

## DEFINING THE AS-IS STATE OF AN ENTERPRISE - OVERVIEW

### BACKGROUND

The methodology provides detailed procedures for rapidly gaining a thorough understanding of the processes, structure and interactions of an enterprise and documenting this in a form that is easy to understand by different audiences.

The methodology deviates from the ‘popular’ approaches, which generally take too much time, consume excessive resources, and produce inconsistent results.

The methodology is derived from experience gained in many large-scale analysis projects. Since the GRADE tool was built to support the needs of such projects, it plays an important role in achieving quality results in record time.

The methodology is based on information gathering in the following ways:

- Ø Forms filled out by a customer,
- Ø Interviews,
- Ø Diverse customer documents.

The information is selectively and systematically entered into an enterprise model structure prepared from preliminary knowledge about the enterprise. The model evolves into a repository of all relevant facts about the enterprise according to the principle, “**a place for everything and everything in its place**”.

The model is a graphical and tabular representation of the enterprise. The model serves (a) To analyze & evaluate organizational, process, and interaction aspects with the objective of optimizing them (b) To estimate the manpower savings and improved service levels achievable by automating the processes, (c) To provide a clear basis for systematically developing a TO-BE model, with subsequent risk free transformation of the AS-IS state to a TO-BE state

### TERMINOLOGY

**Model** – A graphical and tabular representation of the Organization, IT systems, Processes, Interactions and Interfaces of an enterprise. A Model is a snapshot of the AS-IS state or a particular stage of a TO-BE state, and never a mix of these.

**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:

- Ø **Target System** – Organization & systems to be analyzed, e.g., Mortgage Processing Center.
- Ø **External Entity** – Active Object of the environment that the Target System interacts with, e.g., Loan Request Initiators.
- Ø **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.

**Contextual Field** – The context within which the Target System exists in terms of the External Entities it interacts with.

### PRIMARY STEPS IN DEFINING THE AS-IS STATE

This methodology is applicable to engagements lasting several days to several months. Only some of the steps must be done during short engagements. As the engagement scope increases, additional steps or greater depth of already completed steps is required.

**Step 1:** Defining the Contextual Field (**Fig. 1**); see (**Fig. 13**) for a Template example

**Step 2:** Capturing External Triggers and Outputs (**Fig. 2**)

**Step 3:** Capturing Organizational and IT Resource Units (**Fig. 3**)

**Step 4:** Identifying Tasks that service External Triggers and arranging them hierarchically (**Fig. 4, Fig. 5, Fig. 6**)

**Step 5:** Arranging Tasks in dependency sequence (**Fig. 7, Fig. 8**)

**Step 6:** Capturing Internal Triggers and Outputs (analogous to Step 2)

**Step 7:** Identifying & Mapping Processes that service Internal Triggers (analogous to Step 4&5)

**Step 8:** Associating Tasks with Performers (**Fig. 8**)

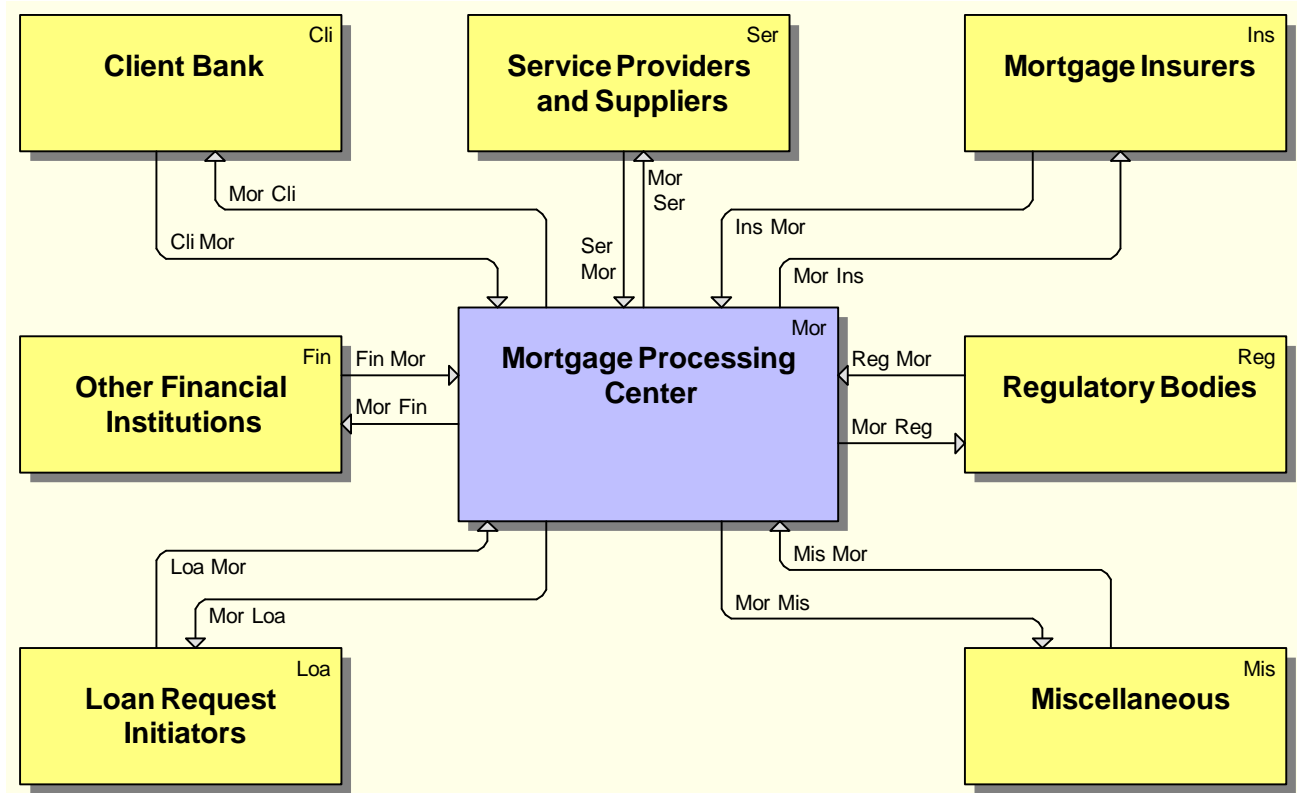
**Step 9:** Capturing quantifiable attributes of Resource Units and Tasks (**Fig. 8**)

