

## Improve Test Results by Piloting: How to Simulate Field Data in the Lab

**Gerald Gendron**  
Booz Allen Hamilton  
Norfolk, VA  
gerald.gendron@gmail.com

**Tammy Crane**  
Scientific Research Corporation  
Virginia Beach, VA  
tcrane@scires.com

**Gerrod Seifert**  
Booz Allen Hamilton  
Norfolk, VA  
seifert\_gerrod@bah.com

### ABSTRACT

There are many misconceptions about the benefits and application of pilot studies, but conducting tests without performing them in advance can be a costly mistake. One misconception is that access to test assets is necessary. Although it can be challenging to run a pilot study without the actual resources used in the test, simulation provides a means to overcome this and validate the effectiveness of the intended design. This paper presents the approach and results from a pilot study enabled through simulation. The project underlying this study is a test event investigating process performance. The team realized the value of incorporating a pilot study, but access to the systems and operators was not possible. They innovated a technique by combining Markov transition state modeling with process mining footprints to generate simulated data. Although simulated, its realism proved sufficient to inform a data collection plan optimized for a small team, and it indicated the main test was feasible. By analyzing the resulting data against investigative questions, the data collection strategy was refined in ways unforeseen before simulating the process. The pilot study clarified assumptions, refined instrumentation, and assessed the utility of expected data. These results demonstrate the power of pilot designs, even when resources are not available. This paper closes with insights extensible to other types of analyses and particularly those involving testing and process mining.

*Keywords: pilot study, domain expert, Markov model, simulation, process mining, process modeling, synthetic data*

### ABOUT THE AUTHORS

**Jay Gendron** is a data scientist with Booz Allen Hamilton. He is a business leader, artist, and author who writes on perspectives of how good questions and compelling visualization make analytics accessible to decision makers. He is an award-winning speaker who has presented internationally. Jay volunteers with Code for America - contributing data science skills to improve civic and municipal access to data. He is the founder of the Meetup "try.Py - Learn Python". Jay is authoring a book entitled *Introduction to R for Business Intelligence* due for release this summer by Packt Publishing this summer. For additional information, please visit <https://www.linkedin.com/in/jaygendron>.

**Tammy Crane** is an aeronautical engineer and operations research systems analyst at SRC. She uses engineering and data analytics expertise to improve human performance in scenario-based M&S training, electronic warfare, weapons testing, and wargaming. She often advises decision makers on requirements analysis, project management, and risk identification. Tammy volunteers with the Society of Women Engineers and endeavors to research: the science and cognition effects on decision-risk analysis, and advanced experimentation for immersive training environments, future sensors, and autonomous systems. Connect with Tammy at <https://www.linkedin.com/in/tammycrane>.

**Gerrod Seifert** is a senior defense research analyst working for Booz Allen Hamilton. He served in the U.S. Navy as a Surface Warfare Officer. Gerrod is experienced in joint integrated air and missile defense and defense acquisition operational testing and evaluation. He has an M.S. in Natural Resources (Hydrogeology), and is a certified Project Management Professional (PMP). Gerrod is pursuing a Ph.D. in Environmental Engineering to improve water resources. He co-authored the evaluations of drinking water and wastewater assets in the 2016 Report Card for D.C.'s Infrastructure (American Society of Civil Engineers, 2016).

## Improve Test Results by Piloting: How to Simulate Field Data in the Lab

**Gerald Gendron**  
**Booz Allen Hamilton**  
Norfolk, VA  
gerald.gendron@gmail.com

**Tammy Crane**  
**Scientific Research Corporation**  
Virginia Beach, VA  
tcrane@scires.com

**Gerrod Seifert**  
**Booz Allen Hamilton**  
Norfolk, VA  
seifert\_gerrod@bah.com

### BACKGROUND AND CONTEXT

This paper presents results of how a simulated pilot study aided a team plan for a test and evaluation event. The team had no access to field data and no resources to conduct a formal pilot event prior to the main test. Recognizing the inherent risks of entering a test “cold”, the team innovated a technique to develop a synthetic dataset that for use as if it were actual field data. The project focused on testing standard operating procedures that are process-oriented, which require the careful completion of a series of activities to attain a desired end state. Testing the process will involve the observation and recording of human-executed procedures because it has no automated feedback and control systems. The value of this paper’s contributions is in the innovations, approaches, and results it presents. This paper purposely presents the project generally in order to encourage the extensibility of this work to other process-centric tests or studies.

