

# **A Decision Tool that Combines Discrete Event Software Process Models with System Dynamics Pieces for Software Development Cost Estimation and Analysis**

**Dr. Carolyn Barrett Mizell, NASA**

**Dr. Linda Malone, University of Central Florida**

---

## **Abstract**

*The development process for a large software development project is very complex and dependent on many variables that are dynamic and interrelated. Factors such as size, productivity and defect injection rates will have substantial impact on the project in terms of cost and schedule. These factors can be affected by the intricacies of the process itself as well as human behavior because the process is very labor intensive. The complex nature of the development process can be investigated with software development process models that utilize discrete event simulation to analyze the effects of process changes. The organizational environment and its effects on the workforce can be analyzed with system dynamics that utilizes continuous simulation. Each has unique strengths and the benefits of both types can be exploited by combining a system dynamics model and a discrete event process model. This paper will demonstrate how the two types of models can be combined to investigate the impacts of human resource interactions on productivity and ultimately on cost and schedule.*

---

## **Keywords**

*Software Development, Project Management, System Dynamics  
Discrete Event Software Development Process Models*

---

Process simulation models, system dynamics models, and static cost models already exist for software development projects. Each of these tools has advantages and disadvantages and the appropriateness of each depends on the application. System dynamics models are useful tools for demonstrating the dynamic behavior of a project and are based on project variables and tasks as a whole with no process details or intricacies being captured. Process models, on the other hand, do provide great detail on the process and can be used to provide guidance on the sequence of process steps and information flows and can also be used to analyze proposed process changes. In addition, process models can support management planning and control activities. This type of modeling, however, does not capture the interactions and structural relationships as effectively as system dynamics modeling. Therefore, it is desirable to combine information from both types of models in order to more thoroughly analyze a project. The combination of continuous and discrete models is present in the literature for only a couple of cases. Two examples of combined models can be found in the work of Martin and Raffo (Martin and Raffo 2000; Martin and Raffo 2001; Martin 2002) and Donzelli and Iazeolla (Donzelli and Iazeolla 2001). Martin added continuously changing sections based on Abdel-Hamid's model to a discrete event process model and affected the discrete event clock to run continuously. Donzelli and Iazeolla This work will consist of a separate system dynamics software and discrete event process model in order to maximize the benefits gained from each. Users will be able to understand and

experiment with the system dynamics model separately and the data of interest will be sent to the discrete event process model to affect it.

For this work, system dynamics will be used to analyze human resource issues such as experience levels and turnover. This information will be combined with a discrete event model of a waterfall lifecycle process model. The human resource area was selected because of its potential impact on a project, especially early on when managers may perceive staffing issues too optimistically or not at all. A simplified system dynamics model is used to make it easy to understand and to capture the key variables of interest. Existing hybrid models such as Martin's are very powerful, but also very complex. Previous work has shown that smaller and less complicated models are better for presentation to those that are not familiar with process models and simulation (Madachy and Tarbet 2000). The goal of this work is to communicate to decision makers the potential impact of turnover and experience levels on the cost and schedule of a project.

### **Discrete Event Process Simulation Model**

The Process Analysis Tradeoff Tool, PATT ©, is a discrete event process simulation model that was developed for NASA to assess the benefits of Independent Verification and Validation (IV&V) on the IEEE 12207 software development process (Raffo and Wakeland 2003). The tool is intended to enable adaptation to multiple projects and IV&V techniques. The model uses industry average data for input variables such as product size, productivity (LOC/Hr), and defects (per KSLOC). The user provides % of overall effort that should be allocated to each process step as well as the number of desired staff for each step. The model outputs the size, effort, rework effort,

entire process duration, average duration, number of injected defects, detected defects, and corrected defects.

The use of probability distributions for key variables such as size, productivity, and defects is a truer model of reality, especially in the early stages of a project. The model's outcomes will be driven by random variables drawn from probability distributions. Numerous runs of the process with different random numbers will provide more meaningful information.

Productivity rates are highly variable and algorithmic cost estimation models such as COCOMO do not model the factors affecting productivity very well (Kemerer 1987). Major variations from constant productivity can occur, especially with system programming products that use hundreds of thousands of lines of code, built by multiple teams and several layers of management (Putnam and Myers 1992).

