



ľ

2

MICROCOPY RESOLUTION TEST CHART NATIONAL BUREAU OF STANDARUS 1967 A



NAVAL POSTGRADUATE SCHOOL Monterey, California

AD-A152 101

FILE COPY

JU

O





THESIS

DATA DICTIONARY DESIGN AS A STEPPING-STONE TO DBMS IMPLEMENTATION IN THE INDONESIAN ARMY DATA COLLECTING AND PROCESSING SERVICE

by

Bambang Sutedjo

September 1984

Thesis Advisor:

Daniel R. Dolk

Approved for public release; distribution unlimited

85 03 18 114

REPORT DOCUMEN	NTATION PAGE READ INSTRUCTIONS BEFORE COMPLETING FORM
REPORT NUMBER	2. GOVT ACCESSION NO. 3. RECIPIENT'S CATALOG NUMBER
	AD-A152101
TITLE (and Subtitle)	5. TYPE OF REPORT & PERIOD COVERE
Data Dictionary Design	n as a Stepping- Master's Thesis
stone to DBMS Implement	ntation in the Indo- September 1984
nesian Army Data Colle	ecting and 6. PERFORMING ORG. REPORT NUMBER
Processing Service	A CONTRACT OF CRANT NUMBER(A)
AUTHOR(3)	a. CONTRACT OR GRANT NUMBER()
Bambang Sutedio	
PERFORMING ORGANIZATION NAME AN	ND ADDRESS 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
Naval Postgraduate Scl	hool
Monterey, CA 93943	
CON ROLLING OFFICE NAME AND AD	DRESS 12. REPORT DATE
Naval Postgraduate Scl	13. NUMBER OF PAGES
Monterey,CA 93943	85
MONITORING AGENCY NAME & ADDRE	ESS(If different from Controlling Office) 15. SECURITY CLASS. (of this report)
	UNCLASSIFIED
	15- DECLASSIFICATION DOWNGRADING
	SCHEDULE
DISTRIBUTION STATEMENT (of the abo	erease, distribution unrinnited
. DISTRIBUTION STATEMENT (of the abe	erease, distribution unrinnited
. DISTRIBUTION STATEMENT (of the abo	erease, distribution unifinited
APPTOVED TOT PUDITC T DISTRIBUTION STATEMENT (of the abo SUPPLEMENTARY NOTES	Inecessary and Identify by block number)
EY WORDS (Continue on reverse elde if Data Dictionary; DBMS Dictionary Schema: DI	Increase, distribution unifinited fraceseery and identify by block number) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features:
SUPPLEMENTARY NOTES KEY WORDS (Continue on reverse elde if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi	Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes
SUPPLEMENTARY NOTES KEY WORDS (Continue on reverse eide if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi	Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes
Approved for public fr DISTRIBUTION STATEMENT (of the abo SUPPLEMENTARY NOTES KEY WORDS (Continue on reverse elde if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi	Intract entered in Block 20, if different from Report) Intract entered in Block 20, if different from Report) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes
Approved for public fr DISTRIBUTION STATEMENT (of the ebe SUPPLEMENTARY NOTES KEY WORDS (Continue on reverse elde if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi ABSTRACT (Continue on reverse elde if f	Inscreece, distribution unifinited Inscreece in Block 20, if different from Report) Inscreece and identify by block number) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes Inscreece and identify by block number) Increece and identify by block number) Increece and identify by block number)
Approved for public fr DISTRIBUTION STATEMENT (of the abo SUPPLEMENTARY NOTES KEY WORDS (Continue on reverse elde if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi ABSTRACT (Continue on reverse elde if the organizational str the Indonesian Army Da	Increase, distribution unifinited (necessary and identify by block number) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes necessary and identify by block number) ucture, tasks, and system configuration of ta Collecting and Processing Service
Approved for public fr DISTRIBUTION STATEMENT (of the abo SUPPLEMENTARY NOTES KEY WORDS (Continue on reverse elde if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi ABSTRACT (Continue on reverse elde if the organizational str the Indonesian Army Da DISPULLAHTAD) are pre	Inscreed in Block 20, If different from Report) Inscreed in Block 20, If different from Report) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes Incomeany and identify by block number) Tucture, tasks, and system configuration of ta Collecting and Processing Service sented briefly in order to provide a back-
- DISTRIBUTION STATEMENT (of the ebe - SUPPLEMENTARY NOTES - KEY WORDS (Continue on reverse eide if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi - ABSTRACT (Continue on reverse eide if the organizational str the Indonesian Army Da DISPULLAHTAD) are pre tround for succeeding	Increase, distribution unifinited Anter and identify by block number) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes Increase and identify by block number) ucture, tasks, and system configuration of ita Collecting and Processing Service sented briefly in order to provide a back- discussions. Features of available data
- DISTRIBUTION STATEMENT (of the abo - SUPPLEMENTARY NOTES - SUPPLEMENTARY NOTES - SUPPLEMENTARY NOTES - SUPPLEMENTARY NOTES - Dictionary Schema; DI Entities; Relationshi - ABSTRACT (Continue on reverse elde II r The organizational str - the Indonesian Army Da DISPULLAHTAD) are pre ground for succeeding lictionary systems (DD	<pre>Precase, distribution unifinited Instruction Block 20, If different from Report) Instruction Block 20, If different from Report) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes necessery and identify by block number) ructure, tasks, and system configuration of ta Collecting and Processing Service sented briefly in order to provide a back- discussions. Features of available data OS) and the initial design of a data diction</pre>
- DISTRIBUTION STATEMENT (of the ebe - SUPPLEMENTARY NOTES - KEY WORDS (Continue on reverse elde if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi - ABSTRACT (Continue on reverse elde if the organizational str the Indonesian Army Da DISPULLAHTAD) are pre ground for succeeding lictionary systems (DD try for current applic	<pre>Precessery and Identify by block number) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes necessery and identify by block number) Ucture, tasks, and system configuration of ta Collecting and Processing Service sented briefly in order to provide a back- discussions. Features of available data OS) and the initial design of a data diction ations at DISPULLAHTAD are presented.</pre>
- DISTRIBUTION STATEMENT (of the abo - SUPPLEMENTARY NOTES - KEY WORDS (Continue on reverse elde if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi - ABSTRACT (Continue on reverse elde if the organizational str the Indonesian Army Da DISPULLAHTAD) are pre ground for succeeding lictionary systems (DD try for current applic - Finally, based on that	<pre>Precase, distribution diffinited Precessery and identify by block number) Implementation; Indonesian Army; Data SPULLAHTAD; Data Dictionary Features; ps; Attributes Precessery and identify by block number) Precessing Service Sented briefly in order to provide a back- discussions. Features of available data DS) and the initial design of a data diction ations at DISPULLAHTAD are presented. initial design, a recommendation for data- </pre>
Approved for public from public for public for public for public for the ebe SUPPLEMENTARY NOTES KEY WORDS (Continue on reverse eide if Data Dictionary; DBMS Dictionary Schema; DI Entities; Relationshi ABSTRACT (Continue on reverse eide if for the organizational str the Indonesian Army Da DISPULLAHTAD) are pre pround for succeeding lictionary systems (DD ary for current applic inally, based on that base management system	<pre>""""""""""""""""""""""""""""""""""""</pre>

(

Approved for public release; distribution unlimited.