**Step 10:** Validating the AS-IS Model (**Fig. 9, Fig. 10, Fig. 11**)

**Optional Step 11:** Comparing AS-IS with TO-BE through Dynamic Simulation (see **TO-BE** chapter)

EXAMPLES

**Fig. 1 Contextual Field Diagram**

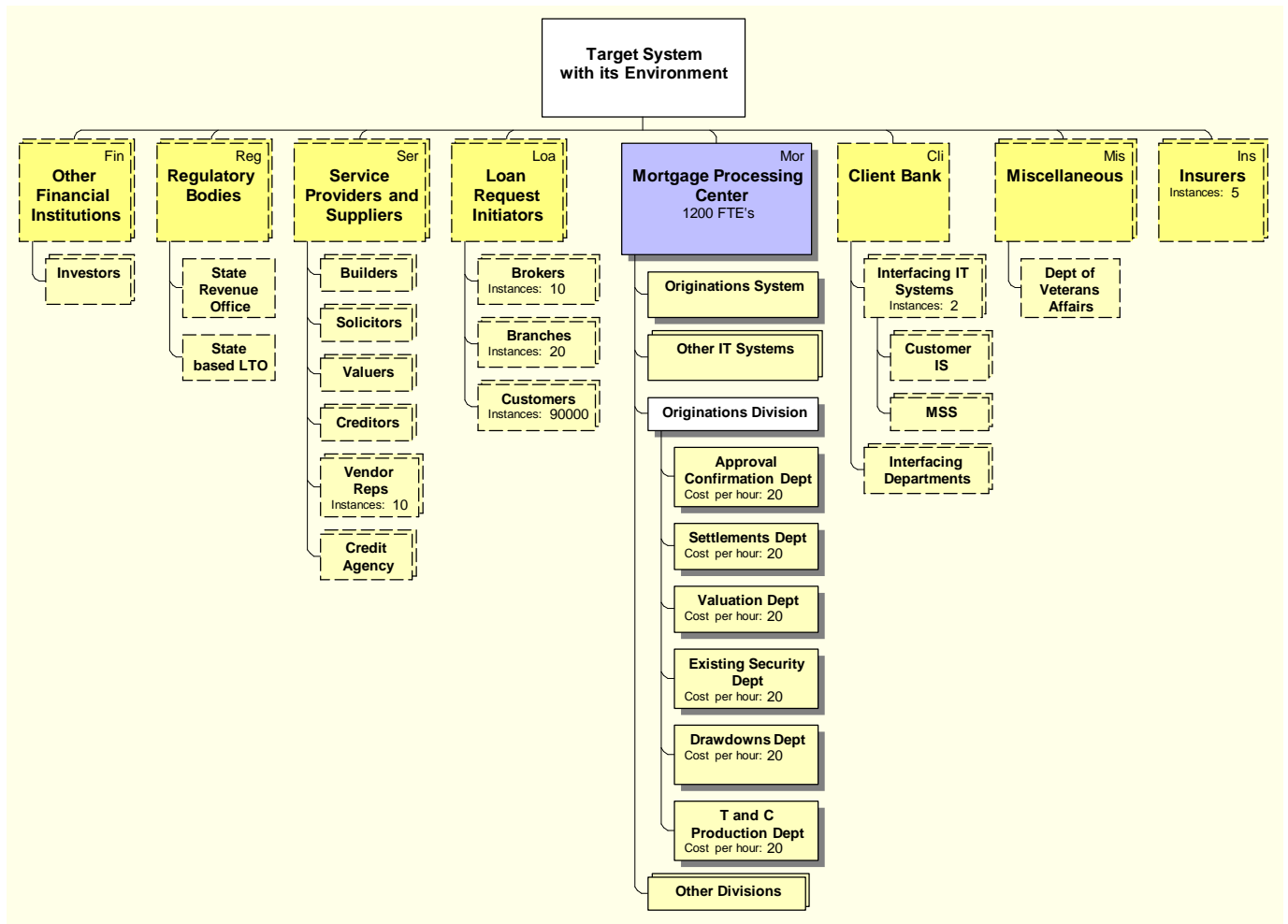


**Fig. 2 External Triggers and Outputs** (fragment)

No:	TRANSFER OBJECT/ EVENT	DESCRIPTION	SENDER	RECEIVER	TRANSFER MODE	LINK TO PROCESS
1	<b>Loan Application</b>	A package consisting of: - Application - Income verification - Property information - etc.	Loan Request Initiators	Mortgage Processing Center	Mail, Hand Delivery, FAX	<a href="#">Loan Application Processing</a>
2	<b>Application Status Inquiry</b>	Inquiry Types: - Approval Status - Decision expectation - etc.	Loan Request Initiators	Mortgage Processing Center	Phone, FAX, email	<a href="#">Inquiry Processing</a> *