It is important to communicate to decision makers how delays such as the time it takes for inexperienced staff to become as productive as experienced staff can affect a project. These types of delays can have major impact and yet are not always formally considered. Research has shown that schedule overrun problems can be attributed to the interaction of manpower-acquisition policy and turnover in addition to software estimation accuracy (Abdel-Hamid and Madnick 1991). Even when managers are aware of such delays, studies have shown that it is difficult to deal with the delays without tools that help to develop an adequate mental model of the dynamics of the system (Sengupta and Abdel-Hamid 1999).

Productivity is very dependent upon the skill and availability of the workforce for a project. Experienced staff will be more productive than inexperienced staff, and it is

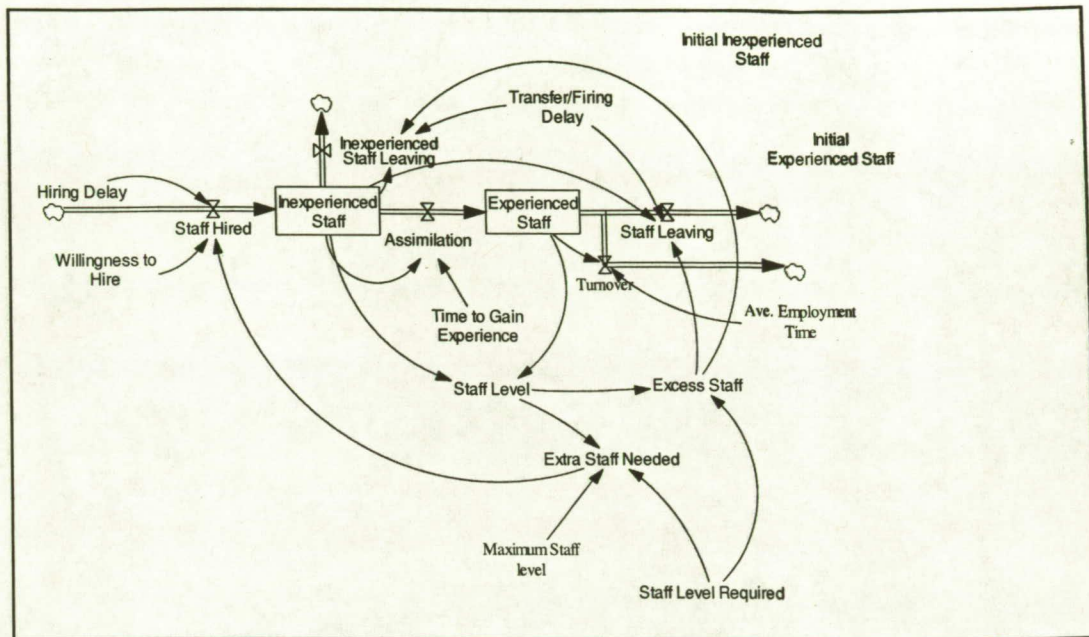
unrealistic to expect that all the staff on a particular project will be experienced on day one. Inexperienced staff must be trained and assimilated into the project environment and this takes time. Turnover of employees is an issue that will affect staffing and ultimately, productivity. Turnover rates as high as 34% are often seen on software development projects (Abdel-Hamid and Madnick 1991). As experienced employees leave a project, qualified new employees must be found and hired or transferred to the project. These inexperienced workers must now be trained and assimilated before their productivity levels can match those of experienced project personnel.

The interrelationships between such staff-related variables are best captured with system dynamics. The human resource sub-system of the Abdel-Hamid and Madnick system dynamics model of a software development project was used as a guide for adding continuously changing staff levels to this model (Abdel-Hamid and Madnick 1991) and so was the combined model developed by Martin (Martin 2002). Martin developed a combined model by integrating the entire Abdel Hamid and Madnick system dynamics model of the software development environment with a discrete event model of the standard ISPW-6 software process. This work showed the benefits of combining continuous and discrete event models.

### **Addition of Continuous Simulation to Model**

In the system dynamics model, resources are divided into experienced and inexperienced groups and the time-changing levels will be derived based on a turnover rate and assimilation rate. The ratio of experienced staff to total staff will be calculated and then sent to the discrete event model to affect productivity.

The System Dynamics model was created in Vensim software. Vensim models graphically display the connections and feedback loops of the system. It is possible to instantly see simulation results for all variables on the screen to view more detailed results of any selected variable of interest with different analysis tools.