### Testing a Process

Testing the performance of standard operating procedures is challenging because one must evaluate a process subject to human interpretation and execution of written instruction. Coincidentally, human judgment is an expected activity in the completion of this process. Therefore, the test must be able to discern whether an outcome results from discrete, logical procedures or human execution of those procedures. In this project, the system under test focused on

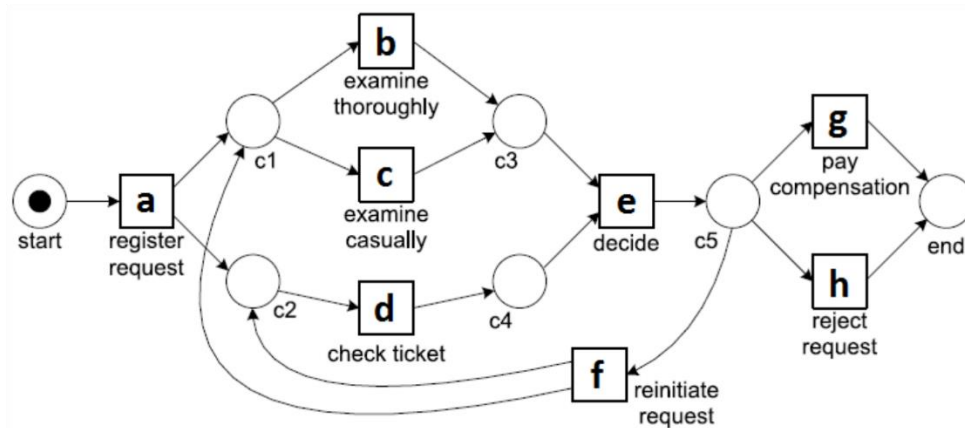
- a process
- relying on the transfer of data
- involving geographically separated work units.

The test will investigate the effectiveness of the standard operating procedures and the results of human activity in a time-sensitive environment. Test professionals rely on non-subjective measures to analyze and evaluate the system under test, which in turn require data. Pilot studies are important because they help determine if the data are objective and sufficient to conduct analysis and evaluation.

### Processes and Process Modeling

The online Oxford Dictionaries defines *process* as “a series of actions or steps taken in order to achieve a particular end” (Process, n.d.). The human experience is full of processes. In fact, the human experience is itself a process (e.g., birth, grow, age, death). The many processes surrounding us come from any imaginable domain - medical, social, financial, engineering, artistic, management, to name a few. This paper resulted from a creative process. The team decomposed the end state - this paper - into steps: ideate, storyboard, assign writing sections, draft, refine, monitor progress, finalize, and submit.

Aalst (2011), a leader in the field of process modeling, developed a branch called process mining. Process modeling is a structured method of describing a process as activities and relationships to one another. The relationships are sequences of activities traversed in order by causal dependencies. He notes, “The process model may also describe temporal properties, specify the creation and use of data, e.g., to model decisions, and stipulate the way that resources interact with the process (e.g., roles, allocation rules, and priorities)” (2011, p. 4). Figure 1 presents a process adapted from Aalst (2011) and augmented to label each activity *a* through *h*. This example process uses a model notation style called Petri Nets; however, there are dozens of industry-recognized notations. Process mining techniques can automated the generation of Petri Nets by analyzing event logs from actual process data to determine the order, sequences, and causal relationships.



**Figure 1. Petri Net. A method of communicating and analyzing process models (Aalst, 2011).**

Models help people think about the process and articulate analyses results. There is a cautionary note, however. Box (1979) famously quipped, “All models are wrong but some are useful”. Analysts are wise to take into account the difficulties in modeling anything. Regards process modeling, Aalst highlights three warnings one may encounter when working through models:

- The model describes an idealized version of reality
- Inability to adequately capture human behavior
- The model is at the wrong abstraction level (2011, p. 30)

Nonetheless, Box acknowledges that models are useful. One thing the team held paramount in its approach to designing a test was that domain expertise was an extremely important element to create a better process model and increase outcome success.

