Simulation of Patient Admission Process Using Colored Petri Net

Joseph Barjis[1], Sebastian Freund[2], Christof Schulze[2]
[1] Delft University of Technology
Jaffallan 5, 2628 BX, Delft, Netherlands
[2] Otto-von-Guericke Universität Magdeburg
39106 Magdeburg, Germany

Abstract

In this paper we apply modeling and simulation as an integrated approach to study the patient admission process in a hospital. The span of the processes is of intra and inter-organizational nature, which allows to see the intricacy of organizational processes on one hand, and the process-centric flow of activities on the other hand. This paper resulted from a student research project, where it was aimed to introduce a group of student-researchers to the ideas of business process analysis, design, modeling and simulation. The whole study was organized around the major of Information Systems and the role of business process modeling and simulation as a centerpiece in this major. So, this is not a research paper, but a case study that will be demonstrated.

1 Introduction

This paper is first and foremost about Information Systems in a broader sense: Information Systems Analysis, Design, Engineering, Application, Organizational Impact, etc. However, Information Systems do not operate in isolation; they are designed, developed, and deployed in organizational context (settings). Information Systems are designed for certain objectives and tied to specific organizational processes (situations), e.g., order processing, product development, patient admission and administration, student registration, and so on.

In its turn, an organization is a social system purposefully engineered to carry out a certain mission, thus adding the important human dimension to the equation. Collectively, the organization's business processes and their interrelationship that enable to deliver service to customers or produce goods make up the organization's business system. Hence, an organization is comprised of three major components: people (actors, users), processes (business processes and procedures), and structure. In modern organizations, these components are supported, enabled, linked and interwoven via information infrastructure (information systems, information technology, software applications) such as Enterprise Information Systems (e.g., Enterprise Resource Planning Systems, Human Resource Information Systems, Accounting Information Systems).

In its right, this paper echoes the recognition of the importance of the organizational context and enterprise scope, in which Information Systems are deployed, utilized and set. This leads to the recognition of the notion that information systems are not merely technical, but a socio-technical phenomenon, where organizations (people, process, structure) are integral to them.

Now, how we go about the study of these phenomena/concepts (Information Systems, organization, enterprise, business process). In order to study them, we need to use theories, methodologies, frameworks and concepts. Well, we can develop systems (in a smaller size, prototype) and study their behavior; we can build mathematical models and abstractions of the systems and study them; we can draw systems static pictures using diagrams and then study the diagrams. Each of these approaches has their advantages and disadvantages.

For example, building a system would be very expensive, time consuming, and especially risky if a wrong system is built and we will be forced to demolish it and build another one. Even if the system was built in a smaller proportion, it will be still a huge waste of investment (money, time, resources). Also when the system is built, it takes considerable time to proof whether it resembles the true process.

Similarly, mathematical abstraction will be on one hand very difficult, if not impossible, as we are dealing with informal reality, on the other hand for users, not having expertise in mathematical abstraction, it will be challenging to comprehend the models. An alternative approach, widely accepted, adapted and used, is modeling and simulation.

2 Modeling and Simulation

Modeling is creating a model of reality, which is an abstract representation of reality. It has specific borders and consists of a well defined set of elements. Simulation is a tool that enables its user to predict, what outcome a given change of a given model has. It provides a dynamic view by interconnecting static states of the model. The modeling and simulation theories, methodologies, and approaches allow to analyze, design and study processes and system using artifacts that are designed for this purpose (diagrams, notations, languages, tools).

Over the past decade, the modeling and simulation practice is becoming very popular, attractive, and widespread in the fields of information systems, organizational and enterprise processes. The are many reason for this popularity including (Aguila, Rautert & Pater, 1999; Carson, 2005):

- Modeling has a dynamic aspect illustrating process flow and changes over time.
- A model allows for experiments, changing small parts of the model, allows for fairly easy evaluation and comparison to different changes.
- Simulation enables expectation management and measuring or visualizing change impacts.
- Simulation also allows the use of animation “Don’t tell me, show me”, which is invaluable tool for communicating models to users.

- Simulations can create a performance analysis before dedicating resources to a project.

There are a variety of tools used for simulation. In recent years, Petri nets (different types) have been extensively used in business process modeling and simulation. Among different types, Colored Petri Net (CPN or CP Net) has a wide range of applications. In this paper, we demonstrate an example of patient admission process modeling and simulation using CPN.