Data Dictionary Design as a Stepping-stone to DBMS Implementation in the Indonesian Army Lata Collecting and Processing Service

Ły

Bambang Sutedjo Cartain, Indonesian Army Armed Forces Military Academy, 1973

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAI POSTGRADUATE SCHOOL September, 1984

Bambang Sutedjo

Author:

1

.

Approved by: Laniel Roy Dolk, Thesis Advisor Morman R. Jyons, Co-Advisor Worman R. Jyons, Co-Advisor Willis R. Greer, Jr., Chairman, Department of Administrative Sciences

Kneale T. Marshall, Dean of Information and Policy Sciences

ABSIFACT

The organizational structure, tasks, and system configuration of the Indonesian Army Data Collecting and Processing Service (DISPULLAHTAE) are presented briefly in order to provide a background for succeeding discussions. Features of available data dictionary systems (DDS) and the initial design of a data dictionary for current applications at DISPULLAHTAD are presented. Finally, based on that initial design, a recommendation for database management system (DBMS) implementation is discussed.



TABLE OF CONTENTS

(

I.	INTI	RODU	CTIO	N -	•	• •	•	•	•	•	• •	•	•	•	•	•	•	•	•	•	8
II.	INDO	ONES	IAN	ARM	YC	ATA	c	LI	EC	TI	NG	A N	D	PRO	OCE	ESS	I!	IG			
	SER	VICE	•	• •	•	• •	•	•	•	•	•	• •	•	•	•	•	•	•	•	•	10
	Α.	ORG	ANIZ	ATI	CN,	TA	SK	, A	A N D	S	YSI	TEM									
		CONI	FIGU	RAT	ION	Ι.	•	•	•	•	• •	• •	•	•	•	•	•	•	•	•	10
		1.	PUL	LAH	TAs	s in	sid	le	Ja	va	Īs	sla	n đ	•	•	•	•	•	•	•	12
		2.	PUL	LAH	IAS	5 OU	itsi	ld€	e J	a v	a 1	[sl	a n	đ	•	•	•	•	•	•	12
	B.	APP	LICA	TIC	NS	• •	•	•	•	•	• •	•	•	•	•	•	•	•	•	•	12
		1.	Adm	ini	sti	ati	on	Ma	ina	ge	ner	nt	In	foi	cma	ati	. 0 I	נ			
			Sys	tem	•	• •	•	•	•	•	• •		•	•	•	•	•	•	•	•	12
		2.	Mil	ita	ry	Mar	nage	eme	ent	I	nfo	orm	at	io	n S	Sys	ete	<u>ت</u>	•	•	12
		3.	Pla	nni	rg	anð	L C	ont	ro	11	ing	j M	an	age	ene	ent	:				
-			Inf	orm	ati	on	Sys	ste	e m		• •		•	•	•	•	•	•	•	•	13
	c.	DAT	A RE	DU N	LAN	ICY	PRO	DEI	LEM	S				•	•	•	•	•			13
	D.	STA	GED	DEV	ELC	PME	NT	AE	PR	O A	СН	•	•	•	•	•	•	•	•	•	13
TTT.	DATI		 TO	NAE	7		_	_		-			_	_	-	-	_	_	_		17
	Δ	GEN	FPAT		-	•••	•	•	•	•	• •		•	•	•	•	•	•	•	•	17
	л. Э	GLN.		• • • • •	• N T		•	•	•	•	••••	• • • • •	•	•	• • • •	•	•	•	•	•	10
	D •		JRHA	110 - D	N 1 			- E	na -	AR.	62: 	IEN Bee	1	(1)	נו. א דד		•	•	•	•	10
		1.	Dat	ab	101	.101	ar	r a	IS	τη	e	100	T	OI	Τt	112	•	•	•	•	18
		2.	Dat	a D	101	:101	lar	7	•	•	• •	• •	•	•	•	•	•	•	•	•	13
	_	3.	Dat	a D	ict	:101	ar	1 5	Sys	te	m	(DD	S)	٠	٠	•	•	•	•	•	19
	с.	FEA:	FURE	S C	FA	DA	TA	DI	ICT	10	NAI	RY	•	•	•	•	٠	•	•	•	21
		1.	Imp	lem	ent	ati	.on	ar	nd	Ar	chi	ite	ct	ura	11						
			Iss	ues	•	• •	٠	•	•	•	• •	•	•	•	٠	•	٠	•	•	•	21
		2.	Dat	a D	ict	ion	ar	i a	nd	D	ata	a D	ic	ti	ממכ	ary	,				
			Sch	ema	•	• •	•	•	•	•	•	• •	•	•	•	•	•	•	•	•	24
		3.	Ext	ens	ibi	.lit	y 1	Fac	;il	it	ies	5.	•	•	•	•	•	•	•	•	29

