# DEFINING THE TO-BE STATE OF AN ENTERPRISE – OVERVIEW

## BACKGROUND

The "conventional wisdom" that Requirements define the "what" in terms of traceable statements and Design the "how" has major flaws. The terms are subject to different interpretations and we need a more practical delineation between Requirements (customer's responsibility) and solution Design (Exigen responsibility).

The traditional approaches to Requirements definition focus on requirements for software systems or embedded military systems and don't adequately cover socio-technological systems where people and technology components perform in an integrated manner to produce the desired results.

This methodology describes (a) What constitutes a Requirements Baseline for a socio-technological system, (b) What is the role of GRADE TO-BE Models in a Requirements Baseline (Fig. 1) and (c) How to systematically produce TO-BE models, using AS-IS models as the basis.

## TERMINOLOGY

The following terms, plus the definitions in the AS-IS Methodology chapter (Fig. 2) are key to understanding the TO-BE definition methodology:

**Requirements** – Everything the customer prescribes that restricts Exigen's degree of freedom in providing a solution. This includes the business processes to be automated, and may include screen form definitions or constraints, such as, a certain Legacy system or department must remain is the AS-IS state, etc.

**Design** – Everything that is up to Exigen to decide as to how to fulfill the requirements.

**TO-BE Model** – A graphical and tabular representation of the desired state of an enterprise in terms of its static structure, dynamic behavior, information flow & interfaces. The Model serves as Transformation Blueprint.

**Transformation Blueprint** – A TO-BE Model, which defines a particular state of a Socio-technological system targeted for implementation and deployment. There is a Transformation Blueprint for the End-State and for each Intermediate Stage.

**Process Essence** (or **Essence Processes)** – The end-to-end business transactions, triggered by Transfer Objects and producing Outputs, as if there was only one Performer involved, that is, free of Tasks related to handovers between potentially changing departments and IT systems.

**Process Incarnation** – The Process Essence assigned or mapped to the Performers identified in the ORG Diagram, depicting the flow of work and handovers between different organizations and technology components.

**End-State** – The state of the enterprise in terms of organization, processes, technology, etc., being envisioned and planned as the end goal.

**Intermediate Stage** – A particular juncture in the enterprise transformation process, corresponding to a deployed State of the organization, technology and processes. For each Intermediate Stage there is a Project, which follows the "Gate" Process.

**Requirements Baseline** – A complete specification, related to each Stage to be deployed, that defines the Requirements, in a verifiable manner, to be fulfilled by Exigen and by the customer. Changes to a Requirements Baseline follow a standard Change Control procedure.

## WHAT CONSTITUTES BUSINESS REQUIREMENTS?

Business Requirements define the TO-BE behavior of Organizational & Technological components, in response to Triggers, and their interaction with each other.

Behavior of a Performer may be considered from the Essence or Incarnation perspective. This methodology calls for 'extracting' the Essence from the AS-IS Incarnation and first defining improvements in the Essence before re-incarnating a Process into a new Performer (ORG) structure.

In theory, Essence would be expressed in terms of business rules by which a Performer transforms inputs to outputs. In most practical cases, however, Essence is expressed as a sequence of steps that it takes to produce outputs in response to inputs (Triggers).

In order to keep process complexity manageable, processes may be segregated according to the following three types:

- **Primary Process** – Externally triggered end-to-end business transaction, such as, processing a loan application (Fig. 3)
- **Secondary Process** – Internally triggered end-to-end business transaction, such as, producing a monthly report (Fig. 4)
- **Supporting Process** – Service process for Primary and Secondary Processes, such as, distributing mail (Fig. 5)

## DEFINING THE REQUIRED END-STATE AND INTERMEDIATE STAGES

The transformation of an Enterprise to a Utility requires a model that defines the End-State and one for each Intermediate Stage to be deployed in the transition from the AS-IS to the End-State.

Each Model serves as Transformation Blueprint for implementing the changes in organization, technology, infrastructure, and processes, corresponding to a deployment Stage.

The objective of the models is to know where you want to get before embarking on the journey. This elementary "wisdom", practiced in most human endeavors, except in traditional BPR and software projects, is a cornerstone of this methodology.

The methodology defines how to develop a Transformation Blueprint, but not the transformation process from AS-IS to TO-BE. This is a process in a different domain and should be defined for each Intermediate Stage, utilizing an MS Project or GRADE Model Template.

## OVERVIEW - THE END-STATE MODEL

The End-State Model defines the TO-BE enterprise in terms of the following major components:

- Organization (human resources)
- Technology (software & hardware components)
- Processes (flow of work and business rules)
- Transfer Objects (documents & messages)
- Interfaces (format of documents & messages)

The major components are inter-linked in a unified GRADE Model framework, so that all relevant information is at the fingertips, such as, who performs a task, what are the inputs to it, what outputs it produces by what rules, what is the task sequence, how long the task takes, what resources it requires, etc.

## PRIMARY STEPS IN DEFINING THE END-STATE MODEL

This methodology is aimed at streamlining business processes for improved service levels and customer experience, and to reduce human resource requirements. It assumes that an AS-IS model has been built. If FTE reduction is the sole objective, a simplified procedure may be applicable.

Several of these steps can be performed concurrently and several iterations may be required before the Model qualifies as the Transformation Blueprint.

**Step 1:** Finalize which Organizational and Technology Performers shall remain in the AS-IS state unchanged, by color-coding those Performers in the AS-IS ORG Diagram.