When using Petri nets, the model has to introduce places, transitions, tokens and arcs, which are the basic elements of the Petri net language. Tokens move from input place to output places only by crossing transitions, i.e., transitions that are enabled fire and as a result tokens move to output places. Transitions and places are connected by directed arcs. Each item (arcs, transitions and places) can have constraints that determine the conditions for it to be active. The placement of tokens in the net represents the model state. The movement of tokens represents a transition from one state to another.

To apply Petri net for modeling and simulation patient admission process in a regional hospital, we created a number of tasks that will be explained in the following section:

1. **Transactions Identification:** Identify all business transactions.
2. **Business Process Model Construction:** Construct a business process model based on the identified transactions.
3. **CPN Model Design:** Design a CPN model based on the Business Process Model.

3 GMC Case Study – Transaction Identification

The GMC case example is taken from (Barjis, 2008).

3.1 GMC Background

Grand Medical Center (GMC), the busiest medical center for its size in the United States, is a six-building medical campus with 300 total beds. GMC saw 120,000 emergency room visits this past fiscal year, up 3 percent from 2005. To extend its services, in 2006, GMC planned a $93 million expansion to add 129 acute care beds in a five-story patient tower. In 2005, with over 4,000 employees and 750 physicians, the System provided care to almost 400,000 patients. In 2006, GMC received the HealthGrades® Distinguished Hospital Award for Clinical Excellence™, ranking among the top 5% of all hospitals in the United States for overall clinical performance. This recognition urged GMC to embark on extensive IT innovations including a state-of-the-art patient admission management system, discussed below.

3.2 Patient Admission Process

In order to determine correct requirements and build accurate models, the researchers spent several days observing the patient admission process. In addition, they studied

existing documents and conducted interviews with the nurses and other personnel. The following is a significantly abridged description of the patient admission process:
The admission process usually originates from a physician office. If decided to refer the patient to the hospital, the physician office calls the hospital's Admissions RN (Registered Nurse) to make arrangements, and contacts the patient insurance company, if any, to notify them of the admission request. In certain circumstances, the patient may need to be transferred to another hospital (3rd party provider). If transfer is the case, the transfer is arranged by GMC via ambulance and the patient's insurance will be charged for the transfer fees. In normal circumstances, the Admissions' RN arranges admission of the patient and notifies the physician office of the decision. After the admission is arranged, the patient arrives at the hospital to be placed in designated unit. Upon patient's arrival, the Admissions Clerk then obtains the patient's personal information and creates a new profile.
Once brought to the room, a Case Manager is assigned to the patient. The Case Manager does two things: verifies that the insurance company was notified of the patient's admission by the referring clinic and creates a new record in the hospital information system (HIS) for future reference; the Case Manager also calls the contact person at the insurance company and gives the clinical information needed to approve the patient's stay, including the bed type. The patient continues staying in the hospital until they get discharged.
After the patient is discharged, a claim is filed with the insurance company. If for any reason the patient's insurance company does not pay the patient's entire bill, the hospital will bill the patient directly. In the case a patient has problems paying the co-pay for the service, the patient can contact the Business Office at GMC and set up a payment plan. If a patient has no insurance, they must contact the Financial Councilor in the Business Office. The Financial Councilor interviews the patient and has them fill out forms for Medicaid and Medicare. The Financial Councilor contacts FairTrial, an outside agency, to conduct an investigation of the patient's financial status. If the patient is found to be at or below the federal poverty guideline, the hospital will write off the patient's bills. FairTrial, after receiving a request for a case, sends an established invoice to the hospital to pay their fee, before they complete their investigation.

3.3 Requirements Specifications

Each software application enables a certain process or processes. The purpose of requirements definition is to accurately capture all essential activities that should be realized as functionalities of the envisioned system to design an adequate software application. At least, these essential activities should lead to a profound prototype with minimum rework. By analyzing description of the patient admission process given above and applying the transaction concept, we identify each essential activity of the process as a transaction. On one hand, by a transaction we mean an activity that cannot be skipped if the condition for its execution is true, on the other hand, each transaction involves two agents/actors, an initiator and an executor. Based on thorough analysis of the above description, a number of transactions were identified (see Table 1). These transactions include both internal processes and inter-organizational processes. They collectively represent an enterprise system of complex inter-relationships.