		4.	Stat	us F	ac	ili	ti	es		•	•	•	•	•	•	•	•	•	•	•	•	30
		5.	Dict	icna	ry	Co	шШ	an	ds	5	•	•	•	•	•	•	•	•	•	•	•	30
		6.	Brid	je F	ac	ili	ti	es		•	•	•	•	•	•	•	٠	•	•	•	•	32
		7.	DDS :	5ecu	ri	ty	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	32
	D.	cos	T/BENI	EFIT	A	NAL	۲S	IS	F	°0 F	Ð	DS		•	•	•	•	•	•	-	•	33
		1.	Cost	5.	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	33
		2.	Fact	ors	fo	rΞ	st	im	at	:ir	ij	Sa	vi	ng	s	ar	nđ					
			Bene	fits	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	35
		з.	Savi	ngs	an	đB	en	ef	it	s	•	•	•	•	•	•	•	•	•	•	•	36
IV.	IN	ITIAL	DATA	DIC	TIC	ONA	RΥ	D	ES	SIG	s n	FC	R									
	DI	SPULL	AHTAD	• •	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	3 ε
	Α.	GEN	ERAL	• •	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	33
	В.	DAT	A DIC	TICN	AR	y s	СН	ΕM	A/	/st	JBS	СН	ΕM	A	•	•	•	•	•	•	•	3 8
		1.	Data	Dic	ti	ona	ry	S	ch	ne	ıa	•	•	•	•	•	•	•	•	•	•	38
		2.	Data	Dic	ti	ona	ry	S	uŁ	sc	he	ma	s	•	•	•	•	•	•	•	•	40
	с.	DIS	PULLA	HTAD	• S	D A	TA	D	IC	T		AR	Y	•	•	•	•	•	-	•	•	41
		1.	Enti	ties	•	•	•	•	•	•	•	•	•	•	•		•	•	•	•	•	41
		2.	Rela	ticn	sh	ips		•	•	•	•	•	•	•	•	•	•	•	•	•	•	58
		3.	Attr	i but	es	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	59
		4.	Exam	ple	of	Da	ta	Ð	ic	ti	ion	ar	7	s	Qu	ıer	ie	s	•	•	•	72
۷.	TH:	E IMP	LEMEN	TATI	ON	OF	D	AT	AE	A S	SE	AI	2	IS	21	JLI	LAE	IT	E	•	•	74
	A.	DAT	A DIC	FICN	AR	Y D	ES	ÍG	N	A S	5 A	S	TE	PE	I!	NG -	-sı	101	Ξ	•	•	74
	Β.	DIS	PULLA	HIAD	• S	D A	TA	EA	SE	2 1	DES	IJ	N	•	•	•	•	•	•	•	•	74
	с.	CHO	OSING	THE	D.	ATA	D	IC	TI	01	IAR	Y	SY	SI	E	í	(DI)s)		•	•	7 5
		1.	Feat	ures	•	•	•	•	•	•	•	-	•	•	•	•	•	•	•	•	•	75
		2.	DISP	JILA	HT.	A D •	s	כס	s	Re	egu	ir	em	en	ts	5	•	•	•	•	•	77
		3.	Reco	nieu	da	tio	n	•	•	•	•	•	•	•	•	•	•	•	•	•	•	7 9
VI.	CC	NCLUS	ION.	• •	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	81
IIST	CF F	EFERE	NCES	••	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	83
INIT	IAL D	ISTRI	BUTIO	N II	ST	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•	85

and the factor of the second se

LIST OF TABLES

I.	Comprehensive Flan for DDS's Usage	ŧ
II.	Personnel Application Data Entities 42	2
III.	Payroll Application Data Entities 47	7
IV.	Intelligence Personnel Application Data	
	Entities	9
۷.	Territorial Personnel Application Data	
	Entities	۱
VI.	Process Entities	3
VII.	User Entities	7
VIII.	Relational Model for DISPULLAHTAD Dictionary	
	System	5
IX.	Examples of Data Entity Relations 67	7
Χ.	Examples of Process & User Entity Relations 69	9
XI.	Examples of Relationship Relation	1

LIST OF FIGURES

C

(a)

[🕫

 \mathbf{O}

2.1	Indonesian Army Organizational Structure
	(Simplified Chart)
2.2	Transactions by Application at DISPULLAHTAD 15
4.1	Logical Structure of DISPULLAHTAD'S DD
	(adapted from Allen et al [Eef. 5])
5.1	Criteria for Choosing Commercial Data
	Dictionary System
5.2	Features of Current Commercial DDS
	(summarized from Lefkovits et al [Ref. 1]) 80

I. INTRODUCTION

As a young and developing information system organization, the Indonesian Army Data Collecting and Processing Service (DISPULLAHTAE) has a tremendous proliferation of application files. It is no surprise that there is much redundancy of data and efforts. Data redundancy waster a limited resources, and furthermore, it raises the problem of inconsistent data, that is, the same element of data having different values within different files. The implementation of database management system (DBMS) could handle this problem by providing more control and more effective management of data.

ſ

(/

Cn one hand, the information generated electronically becomes more and more in demand to the point where it has become a critical issue for the Indonesian Army. Cr the other hand, the personnel generating and maintaining this information move dynamically because of requirements for military tour of duty and tour of area. This situation creates problems in keeping accurate and up to date information. Even though the manual documentation is always done properly, this is not always adequate. It is often the case that many applications are highly dependent up on the personnel responsible for such applications. Standardized and centralized documentation is a "must", especially when a DBMS is implemented. In this regard, the data dictionary is a powerful vehicle that supports such documentation.

According to Dolk [Ref. 1], "a data dictionary is a collection of an enterprise's meta-data designed as one or more databases which can be retrieved and analyzed using standard database management system capabilities". This will be discussed further in a subsequent chapter, and will

te considered as a fasis in choosing the most appropriate DBMS to be implemented by DISPULLAHTAD.

ſ

C

Re

E

The organizational structure of DISPULLAHTAD, its system configuration, and its current various applications will be describes briefly in order to provide a background for the succeeding chapters. A discussion of database management system and a recommendation of the most appropriate DBMS to be implemented comprises the last chapter.

II. INDCNESIAN ARMY LATA COLLECTING AND PROCESSING SERVICE

(<u>DISPULLAHTAD</u>)

A. CRGANIZATION, TASK, AND SYSTEM CONFIGURATION

ś

DISPULLAHTAD is an acronym in the Indonesian language that stands for the Indonesian Army Data Collecting and Processing Service. It was initiated in Fiscal Year 1973/1974 and formally organized in Fiscal Year 1975/1976 [Ref. 2]. DISPULLAHTAD is located in the Indonesian Army Headquarter - Jakarta, the capital city of the Republic of Indonesia.

The DISPULLAHTAD's main task is to provide all information processed electronically for all organizational elements of the Army requiring the information [Ref. 2]. In order to be able to accomplish this task, DISPULLAHTAD is equipped with several computer configurations. As of 1984, these include an IBM System 4341, an IBM System 370, several IBM System 3740s, and several TRS-80s.

In the lower organizational level such as Military Area Commands (KODAM), Army Finance Service (JANKUAD), Army Administrative and Personnel Service (JANMINPERSAD), Army Development and Educational Command (KOBANGDIKLAT), etc., each has its own Data Processing Service and it is called as PULLAHTA KOTAMA/LAKPUS or PULLAHTA for short. A simplifiel organizational structure of the Indonesian Army is presented as figure 2.1 and the position of both DISPULLAHTAD and FULLAHTAS can be clearly visualized.

Each PULLAHTA has two roles: first to process and provide all pertinent information requested by the organization to which it is attached, and secondly, to provide data



ľ

5

.