\*Several Transfer Objects may have the same Process Links

**Fig. 3 Organizational and IT Resource Units** (fragment)



**NOTE:** This information should come largely from existing organization charts and IT configuration documents. This can be done in parallel with other activities but with the caution that, when identifying Performers of Tasks or finalizing Senders & Receivers in Transfer Object Tables, these Names should be picked from the ORG diagram.

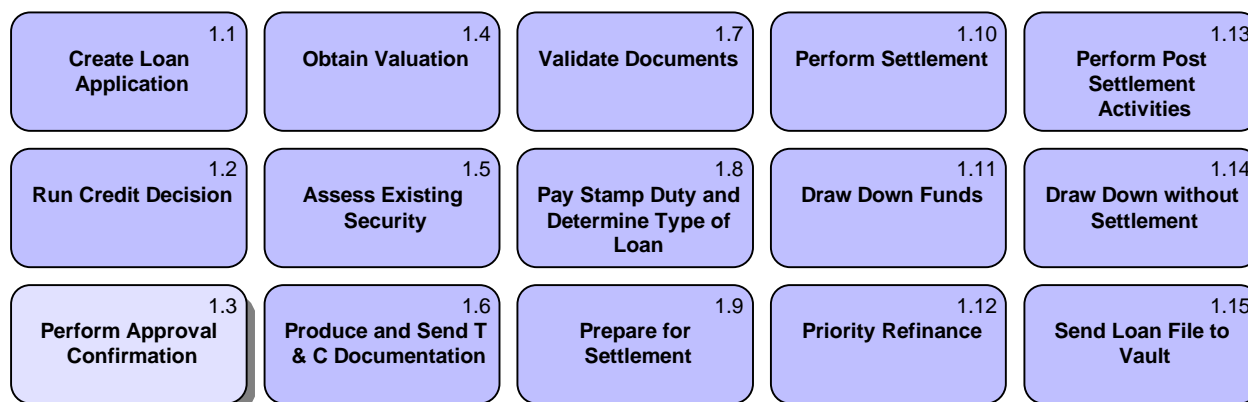
**Fig. 4 First Level Process Inventory of the Mortgage Processing Center** (fragment)



**NOTE:** Boxes with shadows are further refined below and “Loan Application Processing” is shown as a Process Diagram in Fig. 7.

**NOTE:** The **W** icon represents a hyperlink from GRADE to the client’s documents, such as, procedures manuals or existing Visio maps, etc., which describe the Loan Application Process. Such hyperlinks should be used throughout the model to relevant materials outside the model, forming a web-navigable knowledge base.

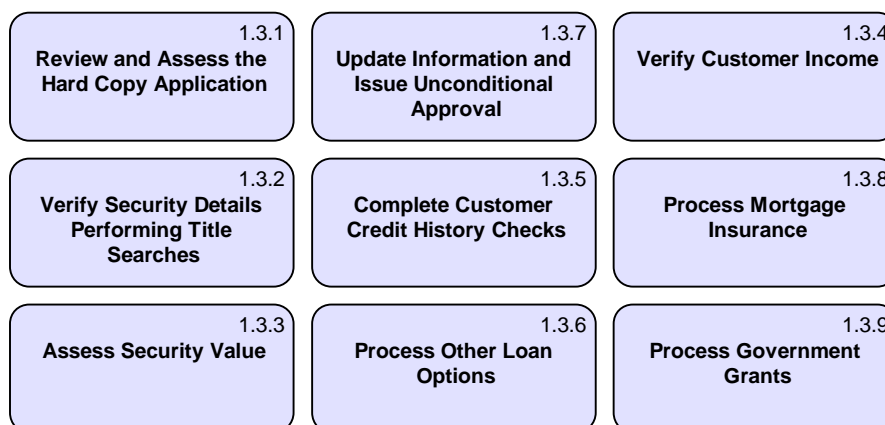
**Fig. 5 Second Level Task Inventory of the Process “Loan Application Processing”**



The Tasks shown in Fig.5 constitute a **Primary Process**, i.e., externally triggered end-to-end business transaction as shown in (Fig. 7). **Secondary Process** Tasks, i.e., internally triggered end-to-end business transaction, may need to be captured as well. If Workflow Management implementation is envisioned, **Supporting Process** Tasks, i.e., Service processes that support Primary and Secondary Processes, need to be captured.