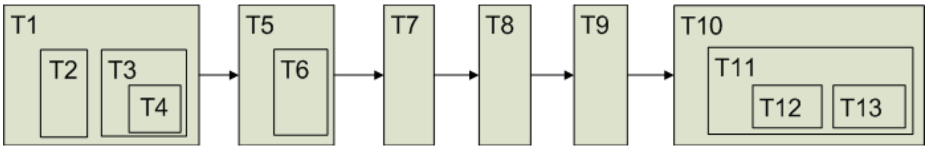
T1:	**Request Admission**
Initiator	Physician (Clinic Secretary)
Executor	Admission RN
Result	Admission request is granted
T2:	**Ask Insurance pre-Authorization**
Initiator	Physician (Clinic Secretary)
Executor	Insurance Company
Result	Pre-Authorization is issued
T3:	**Transfer Patient to Another Hospital**
Initiator	GMC (Admission RN)
Executor	Another Hospital
Result	Patient is transferred to another hospital
T4:	**Pay Transfer Fee**
Initiator	GMC (Business Office)
Executor	Insurance Company
Result	The transfer fee is paid
T5:	**Place the Patient into Corresponding Unit**
Initiator	Patient
Executor	GMC (Admission Clerk)
Result	The patient is paced into corresponding unit
T6:	**Create Patient Profile**
Initiator	GMC (Admission Clerk)
Executor	Patient
Result	The patient profile is created
T7:	**Arrange Patient's Stay Approval**
Initiator	GMC (Case Manger)
Executor	Insurance Company
Result	The patient stay is approved
T8:	**File Insurance Claims For Clearing**
Initiator	GMC (Business Office)
Executor	Insurance Company
Result	Insurance claims are cleared
T9:	**Bill the Patient for Service**
Initiator	GMC (Business Office)
Executor	Patient
Result	The service bill is paid
T10:	**Set Up Payment Plan**
Initiator	Patient
Executor	GMC (Business Office)
Result	A payment plan is set
T11:	**Investigate the Patient Financial Status**
Initiator	GMC (Business Office)
Executor	FairTrial
Result	The patient is investigated
T12:	**Pay the Investigation Fee**
Initiator	FairTrial
Executor	GMC (Business Office)
Result	The investigation fee is paid

Tab. 1. Patient admission and service transactions

Information of Table 1 helps to build a model that illustrates all the actors/agents and the activities they are responsible to perform.

4 Business Process Model Construction

The model was designed in a top-down approach. Using this approach, it is possible to have a rather simple view on the whole process that can be refined by zooming in to transactions on the top level. There are three layers of the hierarchy. Determining the structure of the model can be done by illustrating the nesting structure of the transactions and by showing which transactions are solely optional. The resulting Process model can be refined by adding Initiator and Executor to the graph and by splitting the transactions into execution, initiation and result phase, but we decided to do this not until modeling in CPN Tool.



Hierarchy of Transactions

5 GMC CPN Model Design

5.1 Construction of the Model

The main CP Net consists of transactions that are executed within GMC and the insurance company. Those are the locations where the top level transactions are take place. The locations are written on the top of every net. This is a way of keeping an overview of where which process takes place.
Text labels and many other items are docked at guidelines, which can be used to make positioning and moving objects easier. Changes in the model can be done much quicker when all items are attached to guidelines. They also provide the model with a good look and feel because each item is arranged vertically or horizontal to the other.