G

Ó

•

Figure 2.1 Indonesian Army Organizational Structure (Simplified Chart).

entry for all applications centrally processed by DISPUILAHTAD. In order to accomplish these objectives, each FULLAHTA is also equipped with some hardware as shown below.

1. FULLAHTAS inside Java Island

FULLAHTAS that belong to Military Area Command in Java island are equipped with an IBM System 4331 connected to the IEM System 4341 at DISPULLAHTAD in "online" mode via dedicated public telephone lines. This is the first stage in networking all PULLAHTAS throughout the country. The data interchange is done electronically through these dedicated lines.

2. FULLAHTAS outside Java Island

PULLAHTAS sited outside Java island are equipped with an IBM System 3740 and currently work in "off line" mode. Eventually they will be connected to DISFULLAHTAD via dedicated public telephone lines. The data interchange is done manually using floppy-disks transported via airline and it requires one to two days for the data to reach its destination.

B. APPLICATIONS

E

All applications done by DISPULLAHTAD are in order to fulfill its task of electronically providing information needed by the Indonesian Army. These applications are separated into three kinds of Management Information Systems [Ref. 3]:

1. Administration Management Informatic System

This category includes applications in finance, logistics, personnel, and all applications pertinent to development, education, and corps/specialty.

2. <u>Military Management Information System</u>

The applications of intelligence and security, territorial, communication and electronics, and organization, operation, and training are included in this category.

3. <u>Flanning and Controlling Management Information</u> <u>System</u>

Three kinds of applications are included in this category: planning and budgeting, auditing and control, and command, control, and communication.

C. DATA REDUNDANCY FROBLENS

There are many possible data redundancies within those applications. For instance consider data about name, rank, SSN, corps, occupation, etc., that belong to an individual assigned as Intelligence Officer in a Territorial Unit. His data will appear in at least four different files: personnel file, payroll file, intelligence file, and territorial file. If, for instance, there is a change to just one of those data elements, redundant effort is required to update all those four files and a high level of data inconsistency may result.

It happens many times that top level management detects data inconsistency in two different reports generated by DISPUILAHTAD (e.g. the total number of personnel appears differently in the personnel report and the payroll report). This has become very annoying, and tremendously reduces credibility in the computer system. In this regard, an effort should be made to eliminate the problem and one way of doing that is by designing and implementing a data dictionary system (DDS) in concert with a DBMS.

.

D. STAGED DEVELOPMENT APPROACH

The design of any information system is the most difficult and critical step. It should be done with great care and full awareness. As suggested by Sprague and Carlson [Ref. 4], there are three different approaches in initiating an information system:

• Quick-Hit approach.

1

Í F

This approach should be done if there is no clarification whether such an information system is needed or not, but there is a recognized high payoff for initiating the system. This approach requires developing the system in the most beneficial area, capturing the benefits, and then considering what to dc next.

• Staged Development approach.

This approach is done by developing the system in the most beneficial area as in the quick-hit approach, but with some advanced and clear planning. Therefore, part of the effort in developing the first system can be reused in developing the second. This approach is very appropriate for initiating an information system clearly supported by top management, but with limited available resources.

• Complete System approach.

This approach requires the longest development time and highest development costs before any benefits are attained. Before building any part of the system, a full-service system generator and the organizational structure for managing it must be developed first. In this regard, this approach represents the most risky option.

For DISPULLAHTAD, it is anticipated that top management will strongly support the implementation of a DBMS, therefore the Quick-Hit approach is not necessary. On the other hand, limited computer resources makes a Complete System approach infeasible, too. Hence, the most appropriate approach is the Staged Development approach. In the Staged Development approach, identifying the functional area where there is a reason to expect the highest pay-off in starting the project is a crucial thing. Using the highest volume of transaction as the criteria, and by evaluating the transactional data gathered between April 1983 and December 1983

lonth E	Pplic.	Applic.	Armament- Transp.	Applic.	Reports	Applic.	UTREES
pr, 83	24,296	40,772	3,677	648	37,347	0	13, 74 1
lay, 83	123,527	36,969	2,852	7, 446	81,253	0	6,281
un, 83	84,586	36,913	2,902	3, 736	73,672	0	4, 291
1 u1, 83	22,045	28, 782	3,542	0	23,176	0	1,815
.ug, 83	27,560	31,107	10,475	3,779	43,418	10,408	1, 89 3
ie F , 83	32,080	38,978	061 ° E	0	24,713	1,080	0
ct, £3	21,373	35, 294	6,307	1,874	50,260	0	178
op, 83	34,202	37, 983	6,473	13, 177	30,427	0	æ
ec, 83	24,778	39,952	066 " †	101,101	18,546	0	3, 049
otal	394,447	326, 750	44,408	31,761	382,812	11,488	31,256
ercentage	32-25	26.72	3.63	2.60	31.30	h 6 •	2.56

•

(7

 \mathbf{O}

(<u>_</u>

5.

O.

.

.

Transactions by Application at DISPULLAHTAD. Pigure 2.2

• •

• •

•

.

.

(see Figure 2.2), there are three applications having high transactional volume: personnel, payroll, and finance report applications. Only the personnel and payroll applications have a master file which is maintained and used continucusly. Besides that, these two applications are the most crucial in maintaining personnel morale and the most often used in relation to the personnel management task.

Based on these evaluations, data used by the personnel and payroll applications will be the first database to be implemented by DISFULLAHTAD. These applications include personnel, payroll, intelligence personnel, and territorial personnel. It is also implied that the discussion on the DDS and CBMS will be limited to those applications.

The plan for this staged development approach are:

1. Initial design of DD covering personnel, payroll, intelligence personnel, and territorial personnel applications.

2. Implementation of personnel and payroll database.

3. Design of DD for all data used by the rest of applications excluded at the first stage.

4. Implementation of other databases.

Ń

III. <u>DATA DICTIONARY</u>

A. GENERAL

(î

5

The revolutionary change in computer technology has created another challenge on how to organize and manage the very large-scale databases made possible by the combination of database management systems (DBMS) and powerful new hardware systems. The need to control the enterprise's data becomes critical due to the proliferation of microcomputers that trigger more and more applications which in turn creates redundancy and data inconsistency problems. At the same time, the number of microcomputer users demanding direct access to the enterprise data is also increasing. This direct access to large and complex databases again creates a problem of how to "coordinate" and control these complex information structures.