**Step 2:** Identify which Tasks in the AS-IS Primary, Secondary and Support Processes could be omitted, changed or done in parallel, if the Incarnation-specific constraints were removed, and revise the Processes accordingly. The results of this activity are the TO-BE Essence Processes.

**Step 3:** Evaluate each Task of the TO-BE Essence Processes for where Exigen technology could be applied. Revise the AS-IS ORG Diagram by adding the Exigen technology components (Fig. 6)**.** Reference these as Performers in the Process Diagram Tasks (Fig. 7).

**Step 4:** Revise the ORG Diagram further by removing the IT components to be replaced by Exigen technology components. The result after this step reflects the IT configuration of the End-State.

**Step 5:** Revise the AS-IS External Transfer Objects Table, to reflect any changes due to introduction of the Exigen components and removal of any existing IT components. Specifically, delete superfluous Transfer Objects, add any new ones, and change Senders, Receivers and Transfer Mode, to reflect all External interaction changes due to technology reconfiguration.

**Step 6:** Revise the ORG Diagram further, to reflect the planned End-State organization, as conceived by organization change management considerations. In an “ideal world” this would be derived from analyzing the TO-BE Processes. More practically, organization concepts can be developed in parallel and validated and adjusted after the Processes are defined.

**Step 7**: Revise the External Transfer Objects Table further, to reflect any changes in Organization. Revise the Internal Transfer Objects Table to reflect changes in the Organization and Technology, deleting superfluous Transfer Objects, adding new ones and changing Senders, Receivers and the Transfer Mode.

**Step 8:** Add the Organizational Performers from the revised ORG diagram to the Process Tasks produced in Step 3. If simulation is planned, to validate assumptions about required FTE's, costs, or other aspects, add Performer instances, costs, skills and other resource characteristics (Fig. 6) to the ORG Diagram, and add Task duration, resource consumption and other performance characteristics to the Tasks in the Process Diagrams (Fig. 7).

**Step 9:** Update the Events in the Process Diagrams to correspond to the entries in the revised Transfer Objects Table. Events in Process Diagrams are shown adjacent to the arrows that connect Tasks and correspond to Transfer Objects in the Transfer Object Table.

**Step 10:** Add Service Level requirements and validate the End-State Model, securing sign-off.

Validation can be done with the help of simulation, to determine whether the planned reorganization can handle the processes without bottlenecks. Additionally, 'what if' scenarios may be used to determine optimal resource allocation and validate the organization concept from Step 6, prior to committing to the enterprise transformation.

If the AS-IS Model has been prepared for simulation (Fig. 8), any conceived process changes can be evaluated incrementally to determine what benefits a particular set of changes would bring, thus providing quantified data to determine transformation priorities and deployment staging (Fig. 9, Fig. 10, and Fig. 11)

**Step 11:** Define the scope, on a high level, of the Intermediate deployment Stages in terms of Organization, Exigen Components, Transfer Objects, Processes, Interfaces and Infrastructure. Ideally this would be a subset of the End-State, but in reality it could include measures to support the Intermediate Stages.

## OVERVIEW - THE INTERMEDIATE-STAGE MODEL

There is an Intermediate Stage Model for each Stage to be deployed. The model is derived from the AS-IS and the End-State models. It defines the Organization, Technology, Infrastructure, Processes, Transfer Objects and Interfaces to be implemented and rolled out at a particular Stage.

Interfaces, referenced in the End-State Model for the Intermediate Stage, are specified in detail:

- Between Exigen components and External IT systems
- Between Exigen components and customer's IT systems & networks
- Between users (screen forms & documents) and Exigen components.

The End-State model is kept current, via the Change Control mechanism, as Intermediate Stages are implemented and any required changes to the End-State Requirements Baseline become evident.

The Intermediate Stage model is also kept current, as the implementation progresses, and serves as the AS-IS Model for defining the next deployment Stage.

## PRIMARY STEPS IN DEFINING AN INTERMEDIATE-STAGE MODEL

The following steps are performed for each Intermediate Stage to be deployed. Note that the TO-BE of the first Intermediate Stage is based on the initial AS-IS Model. Each subsequent Intermediate Stage definition is based on the AS-IS Model, which reflects the changes of the previous Stage.

**Step 1:** Revise the AS-IS ORG Diagram to reflect the Performers of the TO-BE Intermediate Stage, color-coding Performers, which shall be added, deleted, changed or remain unchanged.

**Step 2:** Revise the AS-IS Processes to reflect the TO-BE processes of the Intermediate Stage. For the first Intermediate Stage, take the Incarnation Processes from the End-State model and integrate them into the Intermediate Stage Model.

**Step 3:** Map the Performers from the revised ORG diagram to the Tasks of the Incarnation Processes of the Intermediate Stage

**Step 5:** Specify interfaces between Exigen components and any External IT systems involved in the Intermediate Stage.

**Step 4:** Specify interfaces between Exigen components and any Internal IT systems or networks involved in the Intermediate Stage.

**Step 6:** Specify interfaces between Exigen components and users involved in the Intermediate Stage.

**Step 7:** Define the process of transitioning from the AS-IS State to the next Intermediate Stage. Ideally this is a GRADE model, with schedule milestones mapped to MS Project. A GRADE Model template for the transformation process will be included in the "Cookbook", as a "Gate-Driven Project Methodology".

---

**Fig. 1, The Role of GRADE Models in Requirements Baseline**

This methodology is based on the assumption that an AS-IS Model has been developed. The same basic principles may be followed when skipping the AS-IS definition, but it is not recommendable because in reality different people have a different perception of what the AS-IS is, and as a result false assumptions are frequently made.