### **Role of Domain Expertise**

The team consisted of not only analysts but also domain experts from the field. They worked together to develop the measures to evaluate the procedures. This is an important point. Give a data science professional a dataset and to be sure, the analysts will find "something". However, will they find something relevant to the evaluation? Their intuition may produce useful data, but adding domain expertise helps extract relevant insights from that data.

During this project, the design of the pilot study relied upon the knowledge and experience of the domain expert to provide the context of the roles, information analysis, and decision-making required of the people involved with the process under test. Throughout this project, the analysts relied on domain experts to share knowledge of the process.

They worked together to understand the sequence of expected operator activities, the information flow, and the process structure. Domain expertise provides the necessary insights to the transitions within the process.

## **SIMULATING PILOT STUDIES**

Pilot studies exist in many disciplines ranging from engineering to medical research, the social sciences, finance, and manufacturing. Piloting is a method to assess the feasibility of a practice or study under conditions of insufficient information to make decisions about funding, resources, schedule, and risk. A pilot study is an investigation or exploration to demonstrate that the methods and procedures envisioned for the main study are viable for use at a larger scale (Thabane et al., 2010). An African proverb captures the essence of pilot studies, “You never test the depth of a river with both feet” (Thabane et al., 2010, p. 1). Baker (1994) notes many reasons for conducting pilot studies:

- Developing and testing adequacy of research instruments
- Assessing the feasibility of a main study
- Assessing whether the research protocol is realistic and workable
- Assessing the likely success of proposed methods
- Identifying logistical problems which might occur using proposed methods
- Estimating variability in outcomes to help determining sample size
- Collecting preliminary data
- Determining what resources are needed for a planned study
- Assessing the proposed data analysis techniques to uncover potential problems

The goal of the team’s simulated pilot study was to assess the feasibility of the research questions and determine if analytical methodologies were appropriate to the main study in the test plan.

### **Challenges**

Field data were not available to inform the team as to the actual process under study. Additionally, the team had no access to those who would execute the procedures during the test. The primary focus of a pilot study is feasibility assessment. Issues for consideration include reliability of the outcome, feasibility of measurement, process challenges, resource constraints, management issues, and scientific treatment (Lancaster 2004; Thabane et al., 2004). It is common to conduct pilot studies to gather data for calculating sample sizes, particularly where data from previous studies do not exist to inform the process (Thabane et al., 2010). This was also a consideration in this simulation.

### **Options Available for the Project**

Innovative teams can find multiple options to surmount technical challenges. In this project, the team identified three options:

- Option A - design the main test without conducting a pilot study
- Option B - convene a group of experts to aid in designing the main test
- Option C - simulate synthesized data representative of field data typical from a pilot study

The first two options did not allow the team to assess feasibility of the research questions sufficiently. They researched the literature for methods to simulate and validate synthetic field data. Successful use of simulation

appeared in case studies like the development of the Boeing 777. Nearly all of the design work occurred computationally. This not only avoided cost, but also helped determine design feasibility. It increased the fidelity of interface data so the modules integrated easily after fabrication (Kossiakoff, Sweet, Seymour, & Biemer, 2011). Likewise, in the medical field, Barhak (2015) discusses generating individual records using Monte-Carlo techniques to create study populations for use in chronic disease studies. Although there are no people involved, the data generated for each individual in the synthetic population abides by rules of inheritance seen in actual populations of individuals. Researchers can use these populations to conduct sensitivity analysis or hypothesis testing.

After the literature review, the team selected Option C to simulate synthesized data representative of field data typical from a pilot study. The team recognized the pilot data would need to be representative of the main test. Therefore, the team developed clearly defined simulation project objectives. Table 1 shows the objectives established for the Pre-Test phase. The figure also shows how these support the activities during and after the test.