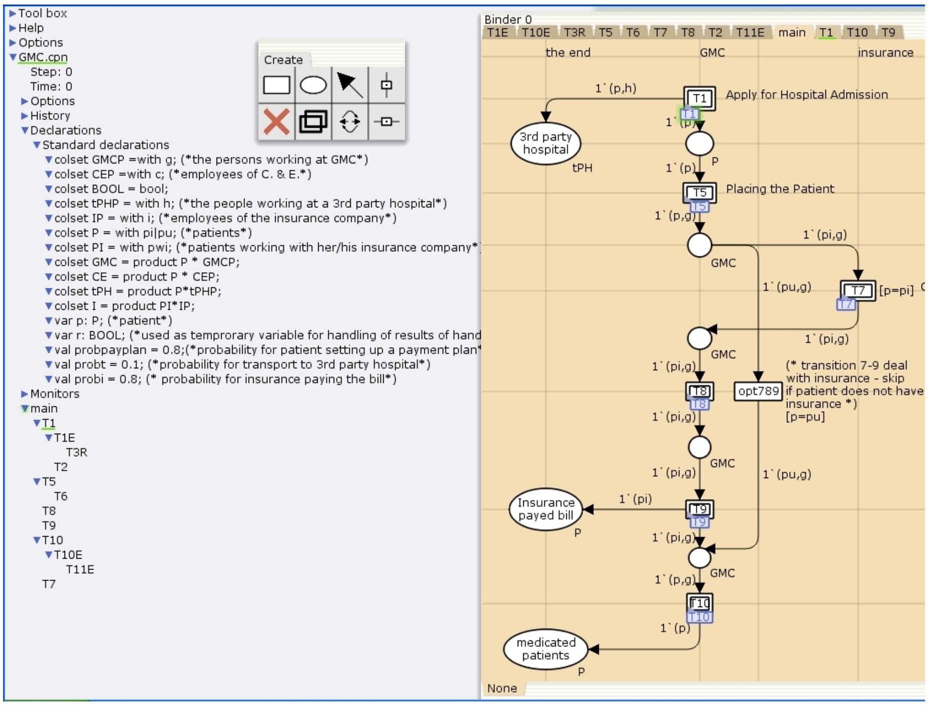


Fig. 1. Overview of the system

Constructing the CPnet is done in 2 phases. At first, the structure of the net is constructed on paper. In the second step, all declarations, place types and arc properties are added. This approach has the advantage that structural errors are recognized early and require less time to adjust. At first, the structure of the net was created on the right pane. When this was done, the left pane was used to add declarations. After adding declarations, the model was completed by adding the definitions and data types to every arc and place.

There are a few basic pre-defined data types (int, bool, enumerated, index). Custom data types may be defined by the user using the pre defined color sets (for example product, record or list).

Since the net is constructed in a top-down approach the structure of the main net was developed first. This is done by using transitions, places and arcs from the create-toolbox (see the left pane of Figure 1). In addition to the transitions representing a transaction, in some places there are transitions that enable tokens to skip a particular transaction in case it is an optional transaction. The next step is the creation of subpages to represent a transaction in more detail. This can be seen as overlapping tabs on the right pane of Figure 1.This detailed presentation should contain all phases of the transaction plus other nested transactions. To do so, one uses the “move a transition to a subpage” tool in the Hierarchy toolbox. This creates a subnet that can be seen under the actual net-icon in the list of all nets. The symbol of the transition turns into a rectangle with a double border. A transition having a subnet is also indicated by a light blue tag on its bottom showing the name of the subnet (e.g, T5, T7, T8). Subnets contain the input and output place and the transition that is to be replaced. Both places are the same as the input and output places on the higher level net. If a token arrives at the input place on the master net, it is also seen on the place in the subpage where it is passed on. Input and output places also can be created by using the *in* and *out* property of the hierarchy toolbox and linked with the fuse property. In this case the place colors and arc constraints need to be the same. That approach can be iteratively used to model all nested transactions up to any depth of layers.

When the structural model is created, one can go on creating the declarations and adding them to all arcs and places. Those are defined in the Standard Declarations toolbar. Any colset, variable, function and global reference can be defined at this place. For the GMC model colsets and variables are used to define everything that is needed. The most important item is the colset that identifies a patient: "colset P = with pi|pu;". The elements pi and pu represent insured and uninsured patients. The other basic colsets are: GMCP (the persons working at GMC), CEP (persons working at FairTrial), tPHP (people working at a 3rd party hospital) and IP (employees of the insurance company). To recognize the patients insurance status throughout the process, the patient token (and with it the information whether the patient has insurance or not) is handed through every stage of the model. This is done by using complex colsets consisting of the patient token and token of the specific institution handling the patient. Those are GMC, CE, tPH, and I (GMC, FairTrial, third party hospital and the insurance company). The last unmentioned item of the used declarations is the variable p that can be either pi (insured patient) or pu (uninsured patient). In this way, insured and uninsured patient tokens can be processed through the same logic. Once all declarations are set, they are available in the CP Nets. Now each arc needs the information of how many tokens it will create or destroy and of which colorset these tokens are. For example this is done by inserting "1`(p,g)". The "1" indicates one token being transferred. The "(p,g)" means that the token consists of a "p" and a "g" element. That indicates that the place connected to the arc must be of the colorset that is able to contain "(p,g)" elements. The used colsets depend on the location where a specific task takes place.