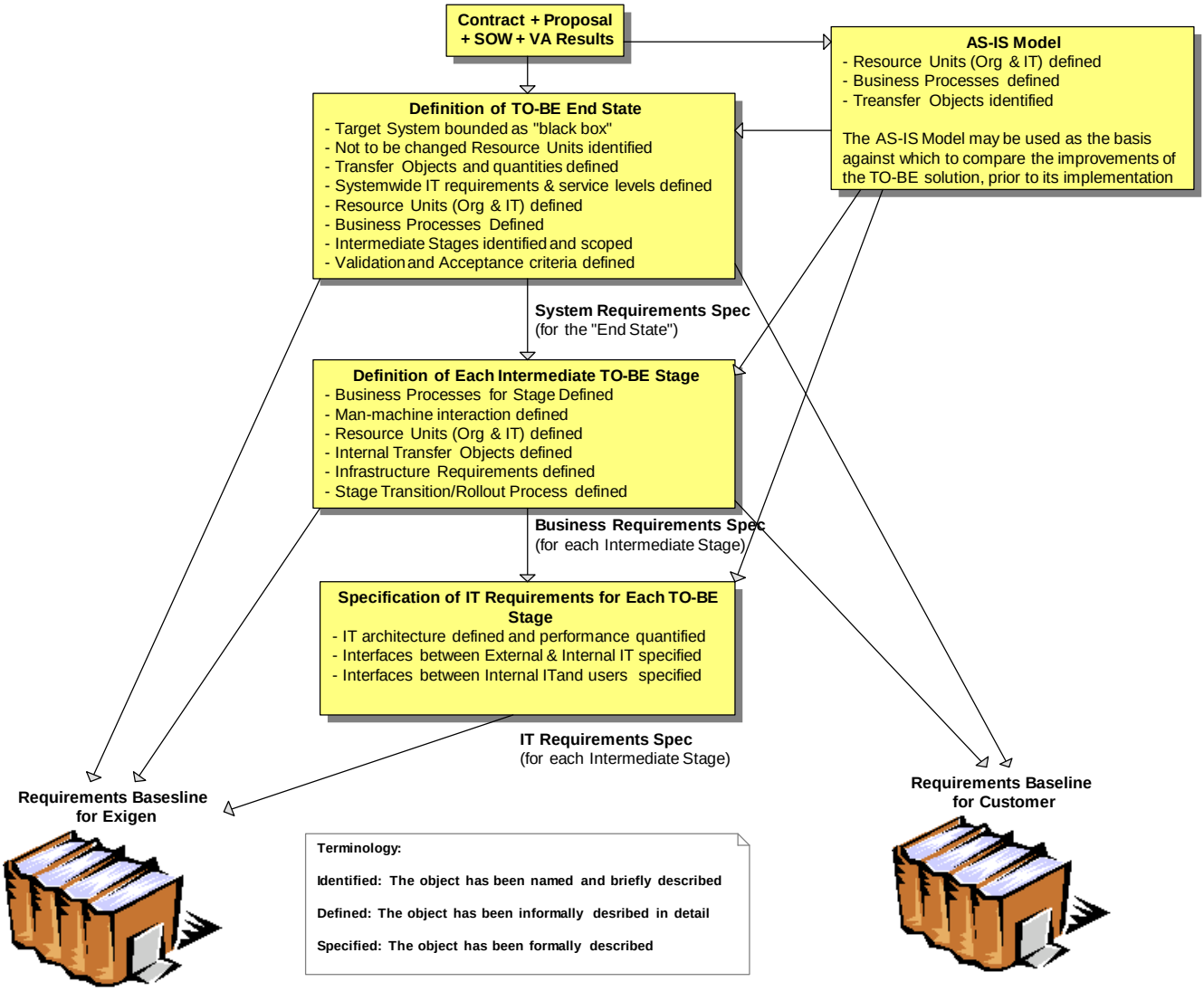


**Fig. 2, Terminology from AS-IS Chapter**

**Active Object** – A physical entity, capable of receiving Transfer Objects, acting upon them and sending them to other Active Objects. The following terms are used to characterize Active Objects:

- **Target System** – Organization & systems to be analyzed, e.g., Mortgage Processing Center.
- **External Entity** – Active Object of the environment that the Target System interacts with, e.g., Loan Request Initiators.
- **Performer** - A Resource Unit, such as a department, an individual or an IT system, that performs a Process, e.g., Mortgage Processing Center, Broker, Legacy System.

**Transfer Object** – Information or a "thing", such as a message or document, which an Active Object sends (Sender) to another Active Object (Receiver) for some processing.

**Trigger** – An Event that triggers a Process, e.g., receipt of a Transfer Object, a certain date, periodicity, or value of a data field or counter. External Triggers are Transfer Objects from External Entities. Internal Triggers are internal to the Target System.

**Process** – A sequence of Tasks initiated by Trigger(s), producing Outputs(s). A Task may be further refined into a Process that consists of Tasks.