**Table 1. Simulation in Testing. A three-phased approach integrating simulated piloting with testing.**

<b>Pre-Test</b>	<b>During Test</b>	<b>Post-Test</b>
<ul style="list-style-type: none"> <li>• Simulate event logs</li> <li>• Create process from event logs</li> <li>• Inform possible data collection nodes from simulation</li> <li>• Decide on sampling strategy</li> <li>• Outline analytic structure</li> </ul>	<ul style="list-style-type: none"> <li>• Create data collection logs from sample</li> <li>• Use expert observations</li> <li>• Apply data collection surveys and interviews</li> </ul>	<ul style="list-style-type: none"> <li>• Examine information process flow</li> <li>• Analyze key process activities</li> <li>• Compare predicted to actual outcomes</li> </ul>

### **Simulate Event Logs**

As previously mentioned, there is no existing data on the process under test. The simulation would produce event logs that could in turn be processed using process mining techniques to generate a process mapping for analysis.

### **Inform Data Collection**

The team needed to understand the test elements in order to generate a plan for the main test. The process mapping and analysis of synthetic field data would show which data elements were required and where to collect them.

### **Sampling Strategy**

Having a sense of data requirements, a sampling strategy would be developed and checked against runs of the simulation to assess sample size and assess how missing data in the synthetic data set would affect the analysis.

## **HIGHLIGHTS OF THIS SIMULATED PILOT STUDY**

The team focused on the challenge of simulating the information process flow, executed primarily by humans, to determine the feasibility of gathering these types of data. The pilot study involved simulation of a Markov transition matrix. Simulation offered the benefits of low cost and completeness because the team could code it themselves and the team's domain experts had sufficient knowledge to develop models and validate them. This section presents highlights of the approach used in the pilot study, the pseudocode used to create the simulation, and discussion of the steps used to refine the synthetic dataset.

### **Markov Models and Process Mining**

The process under test contains three states: data transmission, data receipt, and alternate means to retransmit data. A Markov model depicts states. Rabiner (1989) defines a Markov model as a “system which may be described at any time as being in one of a set of  $N$  distinct states,  $S_1, S_2, \dots, S_N$ ” (p. 259). Moreover, for observable Markov models, one can postulate the process is in some state  $S_t$  at time  $t$  and will be in another state  $S_{t+1}$  at time  $t+1$ . State transition probabilities describe the likelihoods of moving among states in a Markov model. Figure 2 depicts the process under study expressed as a Markov Model. The rows represent the state  $S_t$  and the columns represent the state  $S_{t+1}$ . The cells contain the probabilities associated with moving from state  $S_t$  to state  $S_{t+1}$ , as  $p(S_t, S_{t+1})$ .

	Transmit ( $t$ )	Receive ( $r$ )	Alternate ( $a$ )
Transmit ( $t$ )	$p(tt)$	$p(tr)$	$p(ta)$
Receive ( $r$ )	$p(rt)$	$p(rr)$	$p(ra)$
Alternate ( $a$ )	$p(at)$	$p(ar)$	$p(aa)$

**Figure 2. Markov Model. Notional depiction of state transition probabilities for the pilot project.**

Aalst (2011) represents process models using footprints that have characteristics similar to a Markov model. Figure 3 shows the footprint representation of the Petri Net process model presented in Figure 1. The notation indicates the types of transitions possible from one activity to another. The forward arrow ( $\rightarrow$ ) and backward arrow ( $\leftarrow$ ) indicate a one-way transition is possible between states. Parallel lines ( $\parallel$ ) indicate a transition is possible in either direction. A hashtag (#) indicates that no transition is possible between those two states at any time.

	$a$	$b$	$c$	$d$	$e$	$f$	$g$	$h$
$a$	#	$\rightarrow$	$\rightarrow$	$\rightarrow$	#	#	#	#
$b$	$\leftarrow$	#	#	$\parallel$	$\rightarrow$	$\leftarrow$	#	#
$c$	$\leftarrow$	#	#	$\parallel$	$\rightarrow$	$\leftarrow$	#	#
$d$	$\leftarrow$	$\parallel$	$\parallel$	#	$\rightarrow$	$\leftarrow$	#	#
$e$	#	$\leftarrow$	$\leftarrow$	$\leftarrow$	#	$\rightarrow$	$\rightarrow$	$\rightarrow$
$f$	#	$\rightarrow$	$\rightarrow$	$\rightarrow$	$\leftarrow$	#	#	#
$g$	#	#	#	#	$\leftarrow$	#	#	#
$h$	#	#	#	#	$\leftarrow$	#	#	#