At several points in the model (e.g., see T10E at T10E1, presented in Figure 9) there are two possibilities (directions) where a token can move to. Since we need to be able to define a probability for the token for going to one way or the other, we defined a generic Boolean variable r that is used to hold the result of a Boolean evaluation, which in turn is a comparison of a uniform distribution between 0 and 1 and a probability variable – in this case probpayplan. The semantic of this variable (probpayplan) is the probability of a patient setting up a payment plan to make payment. The outgoing arcs from the transition with the action that assigns this truth value to r have some constraints: if r is true then one of the arcs can transport a patient token, otherwise it transports an empty token. The condition for the second arc is the opposite. This construct fulfills the purpose of being able to define the probability in the declarations and not somewhere else in the net.

Other variables for probability are used: probi and probt which are further explained in the comments of the net.

5.2 Description of the Model

The Main Model

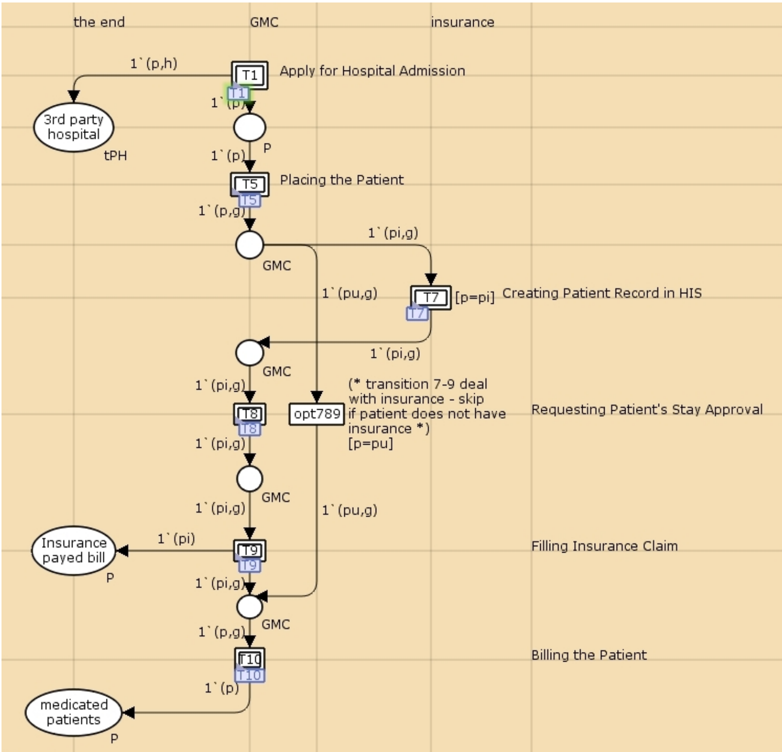


Fig. 2. The Main Model

The main CP Net (Figure 2) shows all un-nested transactions, all possible output places and indicates the position where the transition is executed. In the admission process T1 the patient can either be finally admitted or he is brought to a 3rd party hospital and its token ends in the according end place. After the admission the patient is placed during T5. In T7 a record in the insurance system is created. Since not everybody has insurance this transaction is optional. That is achieved by checking the colorset of the patient element in the token. If it is an uninsured patient, he skips every transactions dealing with the insurance company such as transactions T8 and T9. If the patient has insurance his record is created and the patients stay approval is requested in transaction T8 and the insurance claim is filled in transaction T9. In case the insurance pays the complete bill, the patient leaves the model. In the other case the patient is billed by GMC in transaction T10.

Transaction T1 :

The admission process T1 (Figure 3) includes several actions. Aside from the three steps of the transaction, it contains the starting place, nested transactions and two out-places. During the admission process the insurance has to be asked for pre-authorization in T2. This is just an optional path since this just has to be done for insured patients. The alternative route is opt2. During the execution phase of T1 the decision will be made whether a patient can be admitted or needs to be transferred to a 3rd party hospital. In case the patient is admitted, he leaves this subnet on the left side with T1 completed. Otherwise he ends in the right out place that illustrates the 3rd party hospital.