**Task** – A step in a Process or a placeholder for a Process, e.g., approving a loan application.

## Fig. 3, Example of a Primary Business Process

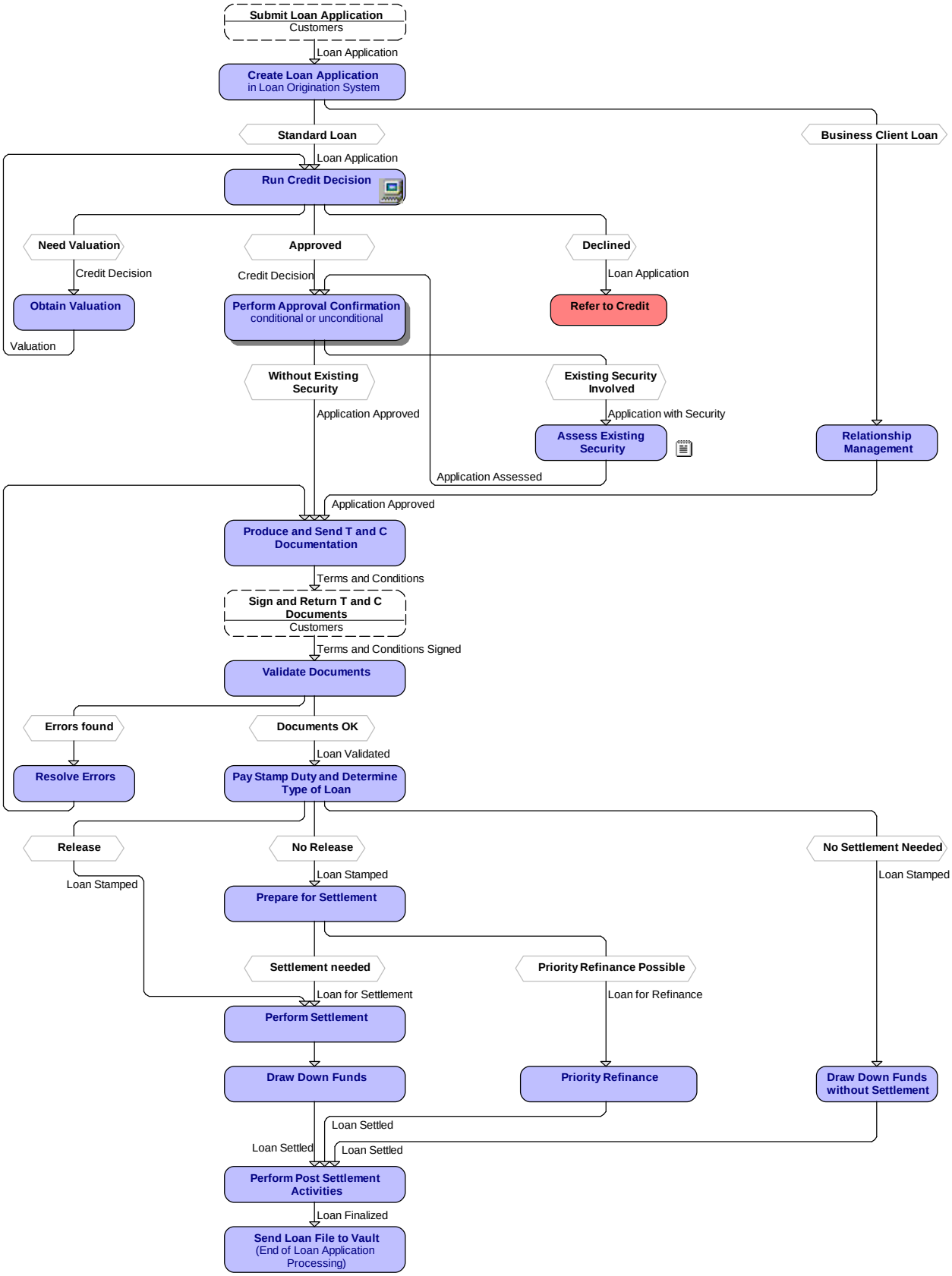
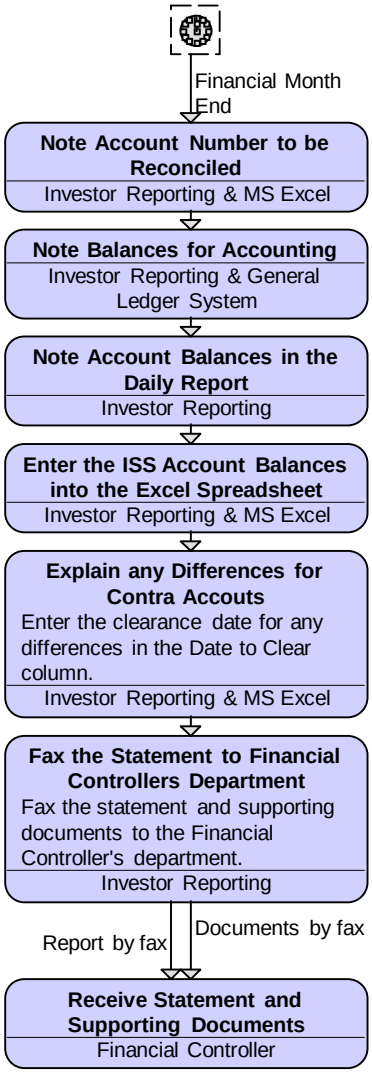