**Figure 3. Process Footprint. Used in process modeling.**

The team innovated a technique to generate synthetic event logs by blending a) the directionality of the process footprint with b) the stochastic nature of Markov models and c) state transition probabilities. This development by the team results in a probability-based process footprint and is a hybrid designed for use in process simulation. Figure 4 provides an excerpt of the project process footprint and its transformation into a Markov-like state transition matrix. It is not a true Markov model because it does not model processes that converge to a steady state

	$a$	$b$	$c$	$d$	$e$	$f$	$g$	$h$	$i$
$a$	#	$\rightarrow$	#	#	#	$\rightarrow$	$\rightarrow$	#	$\rightarrow$
$b$	$\leftarrow$	#	$\rightarrow$	$\rightarrow$	#	$\rightarrow$	$\rightarrow$	#	$\parallel$
$c$	#	$\leftarrow$	#	$\rightarrow$	$\rightarrow$	$\parallel$	$\parallel$	#	$\rightarrow$
$d$	#	$\leftarrow$	$\leftarrow$	#	$\rightarrow$	$\parallel$	$\parallel$	#	$\rightarrow$
$e$	#	#	$\leftarrow$	$\leftarrow$	#	$\parallel$	$\parallel$	#	$\rightarrow$
$f$	$\leftarrow$	$\leftarrow$	$\parallel$	$\parallel$	$\parallel$	#	$\rightarrow$	$\rightarrow$	$\rightarrow$
$g$	$\leftarrow$	$\leftarrow$	$\parallel$	$\parallel$	$\parallel$	$\leftarrow$	#	$\rightarrow$	$\rightarrow$
$h$	#	#	#	#	#	$\leftarrow$	$\leftarrow$	#	$\rightarrow$
$i$	$\leftarrow$	$\parallel$	$\leftarrow$	$\leftarrow$	$\leftarrow$	$\leftarrow$	$\leftarrow$	$\leftarrow$	#

	$a$	$b$	$c$	$d$	$e$	$f$	$g$	$h$	$i$
$a$	#	0.80	#	#	#	0.09	0.09	#	0.02
$b$	#	#	0.42	0.25	#	0.13	0.15	#	0.03
$c$	#	#	#	0.02	0.13	0.40	0.40	#	0.05
$d$	#	#	#	#	0.15	0.40	0.40	#	0.05
$e$	#	#	#	#	#	0.25	0.30	#	0.45
$f$	#	#	0.09	0.09	0.02	#	0.02	0.02	0.13
$g$	#	#	0.09	0.09	0.02	#	#	0.15	0.64
$h$	#	#	#	#	#	#	#	#	0.05
$i$	#	0.05	#	#	#	#	#	#	#

**Figure 4. Process Footprint (left) to Transition Matrix (right) excerpts. The result of transforming the process mining approach with Markov models.**

based on initial conditions. True Markov models result in a vector representing the expected values of the state distribution after repetitive cycles sufficient to attain steady state. Rather, the team’s hybrid approach uses properties of Markov models - transition probabilities - to inform a single pass through a process. Unlike Markov models, which revise their state probabilities over time, the probability-based process footprint does not change over time.

Figure 4 also shows that in transforming the footprint to a transition matrix, there is not a one-to-one replacement of possible moves on the footprint into transition probabilities. This is because the footprint shows both forward ( $\rightarrow$ ) and retrograde ( $\leftarrow$ ) movements. State transition matrices do not have probabilities associated with having been in another state. They focus on probabilities from time  $t$  to  $t+1$ .

## The Approach

Rabiner (1989) used Markov models in the field of speech recognition, which he defined as a process. His work rather nicely explains the application of Markovian approaches for process modeling. Rabiner solved three fundamental problems necessary to use Markov models for the speech recognition process:

- evaluating probabilities of a sequence,
- determining the appropriate sequence states, and
- adjusting parameters of the model to better explain real world.

Those solutions extend to this project. Having devised a hybrid technique of process footprints enhanced by a Markov model, the approach to modeling a pilot design enabled through simulation relied on a development process characterized by three attributes.

