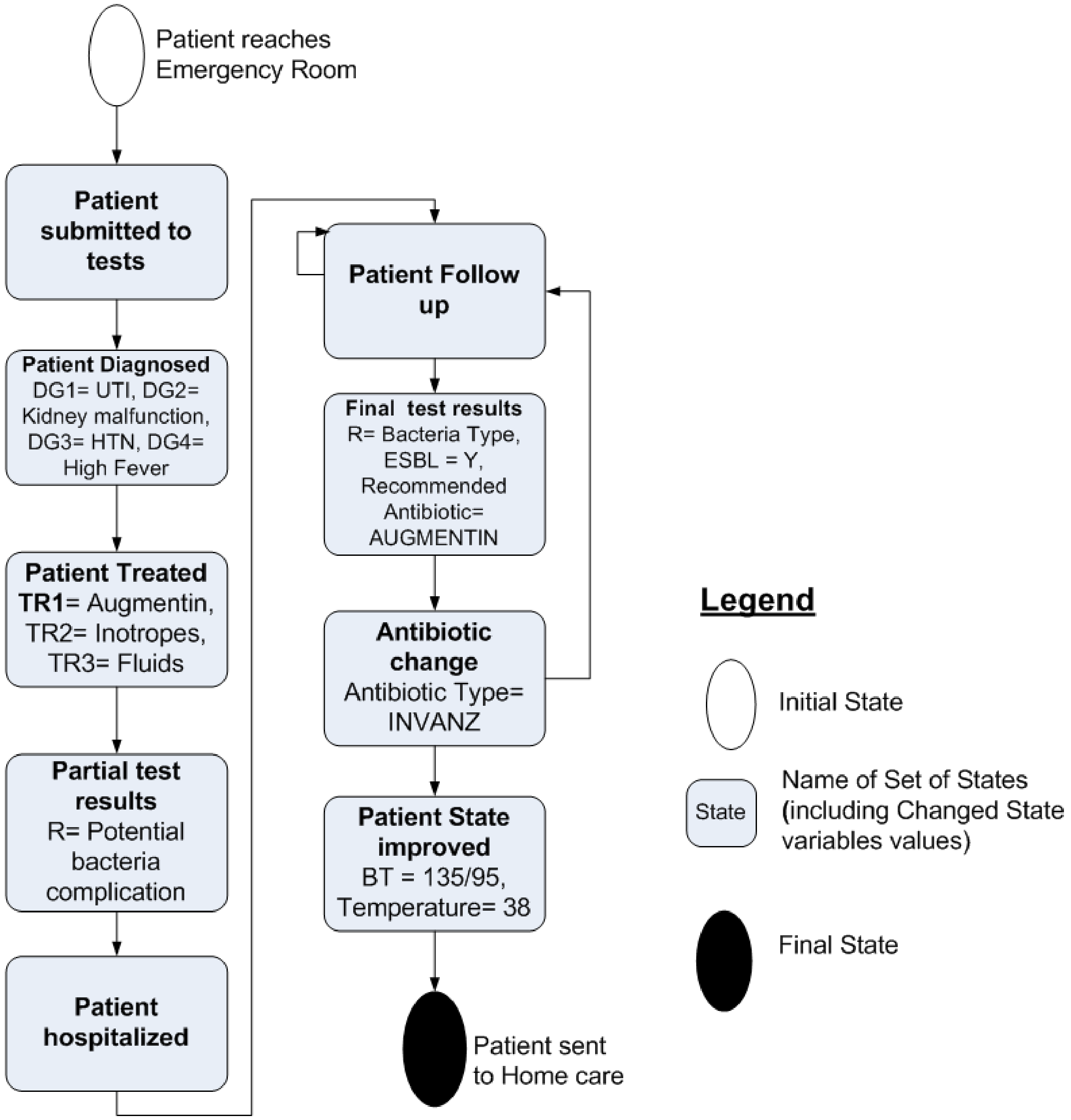
Wang, R., Strong, D. & Guarascio L. (1996). Beyond accuracy .What data quality means to data consumers. *Journal of Management Information Systems*, Spring 1996, Vol. 12, No. 4, pp. 5-34,1996.

Weber B.,Reichert M., Rinderle-Ma S. & Wild W. (2009). Providing Integrated Life Cycle Support in Process-aware Information Systems. *International Journal on Cooperative Information Systems* 18(1): 115-165,2009.

Wein, C.W. (2007). *Urology, 9thedition*. Elsivier,2007.

Zmud, R. (1978). Concepts, theories and techniques: An empirical investigation of the dimensionability of the concept of information. *Decision Design,* 1978, 9(2):187-195, 1978.


Historical Process instance data
I, X, P, G, SG
P, G, SG
1.Context Learning
2. Path learning
3. BP Model Adaptation
BPmodel Specification


Patient reaches Emergency Room
Patient submitted to tests
Patient Diagnosed DG1= UTI, DG2= Kidney malfunction, DG3= HTN, DG4= High Fever
Patient Treated TR1= Augmentin, TR2= Inotropes, TR3= Fluids
Partial test results R= Potential bacteria complication
Patient hospitalized
Patient Follow up
Final test results R= Bacteria Type, ESBL = Y, Recommended Antibiotic= AUGMENTIN
Antibiotic change Antibiotic Type= INVANZ
Patient State improved BT = 135/95, Temperature= 38
Patient sent to Home care
Legend
Initial State
State
Name of Set of States (including Changed State variables values)
Final State