**Fig. 4, Example of a Secondary Business Process**



## Fig. 5, Example of a Servicing Process

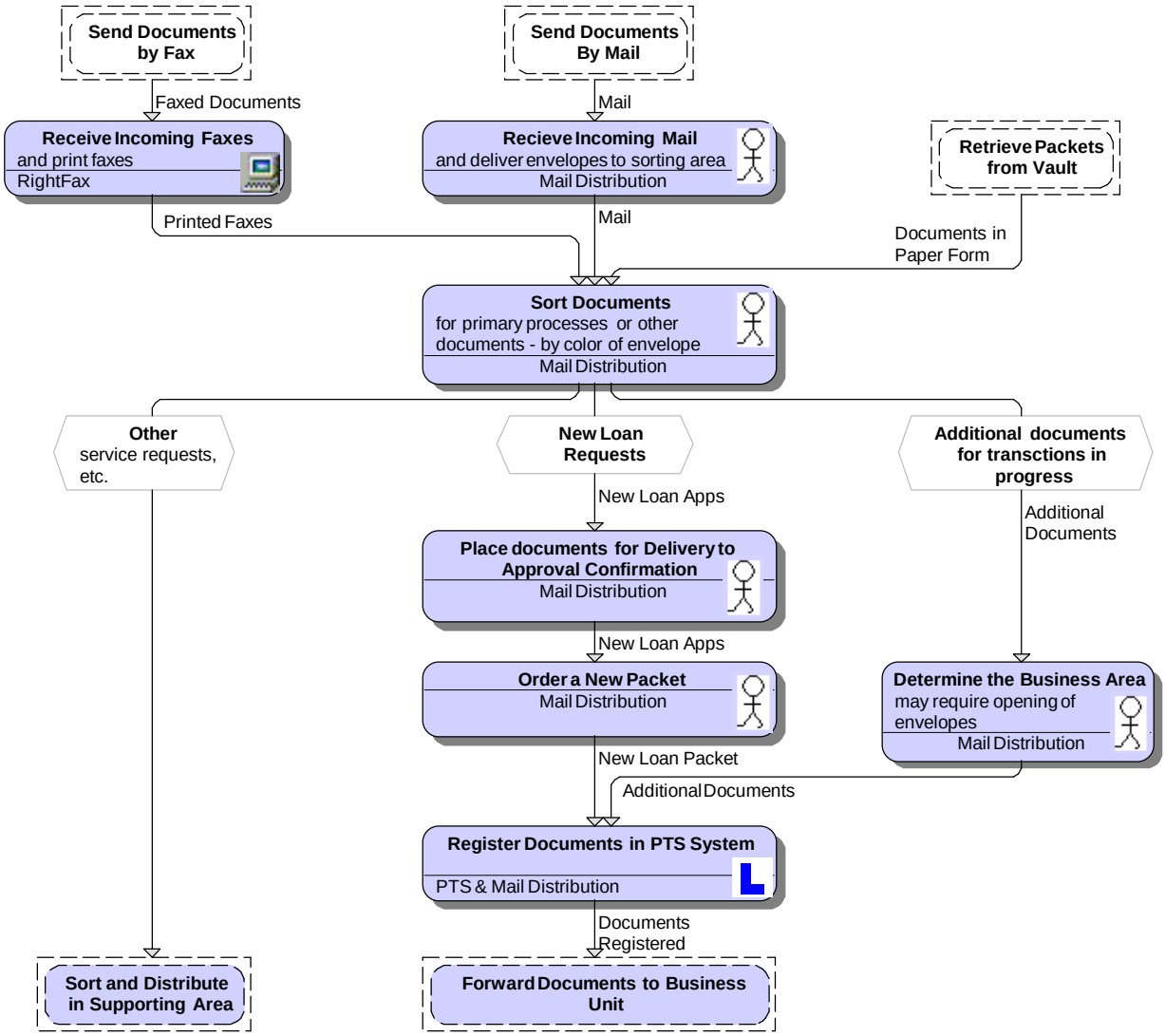
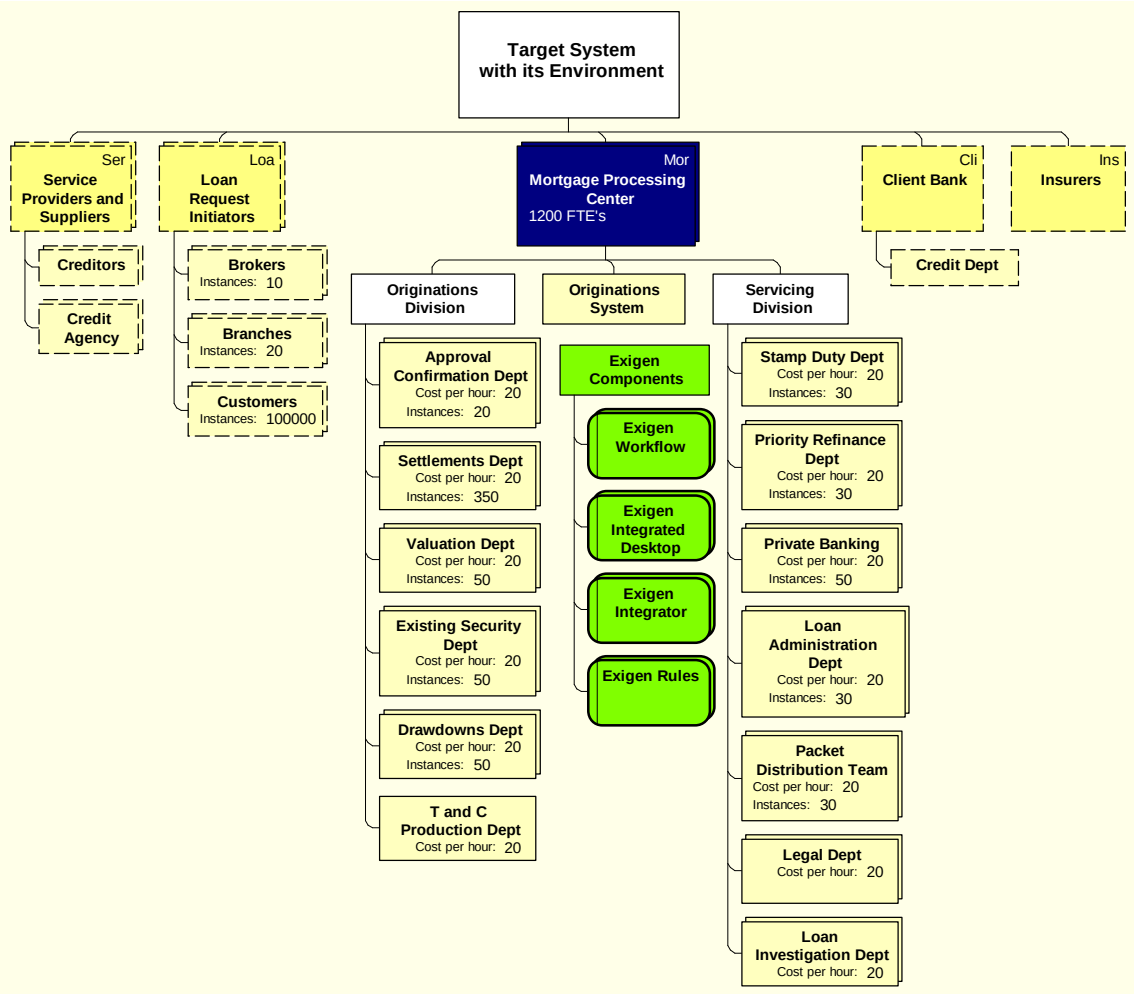