1. Engage domain experts early. A critical lesson of analysis is to understand the problem with the aid of domain experts. In this case, operators familiar with the process under study enabled the project team to capture key elements likely to influence the assumptions coded in the simulation.
2. Use an incremental development style. This attribute appears fundamental, but it is important. When writing the simulation code, the team wrote it in a building-block approach – implementing a step and checking it. The project used two pieces of open-source software for its development environment: Python (Python Software Foundation, 2015) and PRoM Process Mining (Process Mining Group, 2015).
  - Python generated the simulated data and converted it into XML format
  - PRoM executed the XML to simulate a stochastic process flow for analysis. This resulted in descriptive statistics for process flow (i.e., average, min, and max), variance, and choke points
 Using Python resulted in faster development cycles because of the code interprets and executes at run time. This is not the case in compiled languages like C++ or Java. Traces in the code allowed developers to immediately see impacts of the incremental build and adjust at that time.
3. Iterate between domain engagement and incremental development. Incremental development based on domain wisdom allowed the domain experts to get a feel (a proto-validation) of how well the developing code base was representing the process.

Overall, this approach fit the team's needs, personality, and resources. Reflecting upon this approach, it was an important contribution to the overall success of implementing a usable simulation.

## Implementation of Simulation

The simulation generated synthesized data representative of field data typically gathered during a pilot study. The data was nothing more complex than traverses (called traces) through an operational process. Data collected manifest itself as a sequence of letters representing which process activities executed in which order during a particular trace. The simulated process possessed these characteristics:

- Sixteen discrete activities, referred to by the letters *a* through *p*
- The process must begin with activity *a* and end with activity *p*
- Activities execute according to probabilities of a Markov model
- Activities may appear multiple times or not at all in any trace sequence

In real execution, activities will depend on peoples' actions at multiple locations. They may occur in parallel or in a serial fashion. A cue initiates each activity in the process. The simulation gathered traces in order to determine descriptive metrics of process performance, variations of traces through the process, and conformance of the process as compared to the objective process model. The Python codebase is available for review and collaboration at the Git repository found at <https://github.com/jgendron/ProbabilityProcessFootprint> (Gendron, 2016). Here is the pseudocode for the simulation algorithm based on standard lexicon as presented by Dalbey (n.d.).

```

INIT Traces to empty list
SET Paths equal footprints possible in Markov transition state matrix
SET Probabilities equal probability for each state transition
SET Start to A
SET End to P
SET Event to Start
SET Replication to number of simulated traces desired
FOR 1 to Replication size
    INIT Trace to empty list
    STORE first element of Trace as Start
    WHILE Event NOT EQUAL End
        OBTAIN Event as randomly drawn element from Paths
        APPEND Event to Trace
    ENDWHILE
    COMPUTE string transformation of list Trace
    APPEND Trace to Traces
    SET Event to Start
ENDFOR
COMPUTE frequency of each occurrence of Traces
WRITE list of frequencies to file

```

### Refining the Synthetic Data

Capturing the simulated process proved challenging - requiring multiple refinements to the simulation until the domain expert's instinct could validate an output process mapping. The key elements of the solution were the Markov transition footprint stored as a Python dictionary data structure and a process flow simulator using a random number generator to pull samples from that dictionary. A reasonable question for readers unfamiliar with the Python language is, "How random is the pseudo random number generator in the Python library?"

Python uses the Mersenne Twister as the core generator. It produces 53-bit precision floats and has a period of  $2^{19937}-1$ . The underlying implementation in C is both fast and threadsafe. The Mersenne Twister is one of the most extensively tested random number generators in existence (Python Software Foundation, 2016, para. 4).