Data redundancy, data inconsistency, and the need to control the enterprise's data lead to the design and implementation of database systems. The database environment itself assumes an architectural plan designed to minimize redundancy and to emphasize accessibility. It assumes logical and physical structures aimed at separate objec-It also assumes that individual file may serve many tives. different applications. All of this is far too much complexity to be managed without precise and up-to-date documentation and control. The data dictionary is designed to define all appropriate aspects of the enterprise's data, so that it can be used as a tool to control and manage the database system no matter how great its size and how complex its structures.

B. INFORMATION RESOURCE MANAGEMENT (IRM)

ń

(

The concept of IRM is that information is a vital enterprise asset that should be invested in, and used like other resources [Ref. 5].

IRM is the task of managing information resources such as data, processes, users, software, and hardware in an integrated and coordinated manner. IRM includes all management aspects of the information-related operations of an organization, such as policy formulation, resource allocation, implementation, and control.

A definition of IFM was formulated at a workshop on Data Dictionary Systems and Information Resource Management sponsored by the Association for Computing Machinery and the National Bureau of Standards in 1980:

Information Resource Management is whatever policy, action, or procedure concerning information (both automated and non-automated) which management establishes that serve the overall current and future needs of the enterprise. Such policies, etc., would include considerations of availability, timeliness, accuracy, integrity, privacy, security, auditability, ownership, use, and cost-effectiveress.

This definition of IRM was chosen to emphasize the enterprise-wide nature of planning and execution of information policies, actions, and procedures in order that lata can be treated as a true resource. It also reflects the primary shift of data processing uses from processingcentered design methodologies to data-centered methodologies.

1. Data Dictionary as the Tool of IRM

Cne of the problems encountered in IPM is the vast amount of data about information resources required to managed the enterprise data, together with the very complex and numerous relationships existing between them. This is

precisely the sort of task that a Data Dictionary System can be made to do, provided that it has been conditioned to know how to deal not just with data entities or process entities, but with the entire range of information resource entities.

2. <u>Lata Dictionary</u>

A data dictionary is a collection of meta-data (data about the enterprise data) that could consist of: the name of data (including its synumyms and/or homonyms), the location of data, a description of the meaning of data, the relation between data, how the data is used, who is responsible for the data, the source of the data, etc., in short, a store of all the appropriate information about the data.

Recently, there has been a trend towards using data dictionary to include the following functions:

a. Definition of other data constructs such as records and files.

b. Definition of processes such as programs or manual processes.

c. Definition of data users whether individuals or organizational entities.

Along with these definitions, the data dictionary also began to be used to document the cross-references between them and to record their usage and organizational responsibilities.

3. <u>Lata Dictionary System (DDS)</u>

A Data Dictionary System is a combination of software and procedures that aid an enterprise in setting up and maintaining its complex structure of data resources. The software itself may be produced in-house or acquired from software vendors. For the following three reasons, it is often better to purchase instead of building it in-house:

a. Design and Implementation

The task of designing and implementing a DDS, even one of modest functionality, is definitely a nontrivial one. There exists good potential that the magnitude of the task will be underestimated and that greater resources will be needed than those originally estimated.

L. Gaining a Success

1

C

If the use of DDS is to be at all successful, the software itself must conform to high standards of quality assurance. The use of the same software at many other installations, as is the case with a commercial package, aids in the early discovery of possible software errors and their corrections.

c. Technology Progress

There are good reasons for assuming that DDS technology will continue to progress and that substantial enhancements will significantly increase the usefulness and value of the DDS, and it will be difficult for an in-house system to keep pace.

There are several DDS commercially available now, such as DB/DC Data Dictionary of IBM, DATAMANAGER by MSP Inc., Integrated Data Dictionary (IDD) by Cullinet Software Inc., DATADICTIONARY by Applied Data Research Inc., Extended Data Dictionary (XDD) by Intell Systems Corp., JCC TEN by University Computing Company, and Data Control System (DCS) by Cincom Systems Inc.

In the following section, features that one could expect to find on those available DDS will be discussed. It must be pointed out that none of the available DDSs mentioned above will necessary have all explained features, and there is no such implication that all of these features are required in all DDS applications.

C. FEATURES OF A DATA DICTIONARY

C

There are several issues will be discussed here, such as Implementation and Architectural, Data Dictionary (DD) and DD Schema, Extensibility Facilities, Status Facilities, Dictionary Commands, Eridge Facilities, and Data Dictionary System Security.

1. Implementation and Architectural Issues

a. The Relationship Between DDS and DBMS

The primary purpose of a DBMS is to manage data, whereas the primary purpose of a DDS is to manage meta-data. Therefore, it is clear that there is a very little overlap between these two, in fact they are complementary; both functions are required for proper management of information resources.

Some of the functions a DDS will perform are in support cf one or more DBMSs. This is to be expected as the DDS will manage all meta-data, including meta-data where the actual instances of data are stored in a database, which in turn is being managed by a DBMS. An element required for this latter function, the DBMS's management of data in databases, is the knowledge on the part of the DBMS of certain meta-data of the databases which are required by the DBMS in crier for it to its processing. This meta-data is commonly referred to as the DBMS-Directory, and it should be clear that this potentially is one area of overlap between the DES and DBMS. In this sense, it is preferable to design a DDS prior to the DEMS implementation rather than to build a DDS that has to be fitted toward an existing DBMS. This reason together with an existing method of implementing a CDS as one of the CEMS's applications may explain why in this thesis designing the dictionary is done prior to the design of the databases.

t. The Method of DDS Implementation

It is preferable to design a data dictionary prior to the implementation of the DBMS. On the other hand, the implementation of that designed data dictionary as a complete DDS is a good candidate to be one of the DEMS applications, and indeed a number of existing DDS are implemented in this manner. But this is not a single option, theimplementation of DDS can be either :

• DBMS-dependent system.

C

(<u>2</u>

This is a DDS that uses a DBMS in its implementation.

• Free-standing system.

A DDS that doesn't use a DBMS in its implementation is considered to be included in this category.

There is no ultimate answer as to whether a free-standing or DBMS-dependent DDS is the best. There are both gros and cons to this cuestion, and these depend on the enterprise's specialized circumstances, such as whether the enterprise implements a DBMS or not, and whether it uses multiple DPMSs or a single DBMS for its databases. Enterprise (s) that have no intention to implement a DEMS but DDS will give a favor toward a free-standing system. On the cther hand, the enterprise(s) having multiple DBMS шaу implement a DBMS-dependent system and choose one cf its DBMSs to implement it; or, they may implement a freestanding system in order to provide more flexible and fair control.

Other considerations include the DDS security and a view that the scope of DDS usage is substantially broader than the DBMS environment. With a DBMS-dependent system, personnel familiar with the use of the DBMS may find it easier to break the DDS security than would be the case with a free-standing system.