**Fig. 6, Exigen Components Added to the AS-IS ORG Diagram**

The rounded boxes identify the Exigen Components to be uses in the solution.

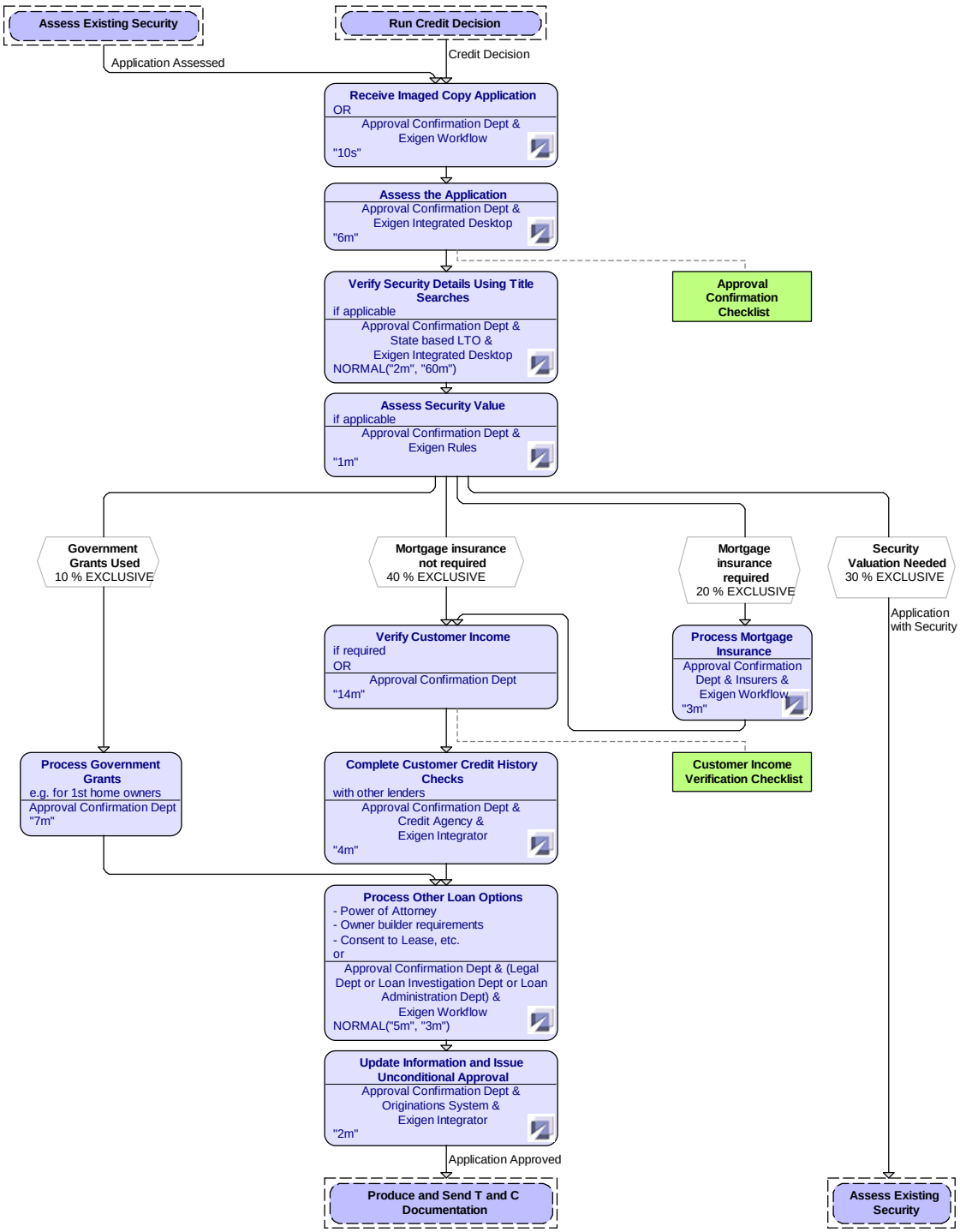



The Simulator takes information about the number of people in the Departments from this diagram and combines it with the Task duration from the Process Diagrams and, based on the total number of Loan Applications processed, comes up with the simulation results shown in the bar charts (Fig. 9 and Fig. 10**)**

The Simulator is driven by a "load generator" (associated with the Task "Submit Loan Application" not shown here), which generates a loan application according to a specified load distribution.