Additionally, the team conducted its own validation of the random samples extracted from the transition dictionary and found them to be satisfactory for project needs. The refinement and validation of the Markov transition occurred over four iterations:

- The first iteration used simplified Markov transition probabilities as a proof of concept. They uniformly distributed across all possible outcomes. For instance, if there were four possible states from time  $t$  to time  $t+1$ , then the initial probabilities were each set to 0.25. The results were not realistic, but they validated process traces in order to verify no critical errors existed.
- The second iteration took a more thoughtful approach. The domain expert considered each of the states and revised the probabilities. This iteration was perhaps the most surprising of them all. It demonstrated how reality could mismatch performance based on a non-data centric weighting of possible outcomes. The transitions deviated greatly from what the domain experts expected to see in actual operations.
- The third iteration provided the largest improvement. It focused on refining transition probabilities to produce reasonable process traces. As the simulation ran, it captured and aggregated all instantiations of traces. The team reviewed outliers manifest as the most common and least common frequencies. This allowed the domain expert to see the most and least common types of process flows.
- The last iteration proved to be the most important. The domain expert altered the transition matrix by removing unallowable transitions and greatly reducing probabilities on extremely unlikely transitions. The result was a Python dictionary containing process flows that the domain expert could review for frequency as well as logic flow.

Having no prior process data, there was no gold standard to compare to the process simulation results. Domain experience was the best available validation of the simulated pilot. The data indicated the model was running well and provided the necessary data.

## RESULTS AND EXTENSIBILITY

Pilot studies result in an assessment of test or study feasibility as well as insights to adjust goals and ensure scientific rigor. The team developed research questions to investigate analytical methodologies before choosing to pilot. A logical plan to apply the results of this methodology and analytical rigor to the larger study would follow. The team's developments are extensible to other types of analyses, particularly in tests involving process flows or process mining where event log data does not exist prior to the test.

### Pilot Study Results

The criteria for pilot study success is assessing feasibility. According to Thabane, et al. (2010), these objectives provide the basis for results interpretation and making a decision whether to proceed to the larger study. The team recognized four options available in light of the situational context:

1. Stop: main study not feasible
2. Continue, but modify protocol: feasible with modifications
3. Continue without modifications, but monitor closely: feasible with close monitoring
4. Continue without modifications: feasible as is (Thabane et al., 2010)

Although simulated, the synthetic field data proved sufficient to inform a data collection plan optimized for a small team. The outcome of the simulated pilot study was successful and the team decided to continue without modifications. This gives indications the three-phased approach in Table 1 is feasible. The pilot study success is due

to the knowledge and understanding provided by the domain expertise working in concert with analysts. The combination of domain expertise, statistical, and computational skills resulted in a simulation that produced synthetic data of sufficient quantitative rigor to substitute for otherwise unavailable field data. The pilot study clarified assumptions, refined instrumentation, and assessed the utility of anticipated data.

### **Clarified Assumptions**

Many assumptions changed during the pilot study. For example, prior to piloting the process flow, the domain expert provided estimated probabilities of moving among feasible states in the Markov matrix. Simulation results from the pilot study indicated errors by as much as 500 percent. A known limitation of human cognition is forecasting. Once the domain expert reviewed the iterations of pilot results, the transition probabilities converged to levels producing more likely outcomes.

### **Refined Instrumentation**

Having determined the efficacy of the revised test design, data collection instruments changed accordingly. Data collection fields changed to ensure they mapped to the synthetic output. This increases the likelihood of collecting data that could create a dataset like the one resulting from the simulation.

### **Utility of Data**

A byproduct of simulating field data in a pilot study is that the analysis process is testable. The team extracted, transformed, and loaded simulation data into the PRoM analytic suite and treated it like real field data. This accelerated the creation of SQL queries and scripts suitable for PRoM upload. These SQL queries are now ready for the main test. PRoM libraries helped animate, summarize, and analyze the synthetic data. This provided a good proof of concept for the analytic process designed for the actual test.

### **Extensibility to Other Tests and Process Modeling**

Insights learned from this project are extensible to other types of analyses - particularly in tests involving process flows or in process mining where there is no database containing preexisting event log data.

### **Testing**

This project showed that simulating field data could serve as a pilot study to refine testing assumptions and to determine data collection requirements. For example, in some test environments multiple tests execute to accomplish these goals. In this project, the time constraints and the lack of opportunity for retesting will require the test to work the one and only time it executes. Simulating a pilot study is applicable to other tests without resources for pilot studies. Additionally, one may argue that simulating a pilot study prior to an actual pilot study has even greater applicability.