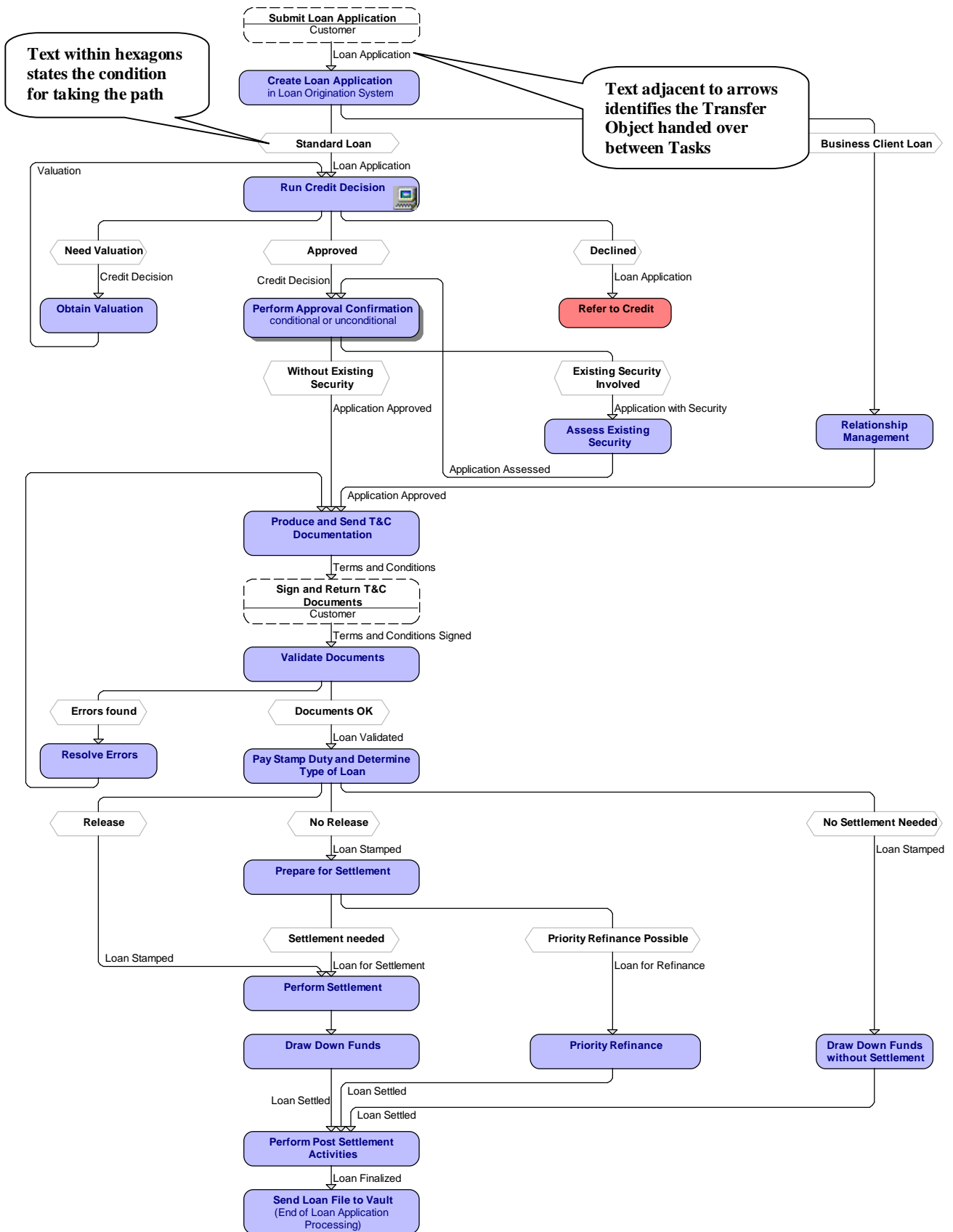
**NOTE:** Such inventory of Processes/Tasks would be created based on Exigen’s Task inventory (to be compiled) from experience with an LOB, in this case “Mortgage Processing”.

**Fig. 6 Third Level Task inventory of the Process “Perform Approval Confirmation”**



This level of detail should be captured for each Second Level Task. If the scope of the engagement covers only a few processes, mapping of the task sequences and associating the tasks with Performers may be called for.