## Fig. 7, TO-BE Version of "Perform Approval Confirmation" Process

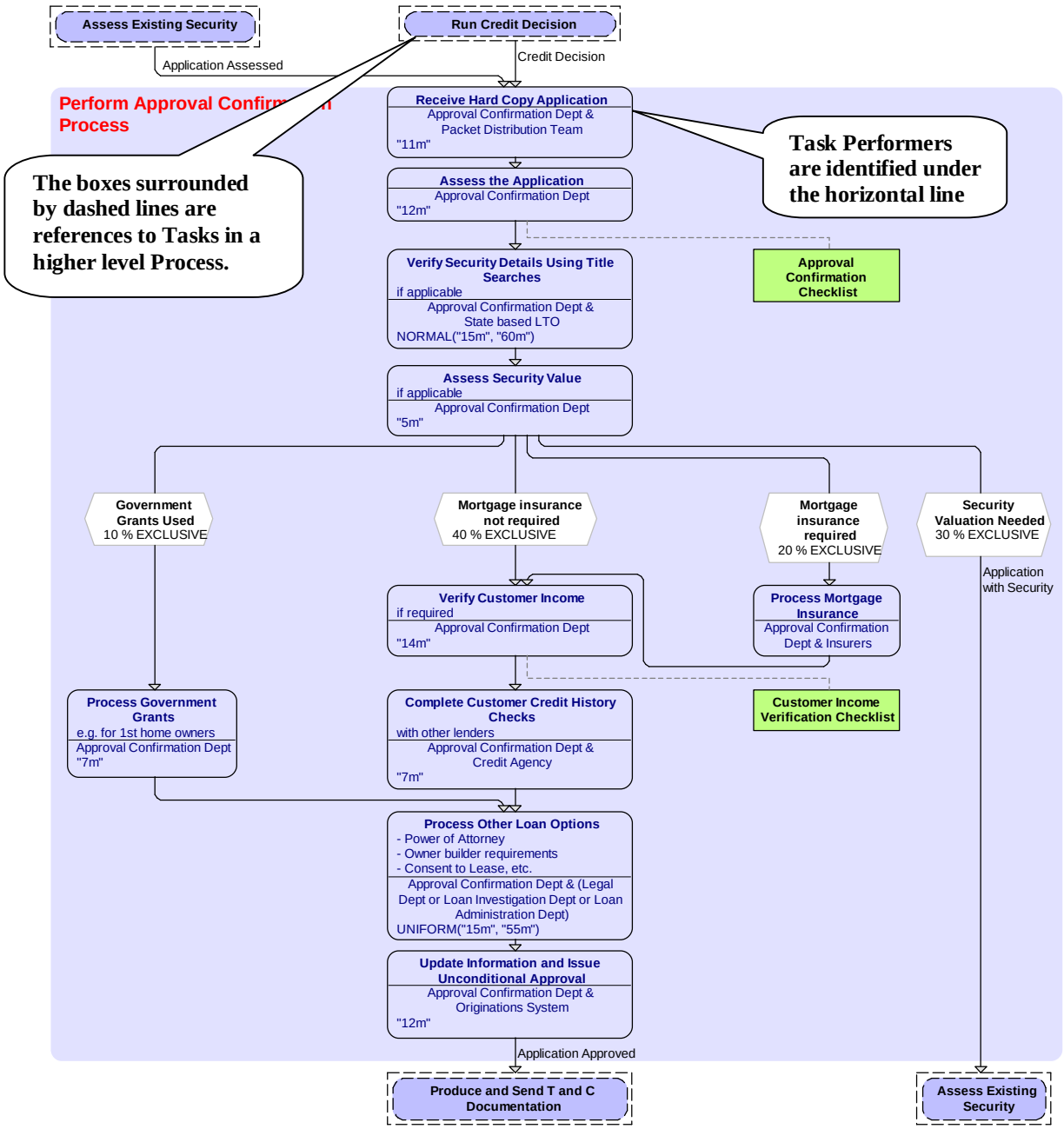
The EXIGEN icons signify that Exigen software products are used in that Task.
See (Fig. 8), for the AS-IS Process used in the simulation example to compare the AS-IS with the TO-BE.