For the sake of completeness, there is another concept of DDS implementation method referred to as integrated DDS. This method offers an elimination of overlap between data dictionary and DBMS-Directory by combining these features into one. The advantage gained by combining these two features is that redundancy of storing the meta-data is eliminated.

c. Active and Passive DDS

ſ

In the processes that require meta-data for its execution, there should be a command or series of commands, representing some DDS functionality that produces the required meta-data. This functionality is called dictionary interface, and there are two kinds of such interface: active and passive interface.

An active interface means that all processes that require meta-data will use the most current meta-data in the data dictionary. Similarly, all processes which in the course of their execution generate meta-data are required to store the generated meta-data in the data dictionary.

On the other hand, the passive interface will have all that an active interface has to do as the option. The other option for all processes that require meta-data in its execution are either retrieve it from data dictionary or some other locations; or if the process already contains the meta-data, there exist an option for the system to check whether or not this meta-data is the most current version in the data dictionary. In the case of generating a meta-data, the process also has an option to store or not to store the generated meta-data.

Therefore, two conclusions can be drawn about the dictionary interface :

(1) A DIS may have some interfaces which are active and others which are passive

(2) The fact that an interface is active is a property not only of the DDS, but also of the overall system of which the DDS is a part.

2. Data Dictionary and Data Dictionary Schema

Data Dictionary denotes the organized and structured collection of meta-data which comprises the contents of the DDS. The data dictionary schema denotes the logical structure of the data dictionary, in a manner analogous to the use of same term in the context of a DBMS.

The structural characteristics of data dictionary and the contents of data dictionary schema will determine what kinds of meta-data can be stored in the data dictionary and what kinds of relationships can be established between them. Some systems have extensibility facilities whereby an installation can customize the data dictionary schema.

The schema is described in logical terms in order to gain a clearer insight about what kinds of meta-data are supported by the DDS. This logical description will, of course, be quite different from the manner in which these structures are actually implemented in a specific system. This description should be made independent of any implementation.

A data dictionary has a conceptual similarity to the Entity-Relationship-Attribute model. The basic unit in the data dictionary is a Dictionary Entity or Entity for short. Entities represent real world objects or things about which certain information exists in the lata dictionary. And the information about entities themself, exists in the form of Attributes which generally denote the qualities or quantities of properties of the entities. Finally, data dictionary also contain information about Relationships

between entities, and relationships may have attributes assigned to them.

1

The term Entity-type is applied to some entities that have similarities among them. For example, if a set of files is described in the data dictionary, each file will be represented by a distinct entity. It then becomes useful to establish an Entity-type called File in the data dictionary and to say that all such entities representing files have the entity-type File. Attributes of entities of the same will exhibit a certain degree of similarity. type Entity-type File will likely have an attribute of what kind of access method used, and maybe another attribute showing the blocking factor used. These both access method anl blocking factor are then called as an Attribute-type which is associated with the entity-type File. Beside the entitytype File, there will be an entity-type Record. The information would exist in the data dictionary explaining which types cf records are included in a given file. All such relationships between these file entities and their associated record entities, then be called as a Relationship-type. In conclusion, the data dictionary schema would be viewed as containing all existing entity-types, relationship-types, and attribute-types. Any one of these three types may also te referred to as a schema descriptor.

Every entity in the data dictionary has a primary name which, depending on the particular DDS, will be unique either in the dictionary or within the entity-type to which the entity belongs. Some systems may facilitate duplicate user-supplied names by assigning them distinct sequential numbers. In this case, the concatenation of user-supplied name and sequential number constitutes the unique dictionary name. The allowable length of the primary name should be sufficiently large enough to convey the meaning of an entity in its primary name. It is common that at least some entities will also be known by other names. Such alternate names are called **aliases** or **synonyms**, and the most important things are the capability of the DDS for tracking them and allowing access to the data dictionary via these alternate names. Sometimes it is convenient if non-unique synonyms are allowed. To fulfill this requirement, DDSs have facilities for tracking synonyms either as attributes of the respective entities, or as separate entities related to the primary entity. Therefore, it is important that the system should be able to recognize the context in which the synonym

The attributes can be differentiated into some attribute-types, among which are: Description = it consists of an English language statement describing the meaning of the entity. Classification keywords = these are attached to the entities which then can be used for selective retrieval of these entities. Audit-attributes = these are attributes, generated by the DDS, indicating the identification of the person who created, the date of creation, the identification of last person who modified, the date of last modification, and the total numbers of modification, all for each entity.

The entity itself can be conveniently separated into three entity-types: Lata, Process, and Usage entity-types.

a. Data Entity-types

ľ

1

The most common of this type, listed with typical attribute-types and relationship-types are:

(1) <u>Item/Data Element</u>. In some systems the lowest entity-type is Item, which is considered to be the atomic unit. In other systems, the lowest entity may be Data Element, which in its turn it may contain other Data Elements. This is usually specified by contains clause, which expresses the relationship.

Commonly provided attribute-types relate to the physical characteristics of the Item/Element, including distinctions between Source, Target, Internal representations, and the validation criteria that may be required for the real-world instances of the Item/Element.

(2) <u>Group/Record</u>. Systems that recognize the Item as an entity-type will contain Group as a separate entity-type, whereas systems that have Data Element as an entity-type do not have an entity-type for Groups. Record is logically the same as a Group, therefore separate entitytypes for both of them may or may not exist. A relationship-type is provided to express the structure of the Group/Record. Commonly available attribute-types relate to the manner in which the constituent elements are aligned, and other physical characteristics of the entity.

E

(3) <u>File</u>. Relationship-types are provided to express the structure of the file. Attribute-types relate to the access method used, blocking and labelling information, etc.

(4) <u>DBMS-related Entity-types</u>. The entitytypes that exist are dependent on the specific DBMS for which the DDS provides support services. In all cases, the entity-types equate to the various data descriptions used by the DBMS, such as Schema, Subschema, Database Directory, etc. Relationship-types and attribute-types are provided to allow the DDS to express the structure of these entities.

(5) <u>Other Data Entity-types</u>. Some DDSs offer Report, Screen, and Form entity-types. In each case, relationship-types are provided that allow the contents of such entities to be specified in terms of the constituent elements.

Process Entity-types

Z

5

There are two most common Process Entity-types. They are Program/Module and System/Subsystem as will be discussed below.

(1) Program/Module. This entity-type represents information about a collection of executable code. Typical attribute-types are the language of the source code, the size, and the characteristics under which it operates. Relationship-types are provided to other Programs/Modules, as well as the data, i.e. databases, files, and elements, or. which it operates. Generally speaking, different relationship-types are provided for input, output, and processing-in-place.