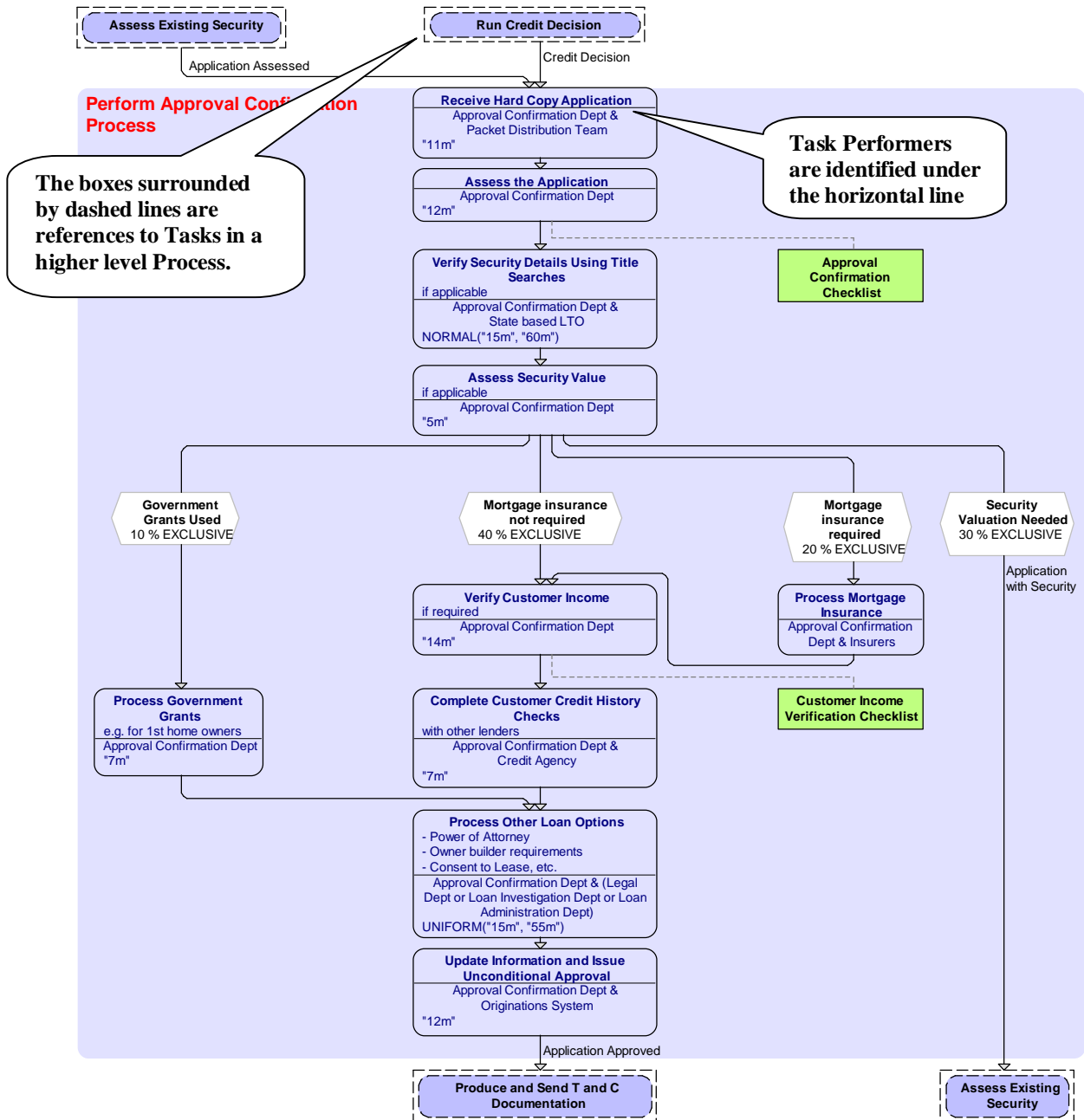
**Fig. 7 Second Level Tasks of “Loan Application Processing” in Dependency Sequence**



**Fig. 8 Third Level Tasks of “Perform Approval Confirmation” in Dependency Sequence, with Performers & Simulation Parameters**

NOTE: 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.

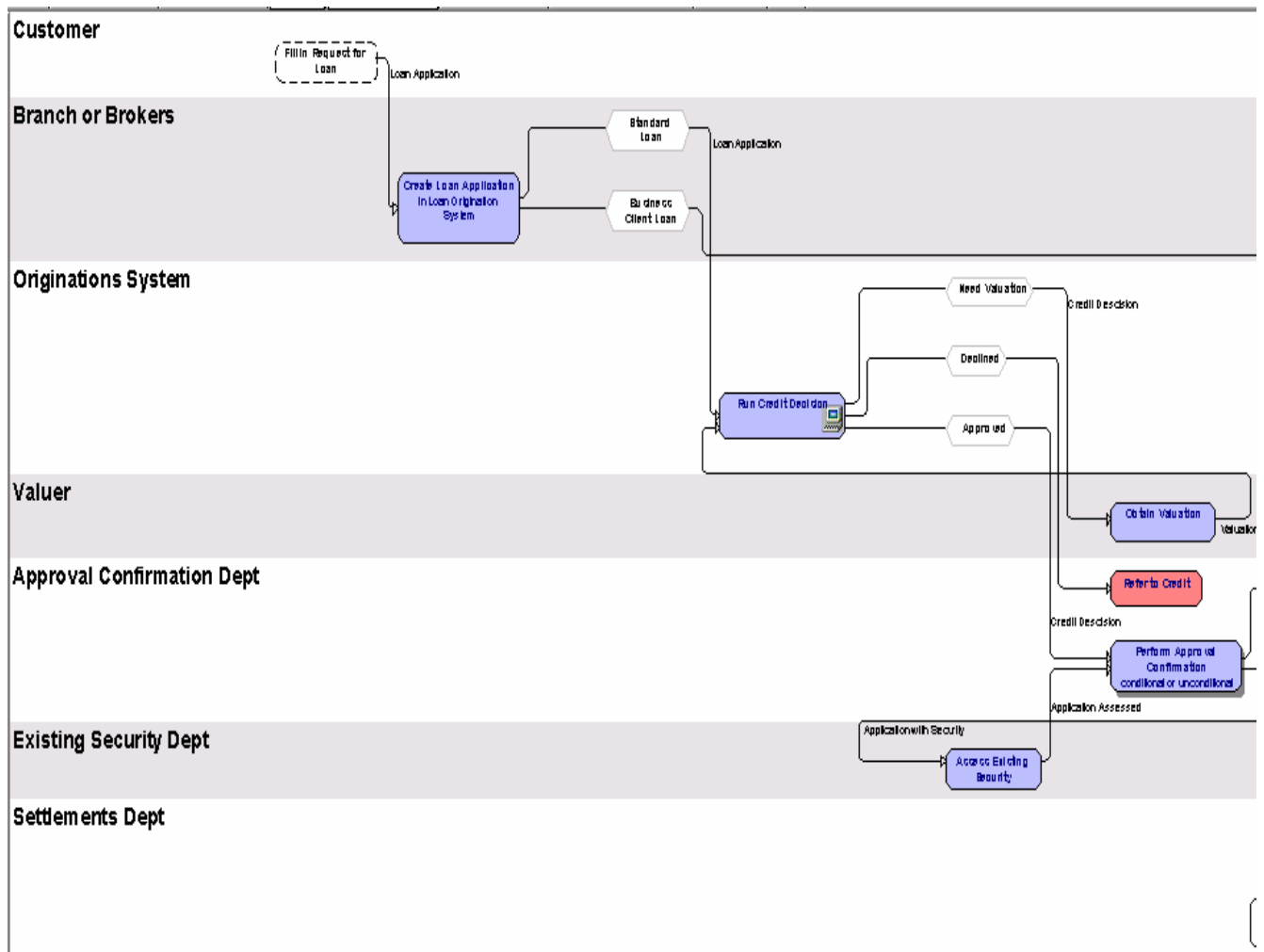
These numbers are required only if simulation is planned; they may be hidden from view if desired, as in Fig. 7, so as not to clutter the diagram.