## Fig. 8, AS-IS Version of "Perform Approval Confirmation"

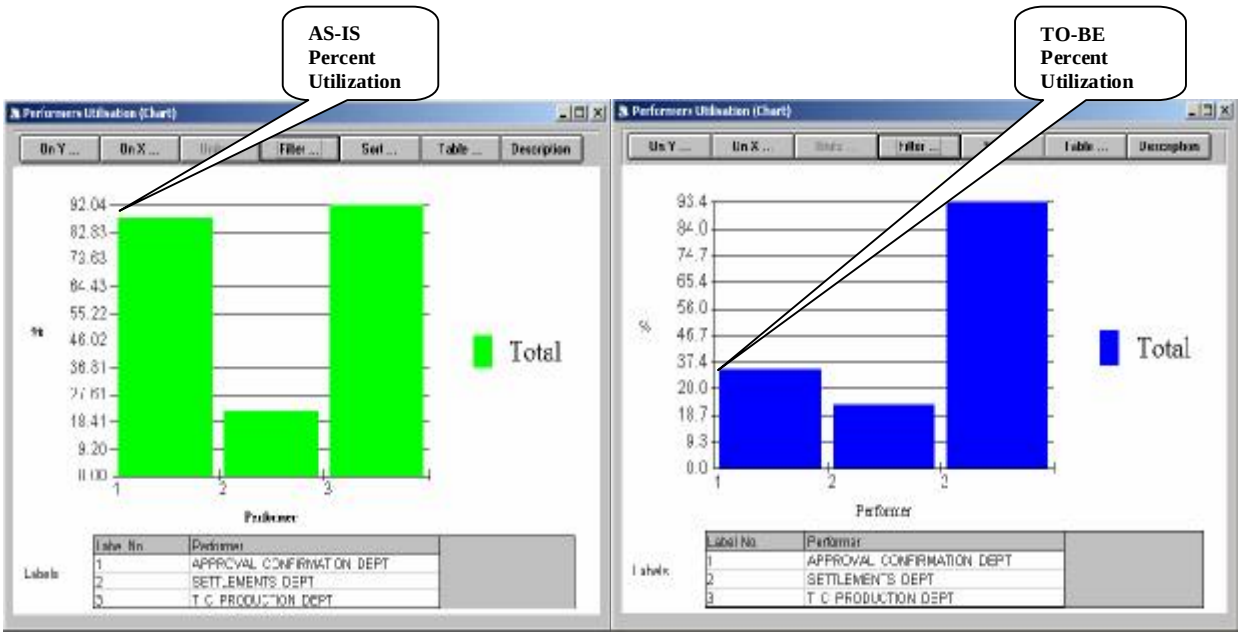
The numbers in the boxes e.g., "11m" means the Task takes 11 minutes.
The 10% EXCLUSIVE in the hexagonal box means that 10% of Loan Applications go this path.
The NORMAL ("15m","60m") means random distribution with average 15 minutes and standard deviation of 60 minutes.

These numbers are required only if simulation in planned; they may be hidden from view if desired, so as not to clutter the diagram.



## Fig. 9, Simulation Results – Resource Utilization

**Utilization of Human Resources in the "Approval Confirmation Department"**




## Fig. 10, Simulation Results – Process Duration

**Average Length of Time of "Perform Approval Confirmation" Process**

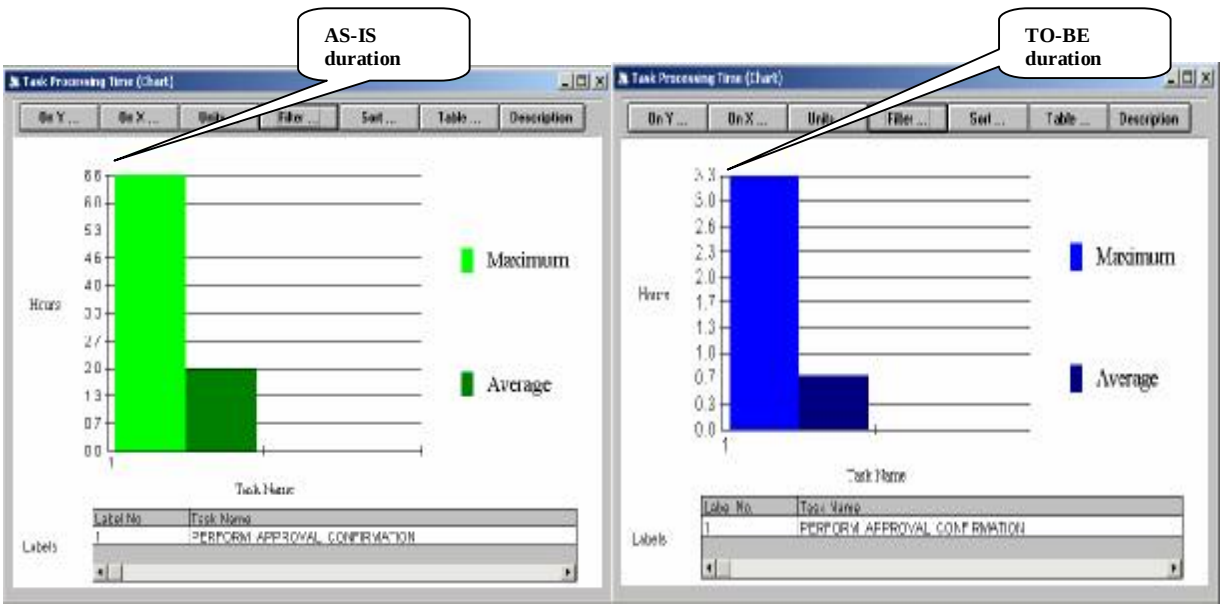


**Fig. 11, Snapshot of "Animation" During a Simulation Run**

The GRADE Simulator provides an "Animation" capability that permits the simulation run to be viewed and to observe where queues are building up due to inadequate resources, and the number of Tasks that are active at any particular time.

Animation is used to visually validate the model to ensure that there are no "dead" paths.