### **Process Mining**

Aalst (2011) presents the activity of process mining event logs from database sources as an optimal method of developing process models. In their absence, people knowledgeable about the process can create process models based on how they believe processes execute. However, these are best guesses of process flow. There is often a great discrepancy between expectations and reality. This project demonstrates it is possible to build better process models without the benefit of event logs. A team can brainstorm possibilities, simulate them, and then run them in the PRoM tool - refining the models iteratively. The challenge is gathering the proper domain expertise. Fortunately, this expertise is often available to teams. Gathering those insights with the benefit of iterations within a simulation leverages the knowledge to refine project assumptions early in a project's lifecycle.

## CONCLUSIONS

The team's pilot effort provided an excellent opportunity to assess feasibility of the larger scale study, without the benefit of operator participation. This pilot study succeeded with well-defined feasibility objectives, a focused path for analysis, and a plan to determine success based on the objectives. Given the potential pitfalls of applying inappropriate methodologies to a full-scale study (i.e., logistics, resources, time, risk, and schedule), the pilot study afforded a tremendous opportunity to identify methods suitable for the main test and increase the likelihood of its success. Analyzing the simulated pilot study data against investigative questions refined the data collection strategy in ways unforeseen before simulating the process. Simulated mapping of the information process flow proved accurate and the data collectors were able to successfully practice with the data collection instruments on a small given sample. By choosing three focused questions for investigation, engaging domain experts early, using an incremental development style, and iterating between domain engagement and incremental development, the team learned that domain experience was the best way to validate the simulated pilot and obtain the data.

## REFERENCES

- Aalst, W. M. P. van der. (2011). *Process mining: Discovery, conformance and enhancement of business processes*. Berlin: Springer.
- Baker, T. L. (1994). *Doing social research* (2nd ed.). New York: McGraw-Hill Inc.
- Barhak, J. (2015). *Object oriented population generation*. Retrieved from [http://www.modsimworld.org/papers/2015/Object\\_Oriented\\_Population\\_Generation.pdf](http://www.modsimworld.org/papers/2015/Object_Oriented_Population_Generation.pdf).
- Box, G. E. P. (1979). "Robustness in the strategy of scientific model building", in Launer, R. L.; Wilkinson, G. N., *Robustness in Statistics*, Academic Press, pp. 201–236.
- Dalbey, J. (n.d.). *Pseudocode standard*. Retrieved from [http://users.csc.calpoly.edu/~jdalbey/SWE/pdl\\_std.html](http://users.csc.calpoly.edu/~jdalbey/SWE/pdl_std.html).
- Gendron, G. R., (2016). Probability process footprint generator [Git repository]. Available from <https://github.com/jgendron/ProbabilityProcessFootprint>.
- Kossiakoff, A., Sweet, W. N., Seymour, S., & Biemer, S. M. (2011). *Systems engineering principles and practice*. Hoboken, NJ: John Wiley & Sons.
- Lancaster, G. A., Dodd, S., & Williamson, P. R. (2004). Design and analysis of pilot studies: recommendations for good practice. *Journal of Evaluation in Clinical Practice*, 10(2), 307-312. Retrieved from <http://www.ncbi.nlm.nih.gov/pubmed/15189396>.
- Process. (n.d.). In *Oxford Dictionaries online*. Retrieved from <http://www.oxforddictionaries.com/definition/english/process>.
- Process Mining Group. (2015). PRoM 6.5.1: Process Mining [Software]. Available from <http://www.processmining.org/prom/start>.
- Python Software Foundation. (2015). Python 3.4.3 [Software]. Available from <https://www.python.org/downloads/>.
- Python Software Foundation. (2016, January 8). 9.6. *random* — Generate pseudo-random numbers. Retrieved from <https://docs.python.org/2/library/random.html>.
- Rabiner, L. R. (1989, February). A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2), 257-286. Retrieved from <http://www.robots.ox.ac.uk:5000/~vgg/rg/papers/hmm.pdf>.
- Thabane, L., Ma, J., Chu, R., Cheng, J., Ismaila, A., Rios, L. P., ... Goldsmith, C. H. (2010). A tutorial on pilot studies: the what, why and how. *BMC Medical Research Methodology*, 10(1). Retrieved from <http://www.biomedcentral.com/1471-2288/10/1>.