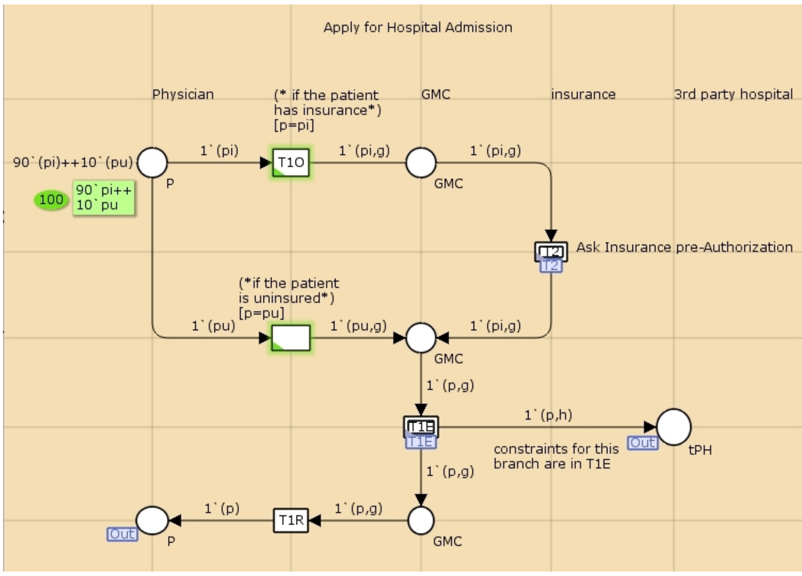


Fig. 3. Transaction T1

During the execution phase of T1, GMC has to decide whether a patient can be admitted or needs to be transferred to another hospital. T3 represents the transfer of the patient that is initialized at GMC and executed at a 3rd party hospital. After completing T3 the patient token ends in an end state.

Transaction T2 :

The subnet of Figure 4 represents the three phases of transaction T2. It starts at GMC and is executed in the insurance company.

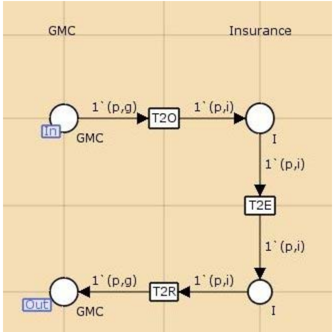


Fig. 4. Transaction T2: execution phase

Transactions T3 and T4:
In the subnet shown in Figure 5, the billing of the patient is modeled. Since GMC charges patients for the transfer, transaction T4 is starting and ending at GMC.

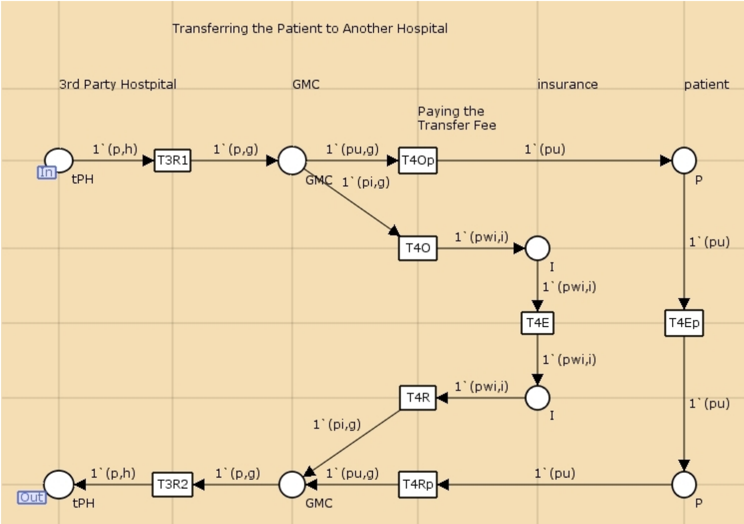


Fig. 5. Transactions T3 (result phase) and transaction T4

Transaction T5 and T6:
After admitting a patient, the patient is placed (Figure 6). During the process of the placement the patient's profile is created in transaction T6. Since the patient provides the profile information he is the executor of this transaction.