Human Resource Model in Vensim

This simplified version of the Abdel-Hamid and Madnick Human Resource model (Abdel-Hamid and Madnick 1991) assumes that the organization is willing to hire and that there is no delay in hiring.

The literature suggests an average productivity of 3.5 LOC/Hr (SEL 1993), but as previously mentioned, there are many dynamic and interacting factors that can cause this value to vary. It is especially difficult to accurately estimate the productivity of a project's staff before a project begins because the availability and skill of the workforce is not known. If a certain staff level and experience level is assumed, this can change

throughout the course of a project due to turnover and hiring practices and it is crucial to consider the potential impacts of these on productivity and ultimately, cost and schedule.

The desired output from the system dynamics model is the number of experienced personnel available throughout the project. This number will be used to develop a ratio multiplier to productivity and will be used in the discrete event software process model.

It is assumed that an inexperienced person is 50% as productive as an experienced

person. Since a ratio of 1 for  $\left(\frac{\#experiencedstaff}{\#totalstaff}\right)$  would equate to a

productivity multiplier of 1 and a ratio of 0 for this quantity would equate to a multiplier of 0.5, the following equation will be used to calculate the productivity multiplier:

$$productivitymultiplier \equiv 0.5\left(\frac{\#experiencedstaff}{\#totalstaff}\right) + 0.5$$

The system dynamics model will be used to calculate the productivity multiplier for each day during the project and this data will be read into the discrete event model.

Each time the discrete event model attempts to draw a productivity number from the productivity distribution, the time will be captured and the associated productivity multiplier will be used in the calculation to affect the productivity draw.

The organization's environment must be taken into consideration when selecting values for the amount of time it takes for an inexperienced person to become experienced (assimilation delay), the hiring delay, and the quit/transfer rate. A large NASA software development project will be used as an example of how to combine data from the models. Considering the NASA development environment, an assimilation rate of 6 months, a quit/transfer rate of 2 weeks, and a zero hiring delay will be used in the model.

## Experimentation

Data from a real NASA project will be used for this experimentation. The first cost and schedule estimate was developed using previous projects for an estimate by analogy. No formal requirements existed and the architecture had not been selected at this very early point. In order to capture and account for the large amount of uncertainty that existed at this point in time, probability distributions will be used for size, productivity, and defect insertion rates. The productivity and defect distributions will be based on data from the Software Engineering Laboratory (SEL). This organization collected software development data from the Goddard Space Flight Center Flight Dynamics organization for over 25 years (Basili, McGarry et al. 2002). The GSFC software development organization was responsible for the development of mission software and used NASA personnel as well as contractors. Probability distributions for defect injection rates are also based on SEL defect data. Since a large amount of the SEL's data was readily available (CeBASE 2005) and since we are using a NASA project for analysis, the environments are similar and it is reasonable to use probability distributions based on this data.

The project's estimate by analogy was based on a size of 1.4 Million LOC and the literature says that this size can be underestimated by a factor of 4 at this early point (Boehm, Abts et al. 2000). The following are the input parameters that will be used for the experiment:



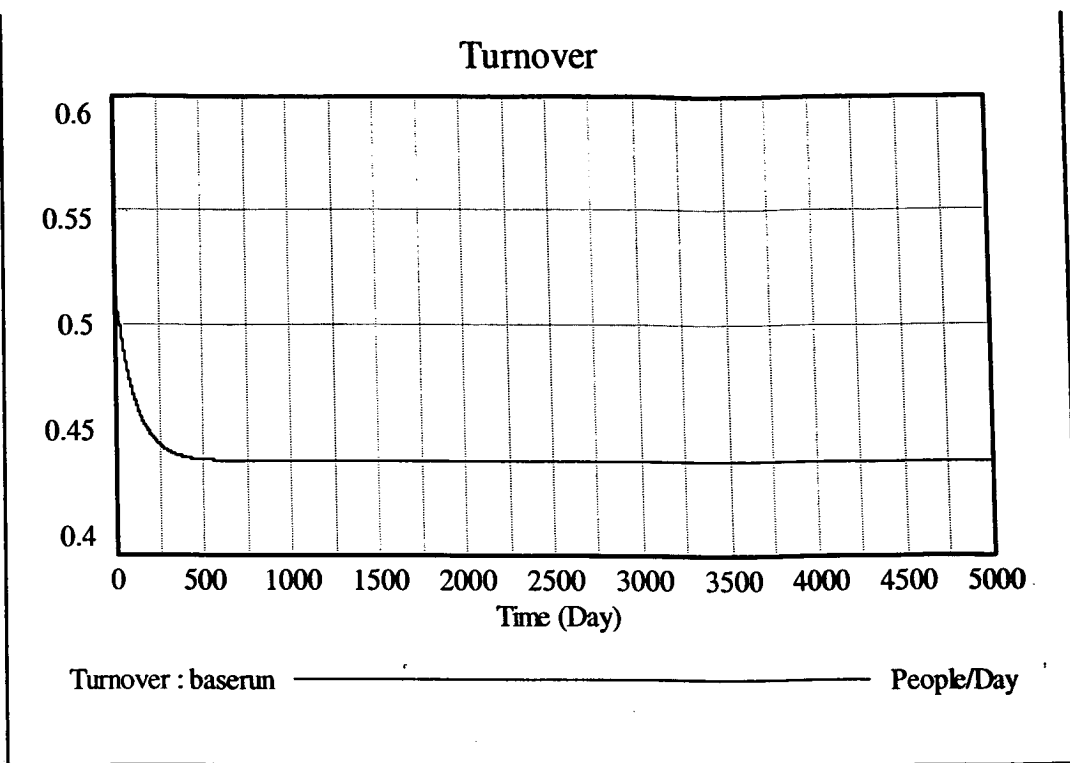
*Size: Uniform (1400, 5600) KSLOC*  
*Productivity: Erlang (1.36, 3) LOC/Hr*  
*Requirement Defect Injection: Lognormal (2.6, 7.1) errors/KSLOC*  
*Design Defect Injection: Lognormal (17.1, 73.3) errors/KSLOC*  
*Code Defect Injection: Weibull (28.4, 0.8) errors/KSLOC*  
*Test Defect Injection: Exponential (40.9, 0) errors/KSLOC*  
*Bad Fix Defect Injection: 30 errors/KSLOC*  
*Documentation Defect Injection: 30 errors/KSLOC*

### Input Parameters

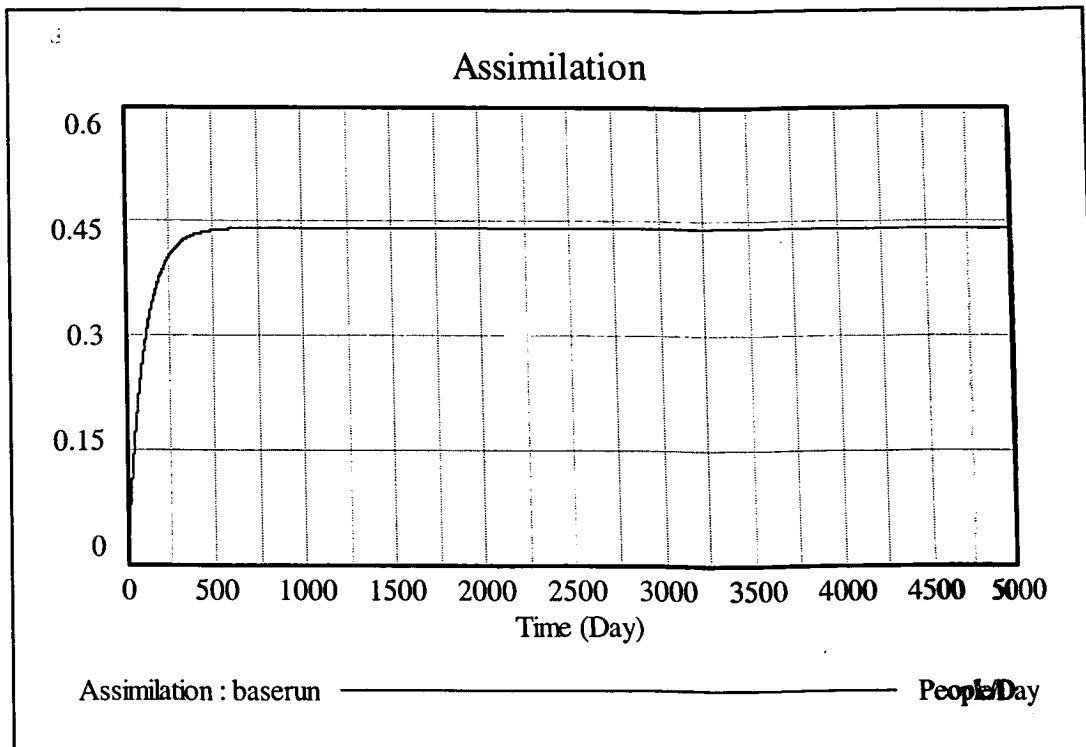
The first run will use the discrete event process model only and will not provide any productivity adjustments based on staffing experience levels. Stated another way, the entire staff is experienced from day one and remains on the project for the entire time. The mean effort for this scenario is 16,705 person-months and the mean duration is 68 months.