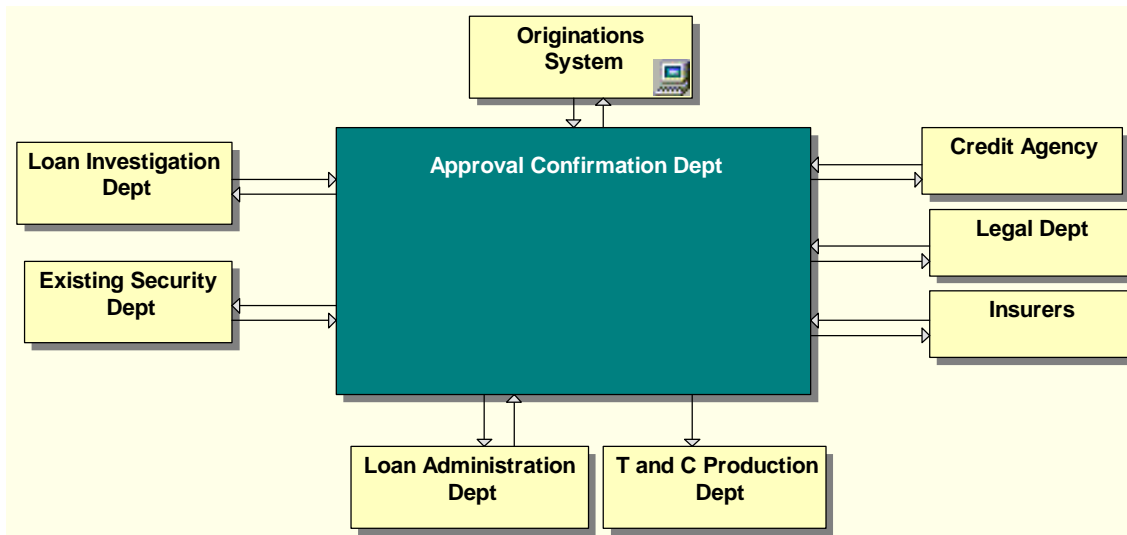
**Fig. 9 Swim Lane view of a Process**

This is one of several different views obtained automatically to aid validation. It highlights the Process Task handovers between different Performers.

A “Horizontal Lanes” view is most useful if a plotter is available to print large, say, 4\*10 foot diagrams for wall viewing. A “Vertical Lanes” view is most useful for Web viewing.



**Fig. 10 Approval Confirmation Department's Interactions**



This is another automatically obtained view to be used when validating the model with the Approval Confirmation Department. Associated with the arrows are the Transfer Objects displayed in the Table below.

**Fig. 11 Approval Confirmation Department's Transfer Objects**

TRANSFER OBJECT/EVENT	SENDER	RECEIVER	LINK TO TRIGGERED PROCESS
Application Approved	Approval Confirmation Dept	T and C Production Dept	Task [Update Information and Issue Unconditional Approval] -> Task [Produce and Send T and C Documentation]
Application with Security	Approval Confirmation Dept	Existing Security Dept	Task [Assess Security Value] Decision Security Valuation Needed -> Task [Assess Existing Security]
Application Assessed	Existing Security Dept	Approval Confirmation Dept	Task [Assess Existing Security] -> Task [Perform Approval Confirmation]
Credit Decision	Originations System	Approval Confirmation Dept	Task [Run Credit Decision] Decision Approved -> Task [Perform Approval Confirmation]
Loan Application	Originations System	Approval Confirmation Dept	Task [Run Credit Decision] Decision Declined -> Task [Refer to Credit]

**NOTE: The column “LINK TO TRIGGERED PROCESS” shows on the first line the Task in the “Perform Approval Confirmation” Process that sends the Transfer Object. On the second line is the Task that receives it.**

Since a model in GRADE is consistent, based on non-redundant information organized in terms of Performers, Processes and Transfer Objects, views relevant for a specific user may be generated automatically. It is very effective to review and correct individual fragments, rather than the entire model, knowing that GRADE will maintain model consistency.