(2) <u>System/Subsystem</u>. This entity-type describes a collection of programs and/or Modules associated with a major function of the enterprise. Relationship-types are provided to associate a System with Subsystems, as well as the constituent Programs.

c. Usage Entity-types

Users and their organizational environment, and the data communication environment can be thought of as Usage (or External) Entity-types. They are not directly components of a system, such as data and processes, but nevertheless play an important role in its operation.

The User and organizational component entitytypes may have relationship-types that allow users to be associated with organizational components. These components themselves, and selected relationship-types that associate users or components with data and process entities may describe the responsibilities assigned to those users.

The example of Data Communication environment entity-types are Terminals, Messages, and other entity-types

that describe the communication networks. Their may provide the associations of relationship-types such entities with other usage entities, i.e. a given terminal is assigned to a certain set of people or a certain crganizational unit. Or it may provide such associations with data and process entities, i.e. a given terminal is authorized to execute only a given set of transaction programs, or to access only certain Files or Databases. It will be appropriate to note here, that the role of the DDS in these matters is strictly a repository for documentation, and that DDS by itself cannot be expected to enforce such the restrictions and limitations. In order for DDS to be used for enforcement, appropriate "active" interfaces would have to exist to assure that the restrictions and controls which are documented in the data dictionary are always invoked at execution time. It will, on the other hand, create more complexity and much overhead.

3. Extensibility Facilities

The concept of extensibility facilities is to allow an installation to modify the system-standard schema as delivered by the DDS vendor. Any new schema descriptor created through the use of extensibility facilities will be referred to as an **extensibility descriptor**.

Extensibility facilities are extremely powerful, and their usage should be done with great care because extensions to the system-standard schema, once they are used, can only be undone with some difficulties. Additionally, changes to the system may create confusion among the users of the DDS as well, and decrease their confidence in the system. Due to these reasons, it is recommended that their usage should be restricted to the Dictionary Administrator, the person who is responsible for DDS function, i.e., the recording of all meta-informations and meta-data and its

maintenance through the use of the DDS, along with making its facilities available to the users of the system.

There are three kinds of such facilities that exist in current DDSs :

a. Entity-type extensibility: the ability to add new entity-types to the dictionary.

b. Attribute-type extensibility: new attribute-types for either entity-types or relationship-types can be declared using this facility.

c. Relationship-types extensibility: this facility allows the installation to declare new relationship-types.

4. <u>Status Facilities</u>

C

ſ

1

These facilities allow the DDS to be used in a System Life Cycle environment where, for instance, a certain entity may both part of a production system and a new test system. Due to its intended usage, it is preferable to maintain the same name for the entity in different stages. Therefore, a facility is required that will allow two distinct dictionary entities to have the same name, yet different attributes and relationships. In some systems, such entities can be distinguished by assigning different version numbers to them. In systems having a status facility, it can be accomplished either by:

a. Appending the entity-status to the entity-name which provides the uniqueness of the name.

t. Logically partitioning the dictionary into separate databases for different statuses, and requiring uniqueness of the name only within each partition.

5. Dictionary Commands

A DDS may have one or more interfaces that allow a user to interact with the dictionary. Such an interface may in the form of :

• A command language.

 $(\mathbf{C}$

• A screen-oriented interface.

• A fixed format batch data entity facility.

• A programmatic interface that allows user written application programs to access the dictionary.

A screen-oriented interface is more user-friendly compared to the others. It results in higher utilization of computer system resources, but makes the DDS available to a larger class of users. Another benefit may be that the error rate is substantially reduced.

The dictionary commands may be differentiated into eight categories on the basis of their functionality:

a. Dictionary Maintenance Commands: this enables an installation to create and maintain its data dictionary.

h. Reports and Queries Commands: an installation may generate reports using meta-data contained within the data dictionary using these commands.

c. Data Structure Interface Commands: enable other systems to use the descriptions of data structures contained within the dictionary.

d. Extensibility Commands: vehicle to exploit the extensibility facilities.

 ϵ . Status-related Commands: the ability to distinguish entities in different stages of the life cycle.

f. Security Commands: used to allow security declarations to be assigned to the dictionary.

g. Dictionary Processing Control Commands: used to control the dictionary process such as logon-logoff, processing defaults, etc.

h. Dictionary Administrator Commands: exclusive commands especially designed to be used by the dictionary administrator.

6. Bridge Facilities

Other facilities of a DDS exist in the form of bridges or interfaces to other systems. The contents of the dictionary may be made available as part of the processing functions of that system. In each case, these systems provide tools whose functionality is outside the DDS but which require data about entities which can be expected to exist in the dictionary. By accessing a dictionary and extracting the required information, the disalvantages of having to store and maintain redundant data is eliminated.

Those such interfaces are :

a. Report and Query System: the ability to surport various kinds of reports and queries.

b. Validation Criteria: support other systems by providing a module which performs the specified validation and which can be inserted into a program.

c. Database Design: the ability to provide basic data needed by automatic database designers.

d. Test Data Generation: support test data generation by providing descriptions of the structures and formats of the files and databases.

7. <u>IDS Security</u>

As mentioned before, the term DDS security is applied here to denote the security of the DDS itself. Entities in the dictionary may have attributes describing access characteristics to the real-world instances of these entities, but this data is entirely informational in nature and cannot be enforced by the DDS, since the system is not part of the loop in the execution of programs against the "real data". On the other hand, unauthorized access to the DDS can be eliminated by applying such security procedures, e.g., the assignment of passwords, or the inclusion of

security levels which control the various kinds of access to dictionary entities. To gain more integrity and reliablity, the security of the IIS must be considered to be related to the security of the entire computer system. The level of security existing in the computer system is influenced by the security of the fasic systems software and the physical security of the installation, as well as the procedures used by the personnel of the installation. These latter are often notably lax, and it is not at all unusual to observe cases where passwords are not kept confidential and may, indeed, openly be shared with unauthorized people.

D. COST/BENEFIT ANALYSIS FOR DDS

As applied to any system acquisition, Cost/Benefit analysis should be done prior to and in order to get a justification for DDS acquisition, implementation, and usage. The following list of costs and savings represent tangible items that can be used in assessing the costs and penefits for an economic study of the feasibility of implementing a DDS.

1. Costs

There are eight possible costs which may be considered:

a. Acquisition cost is the accumulation of lease or purchase cost and the maintenance cost of the system.

b. Data Administration staff cost is self explanatory.

c. Hardware Cost is the sum of storage device cost and CPU time cost.

d. Start-up cost is the total cost of training data administrator staff and all activities such as developing a comprehensive plan (see Table I as an example) that should be done in any data dictionary implementation.