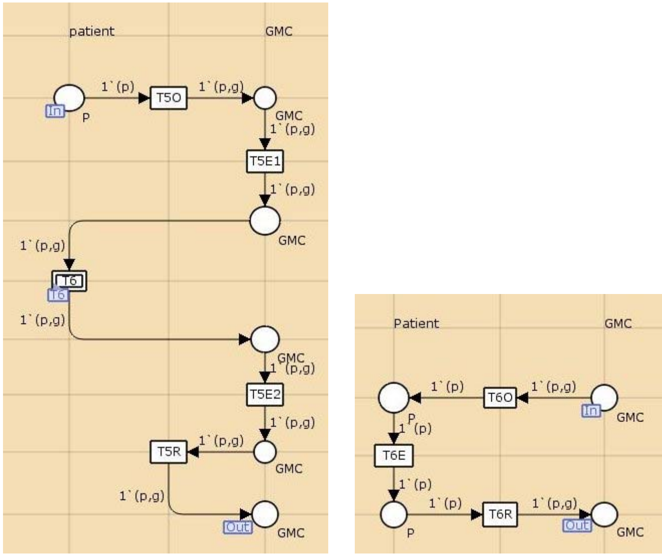


Fig. 6. Transactions T5 and T6

Transactions T7 and T8:
After the patient is placed, the record in HIS has to be updated or created in case it does not exist (Figure 7). When the patient's record is up to date, the insurance is asked for the stay approval.

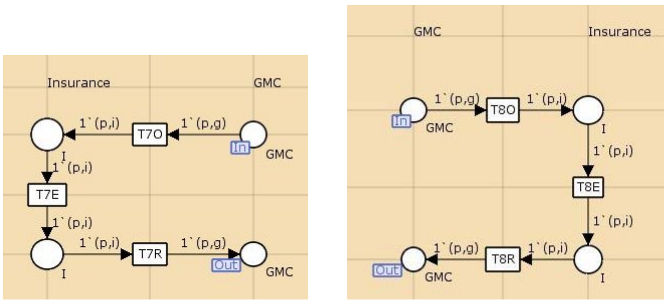


Fig. 7. Transactions T7 and T8

Transaction T9 :
After the treatment the billing process starts. This subnet (Figure 8) has two output places. If the token reaches the left output place, the payment was completely covered by the insurance and the patient token is at an end place of the model. In the other case the bill was not paid by the patient's insurance completely. So the patient goes on in the model by moving to the output place on the bottom.

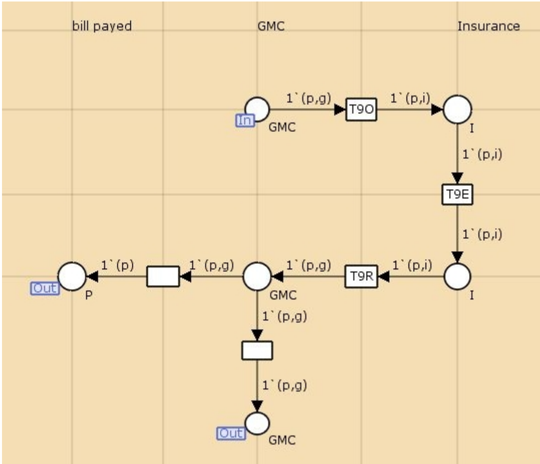


Fig. 8. Transaction T9

Transaction T10 :
In transaction T10 (see Figure 9) the process of billing the patient is executed. After finishing it the patient moves to the last possible end place of the model.