**NOTE: Complete navigation between the different model aspects is provided in HTML so that a reviewer does not need to know GRADE but can browse through the model via the Web. The models are intuitive enough to require no special training.**

## CATEGORIZING INFORMATION IN A GRADE MODEL

In order to keep the complexity of an Enterprise model manageable, information is categorized according to “Basic Building Blocks” of socio-technological systems, as described below. Each of these may be drilled down to expose further details. GRADE maintains full cross-correlations between all levels of all Building Blocks..

**Active Objects** – physical entities that perform processes, e.g.:

- Ø Mortgage Center (an organization);
- Ø Loan Originations system (an IT system);
- Ø Approval Confirmation Team Member (a person);
- Ø Printer (a device or resource);

Active Objects are represented via Communicating Objects (CO) and Organization (ORG) diagrams.

**Passive Objects** – information or “things” which are used or manipulated by active objects, e.g.:

- Ø Loan Application (paper based documents or oral messages);
- Ø Data files/ folders, Database queries (electronic message or variable in the information system);
- Ø Packets with documents, Components and parts (physical objects used in production processes);
- Ø Printed reports (listing obtained from electronic file), etc.

Passive Objects are represented via Transfer Object (TO) tables, Data Definition (DD) diagrams and Entity Relationship (ER) diagrams.

**Processes** – behavior of Active Objects, either in the form of Task sequences and rules by which the Active Object reacts to Triggers producing outputs, e.g.:

- Ø Originate New Loan (process of the Mortgage center as a whole);
- Ø Customer data servicing (processes supported by the legacy information system);
- Ø Document processing (process of the Settlements Department);
- Ø Report printing (activity performed by the Printer), etc.

Processes are represented via Business Process (BP) diagrams.

GRADE supports a view of reality, based on tangible “things” that are objects of the real world, rather than their abstractions, as in software engineering. It is based on a premise that when dealing with complex socio-technological systems, understanding what software needs to be developed, and validating that with the users, represents a far bigger challenge than developing the software.

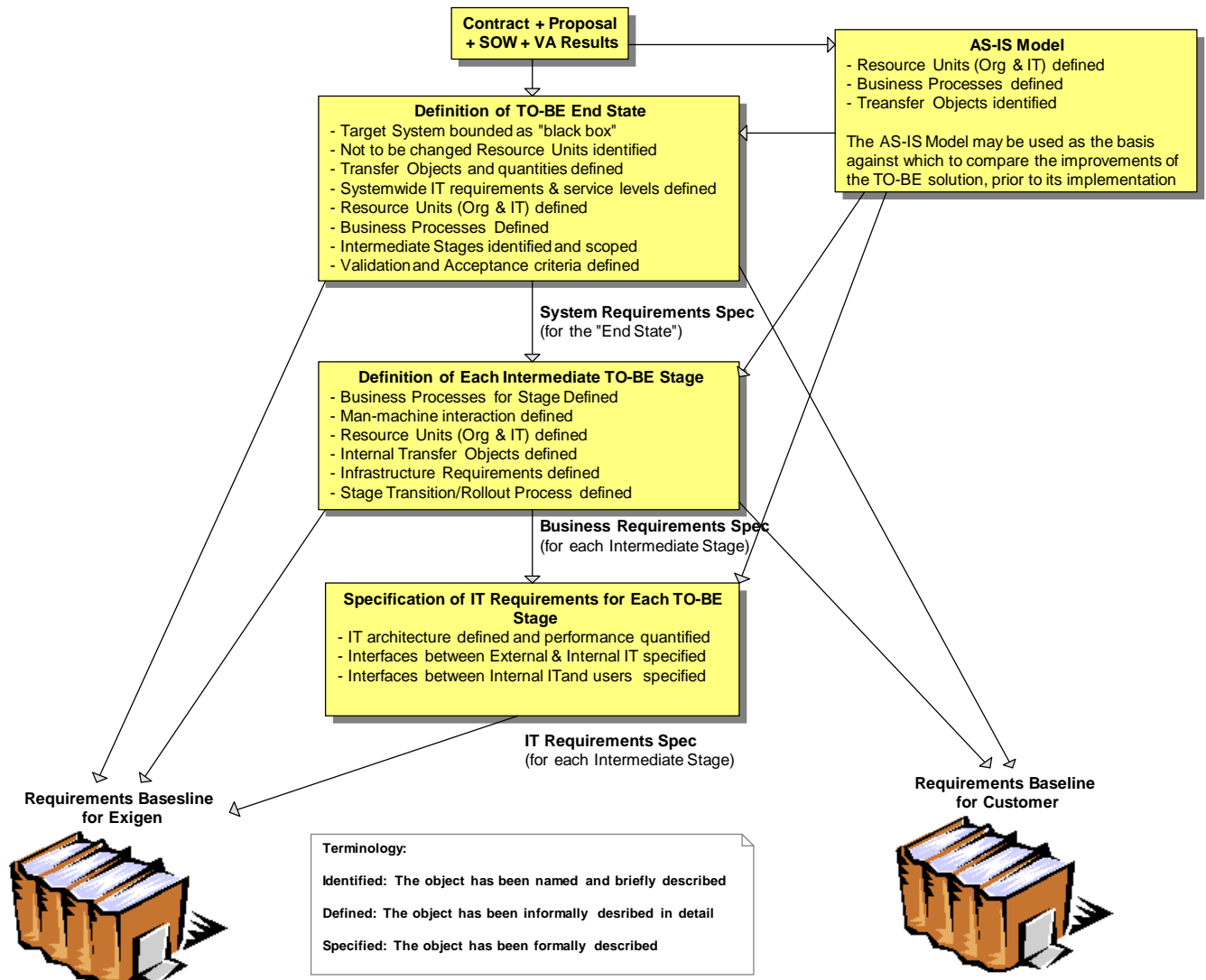
Since Functional Decomposition and Object Oriented software design requires an abstraction of reality in a specific way, user requirements are difficult to validate and are often left poorly defined. Therefore, for Business analysis it is wise to avoid methodologies designed to serve software specification, such as the very popular UML.