In order to test the other extreme situation, it is assumed that the staff of 350 will always be inexperienced and therefore produce at a rate that is 50% of the probability distribution draws. The mean effort for this case is 34,002 person-months and the mean duration is 143.7 months. It is easily seen from this, that the mean effort and duration are more than doubled for this case.

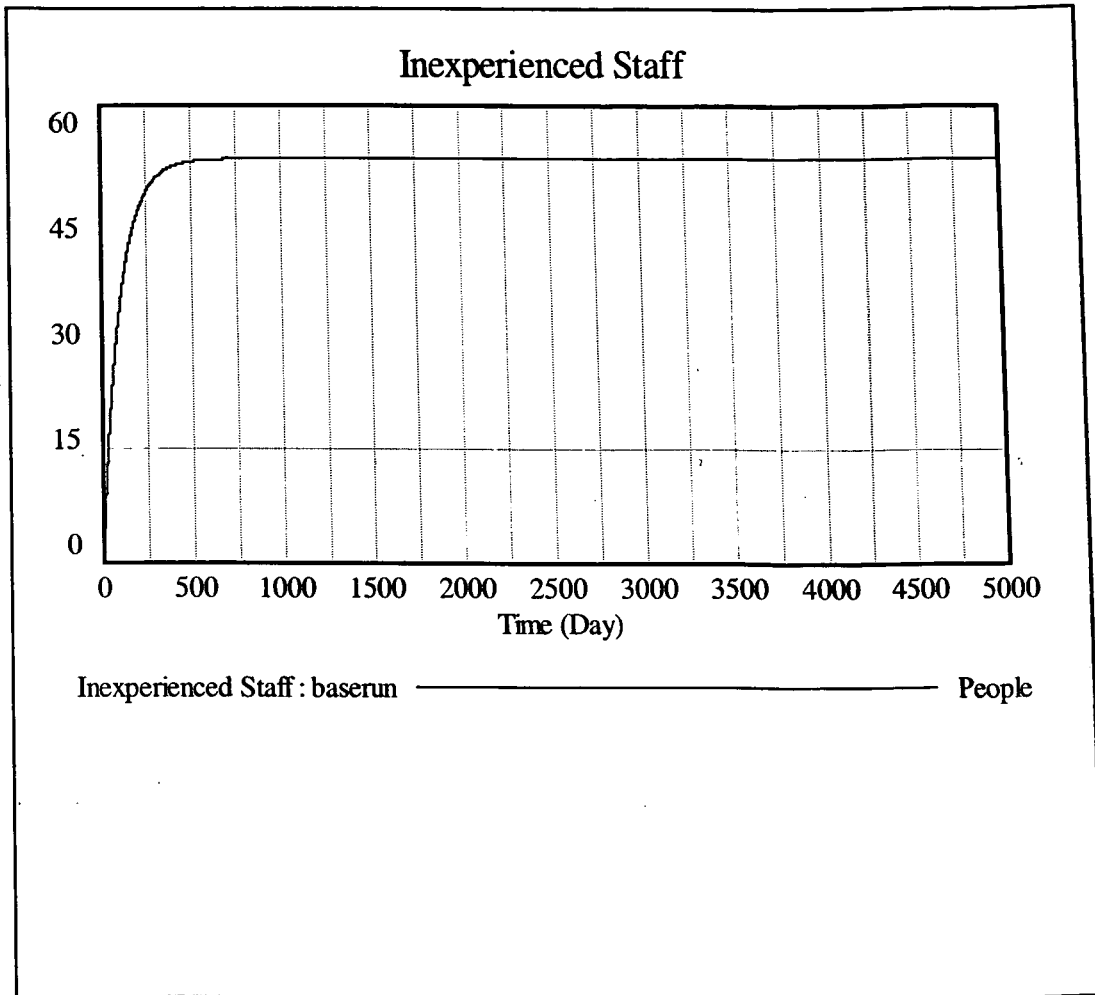
Next, the system dynamics model is used to consider the effects of turnover. Turnover is set at 30% and the assimilation delay is set at 6 months. The entire staff of 350 is considered experienced on day one of the project. The following figures show the output for turnover, assimilation, and staffing levels.