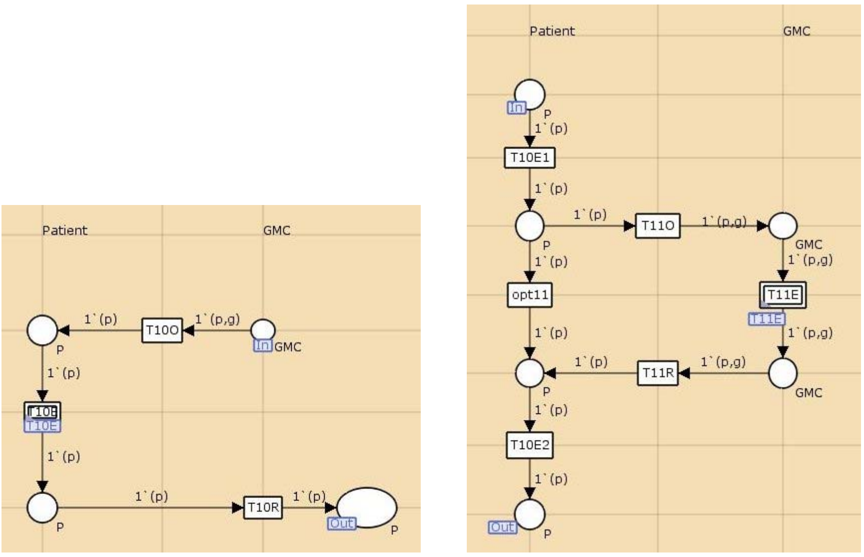


Fig. 9. Transaction T10 and its execution process

During T10E the patient has two options. If he can pay the bill immediately he passes the execution phase of T10 by skipping T11 and passing opt11. In the case he can not pay the bill completely he has to set up a payment plan with the GMC business office which is done in T11.

Transactions T11 and T12 :
In the transaction setting up a payment plan, the GMC business office conducts an investigation of the patient's financial status during T12 (Figure 10). This transaction is started at GMC and executed at FairTrial. When starting the investigation the payment process T13 is initiated and executed.

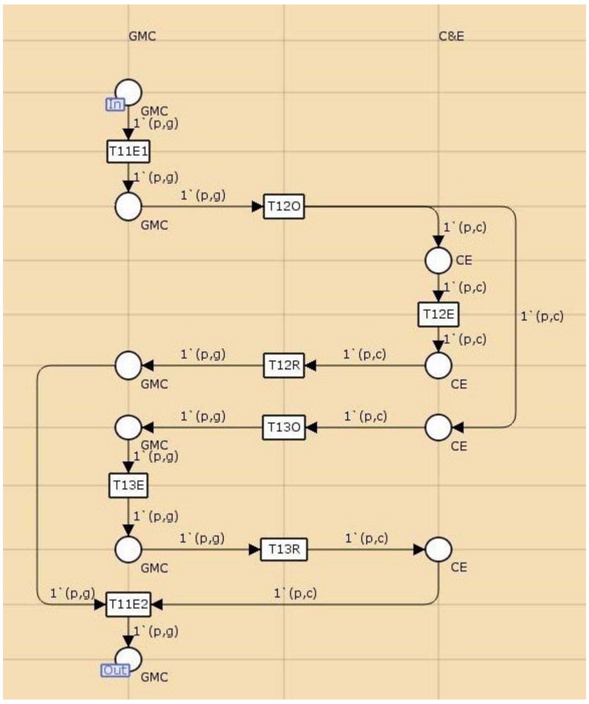


Fig. 10. Transaction T10 and its execution process

6 Conclusion

After working through our list of tasks, (identify the business processes, create a hierarchy, create a Petri net resembling those processes, adding constraints) we ended up with a Petri net model that can be used to simulate business reality. With proper distribution values and timing information, performance analysis is possible. Using this simulation it is not only possible to find more effective ways of organizing the workflow in the hospital, it is also possible to predict the outcome of specific changes.

Also this is a good way of documenting and educating new workers at GMC. They do not have to read documents but can see the process in the diagram and while the simulation is running they can get a dynamic overview on the process.

Communicating this model to other analysts is important to us, because by that we might find further improvements and share our findings. Having used CPN Tool was a challenge for us because it meant to learn the ways of this piece of software. Interesting enough just the most basic features of this software are documented in a user friendly way. For more advanced usage, the best resource still is the mailing list. By writing this document and providing an example that uses more than basic Petri net features, we are hoping to improve this situation.

Reference

1. Aguilar, M., Rautert, T., Pater, A.J.G., (1999) Business Process Simulation: A Fundamental Step Supporting Process Centred Man-agement
2. Barjis, J. (2008). The Importance of Business Process Modeling in Software Systems Design. Science of Computer Programming, Vol. 71, N 1, pp 73-87 (2006)
3. Carson, J.S., (2005) Introduction to Modeling and Simulation.