Software engineering has focused heavily on tooling for speedy construction of software, offering few practical methodologies for understanding and defining what needs to be constructed. Consequently software applications often, instead of solving user needs, turn out to be “solutions looking for a problem”.

#### **GRADE MODELS AND THEIR ROLE IN REQUIREMENTS BASELINE\***

This diagram shows how the AS-IS Model relates to the TO-BE Models and their role in the specification of requirements for the “End State” and the Intermediate Stages to be “rolled out”.

**Fig. 12 Creating Requirements Baseline Using GRADE Models**



\***Baseline** is a Change Controlled document.

The Requirements Baseline for Client specifies the commitments the Client has to fulfill.  
 The Requirements Baseline for Exigen specifies the commitments that Exigen has to fulfill.

The first "Baseline" document is the **System Requirements Spec**, which defines the TO-BE End State, encompassing organizational, technological and process aspects.

For each Intermediate Stage there is a "Baseline" **Business Requirements Spec**, which scopes the "next" implementation project.

Corresponding to the Business Requirements Spec for each Intermediate Stage is a "Baseline" **IT Requirements Spec**.

**TEMPLATE EXAMPLE**

**Fig. 13 Information Gathering for the Contextual Field Diagram**

**Step 1: Identify the organizations which will not be analyzed/changed as part of the LOB transformation, but with which the Utility must interact.**

See Fig. 14 for the corresponding Context diagram.

Category	Type	Count	Comments
<b>Customers</b>			The customers to be served by the Utility
	Ø Individuals	?	
	Ø Brokers	?	
	Ø Etc.		
<b>Client</b>			The 'parent' organization, which the Utility will be serving
<b>Other Financial Institutions</b>			Partner banks, lenders, etc., that the Utility will interact with
	Ø Banks	?	
	Ø Etc.		
<b>Governmental Agencies</b>			Regulatory bodies and agencies that the Utility will be interacting with
	Ø State	?	
	Ø Federal	?	
	Ø County/City	?	
	Ø Etc.		
<b>Insurers</b>			Insurance the Utility will interact with
	Ø Property	?	
	Ø Casualty	?	
	Ø Mortgage	?	
	Ø Etc.		
<b>Service Providers</b>			Suppliers of products and services that the Utility will interact with
	Ø Credit Agency	?	
	Ø Assessors	?	
	Ø Collectors	?	
	Ø Etc.		
<b>Other LOBs</b>			LOBs, other than the Client LOB, that the Utility will interact with
	Ø Pers. Loans		
	Ø Credit Cards		
	Ø Etc.		
<b>Miscellaneous</b>			Various other parties the LOB interacts with, that don't fit other categories
	Ø Loan co-signer		
	Ø Employer		
	Ø Etc.		

- Standardized “**Categories**” can be developed for each Vertical.
- A superset of “**Types**” can be defined and relevant ones picked during an engagement.
- “**Count**” refers to the number of instances of the “**Type**”

**Step 2: Identify the IT Systems which will not be analyzed/changed as part of the LOB transformation, but with which the Utility must interact.**

<b>Category</b>	<b>Applications</b>	<b>Comments</b>
<b>Customers</b>	Applications (a) ? (b) ? (c) Etc.	
<b>Client</b>	Core Systems (a) ? (b) ? (c) Etc.	
<b>Other Financial Institutions</b>	Applications (a) ? (b) ? (c) Etc.	
<b>Governmental Agencies</b>	Applications (a) ? (b) ? (c) Etc.	
<b>Insurers</b>	Applications (a) ? (b) ? (c) Etc.	
<b>Service Providers</b>	Application (a) ? (b) ? (c) Etc.	
<b>Other LOBs</b>	Applications (a) ? (b) ? (c) Etc.	
<b>Miscellaneous</b>	Applications (a) ? (b) ? (c) Etc.	

**Fig. 14**

# Top Level Contextual Field Diagram

"Type" and "Count" Information is not shown