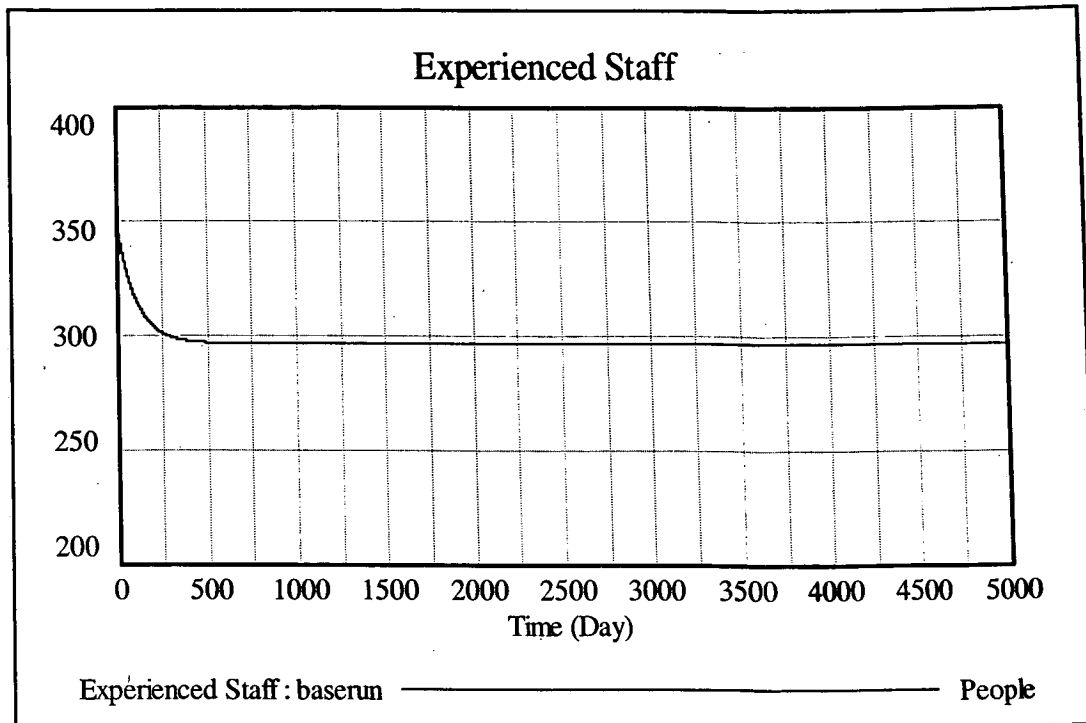
Turnover vs. Time (Baserun)



Assimilation Delay vs. Time (Baserun)



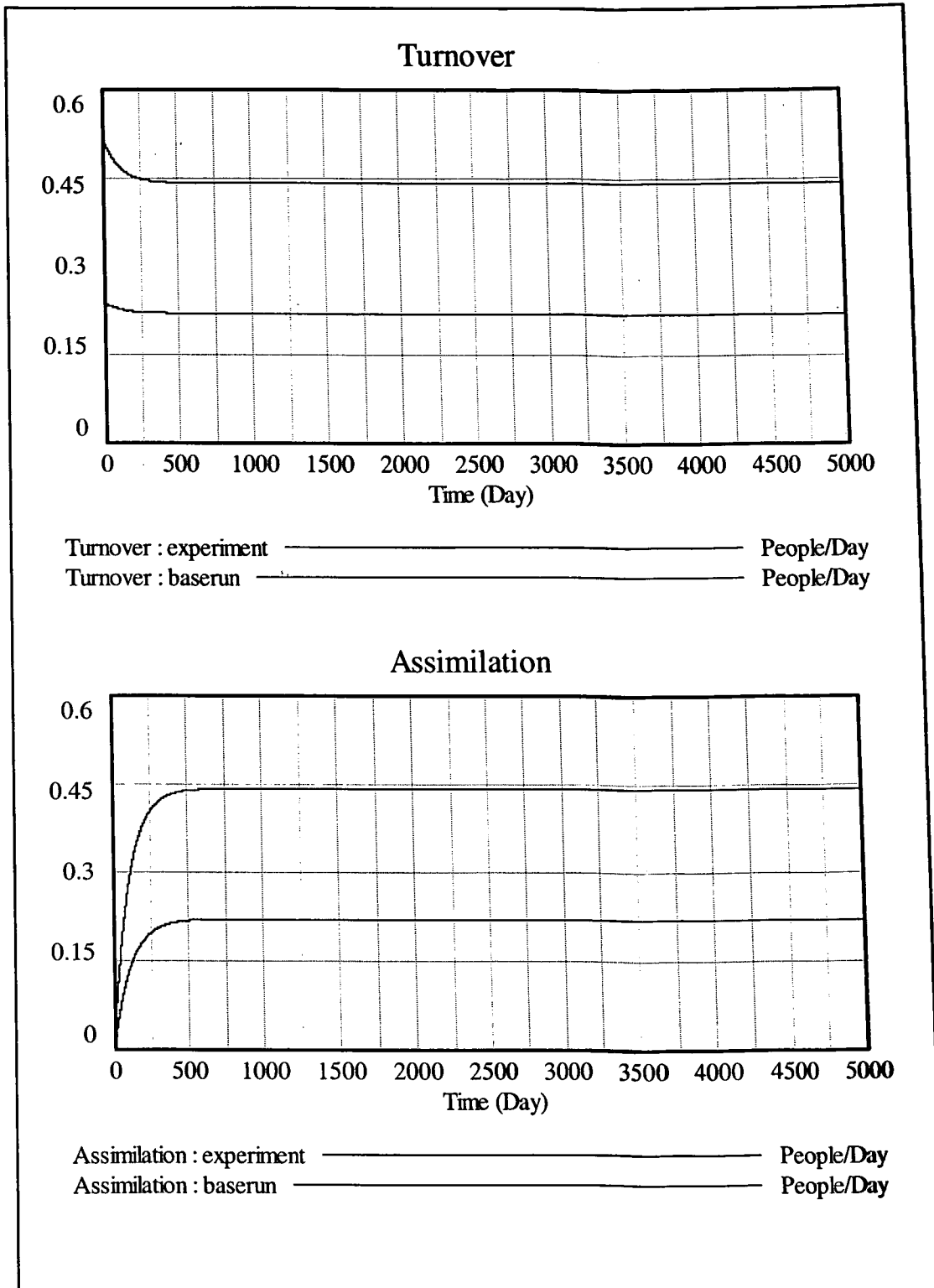
Number of Inexperienced Personnel vs. Time (Baserun)



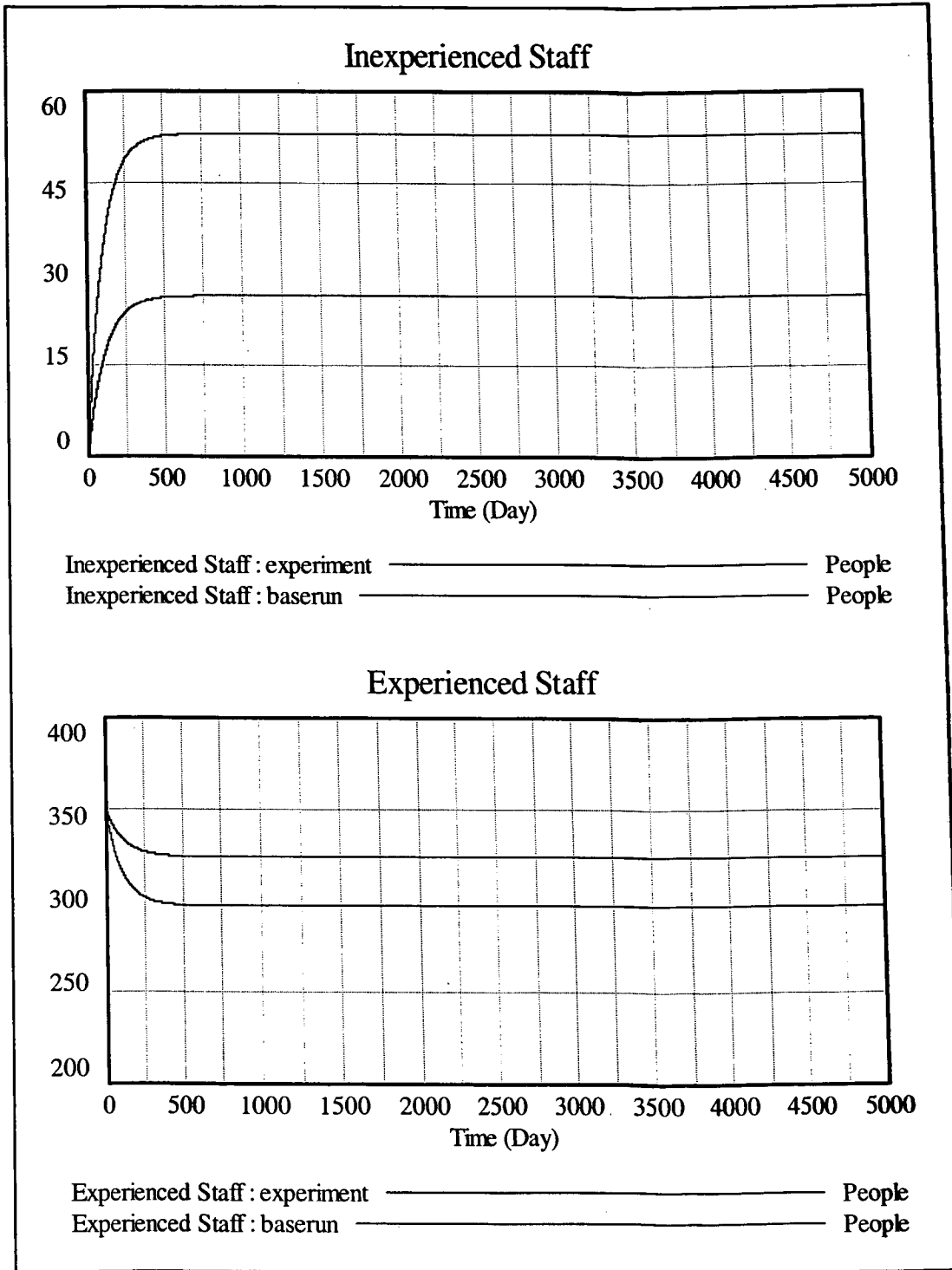
Number Experienced Personnel vs. Time (Baserun)

The productivity ratio is calculated and sent to the process model. The mean effort changes to 17,776 person-months and the mean duration equals 76 months.

A nice feature of Vensim is that the variables of interest can easily be changed and the model can be run with the effect of the changes displayed on top of the baseline run. If the turnover is lowered to 15%, the following results are obtained:



Turnover and Assimilation vs. Time (Experiment, Baserun)



Number of Inexperienced/Experienced Staff vs. Time (Experiment, Baserun)

Running the process model with this data leads to a **mean effort of 17,086 person-months** and a **mean duration of 70.8 months**.

The following table provides a summary of other scenarios and the effect on effort and duration:

Turnover	Starting # Experienced Staff	Starting # Inexperienced Staff	Effort (person- months)	Duration (months)
0	350	0	16705	68
15%	350	0	17086	70.8
30%	350	0	17776	76
15%	175	175	18098	89.2
30%	175	175	19839	89.5
15%	0	350	18497	83.8
30%	0	350	18949	86.9

Summary of Effects of Turnover Rates

### Conclusion

Many different factors will affect the ultimate cost and schedule for a project. This work has shown the potential impact of the human resource issues of turnover and the experience level of the staff. The system dynamics model presents an easy to understand graphical representation of the interrelationships of key human resource factors that affect the experience level of staff. The discrete event process

model utilizes this data to affect the productivity which in turn will affect the ultimate cost and schedule. A simplified system dynamics model is used for the purpose of demonstrating how the two types of simulation models can be used in conjunction to consider an important issue for large software development projects such as the impact of turnover on cost and schedule estimates.