[

Ń

C

e. Data collection cost is a function of the number of entities, attributes, and relationships which are to be put in the dictionary.

f. Maintenace cost will depend on the degree of changes to the application system or systems controlled by the DIS.

g. Application system change cost is a cost pertinent to any change to the application systems due to the implementation of the dictionary for reasons of efficiency, integrity, and maintainability.

h. User education cost is the cost for training reople involved in data dictionary usage in addition to the data administrator staff.

2. Factors for Estimating Savings and Benefits

ſ

There are four factors which can be used as an aid in estimating the quantification of savings. The greater the degree to which these four factors are held to apply to the enterprise and its operations, the more the high end of estimated saving and benefits can be expected. Those four factors are:

> a. The Maturity of an Information Processing Environment

This is a major factor in the benefits that can be attained with a DDS. Increased maturity will help substantially in the integration of dictionary facilities into the operations of the enterprise.

t. The Complexity of the Environment

The number of data elements, files, databases, and programs can be used to measure how complex the information processing environment is. Problems caused by complexity tend to worsen geometrically with the number of such elements, files, databases, and programs. The value of a DDS will be greater as complexity increases.

c. The Degree of Data Sharing

It should be common practice that data elements are shared by different programs, where some of these are from different systems. An important issue is that changes in one part of the system tend to have effects in many other parts of the system. Failure to compensate for such changes can cause production failures and unanticipated costs. Tracking the effect of these changes is a valuable feature of a DDS. It is made possible by evaluating the attributes and relationships of changed entities as the basis for further tracking.

d. Personnel Turnovers

The personnel associated with information processing systems is either data processing organization personnel or user's organization personnel used to deal with the information processing system. In this regard, a DDS offers two advantages: <u>First</u>, information which otherwise might be stored in the minds of individuals and which may be lost to the organization with the loss of the individual is now placed in the DDS. <u>Secondly</u>, the learning curve for new personnel is steeper than it would be without the use of a DDS.

3. <u>Savings</u> and <u>Benefits</u>

٦

There are five areas in which savings and benerits may be expected. The four factors mentioned in the previous section can be used as aids in predicting specific monetary savings in each of these following areas:

a. System design and development: the prime advantage of the DDS is in its use for better communication between users and implementors. This results in fewer changes or iterations and consequently faster programming because the specification is better documented and understood by all parties.

t. System maintenance: better and more complete documentation in the dictionary, the ability to analyze the effect of proposed changes, and the improved communication between users and maintenance programmers on proposed changes or corrections.

c. Data redundancy: reduction of unplanned data redundancy will result in an improved system which has greater integrity and better operability as well as potentially decreased requirements for random storage devices.

d. Database creation: may take advantage of the descriptions contained within the DDS to reduce iterations in the design process and faster concurrence by all parties on the contents of a database.

•

l

1

e. Improved Communication: self explanatory.

IV. INITIAL DATA CICTIONARY DESIGN FOR DISPULLAHIAD

A. GENEFAL

(<

As mentioned before, the initial design of Data Dictionary will be limited to four application areas: personnel, payroll, intelligence personnel, and territorial personnel. This design is a first step in the Stage Development approach used, therefore it may be expanded in the future.

E. DATA DICTIONARY SCHEMA/SUBSCHEMA

In a manner analogous to the context of a DBMS, Data Dictionary Schema denctes the logical structure of the data dictionary (DD). In this regard, then, the term Subschema will dencte a subset of the schema to be seen by a given application (process) or user [Ref. 6], and it is compatible with application views of a database [Ref. 7].

1. <u>Data Dictionary Schema</u>

The structural characteristics of the DD and the contents of the DD schema are important aspects of the usage of a DDS, since by evaluating these, users may know what kinds of meta-data and relationships between them exist within the DD. As suggested by Lefkovits et al [Ref. 8], the structural characteristics of a DD may be described in logical terms in order to gain a clearer insight of what kinds of meta-data are supported by the DDS. The logical structure of a typical DD drawn by Allen et al [Ref. 5] seem to be appropriate for DISPULLAHTAD's DD with only minor changes, and it is presented as figure 4.1.



ſ

5

e

ß

0

.

Figure 4.1 Logical Structure of DISPULLAHTAD'S DD (adapted from Allen et al [Ref. 5]).

There are three kinds of entities (from the left to the right) within the logical DD in figure 4.1 :

a. Data Entities

These consist of database, subschema, relationship, file, group of elements, and data element entities. Wherein each record may consists of some elements, but may or may not have group of elements in it. File and/or record entity is the subject of a process entity, while data element entity is the subject of a transaction or a report entity.

그는 그 씨는 옷에 가지가 봐야 한다. 것이 가지?

t. Process Entities

3

0

A process entity may be an application program, a program module, or a system/subsystem that typically generates a report or does a transaction involving either data element or a group of data elements.

c. Usage Entities

Included in this category are user and their organizational environment (such as processors and terminals), and data communication environment (such as communication network, communication nodes, messages, etc.).

2. Lata Dictionary Subschemas

Since there are four applications included in this initial design, there might be four subschemas accordingly. Due to the relatively small amount of meta-data that will be stored in the initial dictionary, however, and in order to reduce complexity, it is better not to apply subschemas at this point. Later, if the dictionary has grown subschemas at tially, and many applications have been added, the subschemas may be applied in order to improve efficiency and security.

C. DISPULLAHTAD'S DATA DICTIONARY

The design of this DD is based on current applications for which the DBMS has not been implemented yet. The following tables present the lists of Data, Process, and Usage entities and one or two of the actual instances of DISPUILAHTAD's data dictionary.

1. Entities

a. Data Entities

Table II summarizes data entities abstracted from the personnel application [Ref. 9 and Ref. 10]. Table III presents data entities for the payroll application [Pef. 11 and Pef. 12]. Lata entities used in the intelligence personnel application is presented as Table IV [Ref. 13 and Pef. 14]. Finally, data entities from the territorial personnel application are shown in Table V [Ref. 15].

1. Process Entities

Process entities belonging to the four applications are presented as Table VI. This table lists only routine processes, not irregular processes, since the latter ones are not yet standardized (they are done as requested).

c. Usage Entities

Table VII presents user entities for the four applications.

S.S.N. Plus selfcheck Personnel Name Rank Corps/Specialty Unif of Command Administrative Unit Dogistics Unit Personnel Status Fersonnel Status Fersonnel Status Fersonnel Status Date of Cutegory Date of Birth Date of Birth Barital Status Marital Status Marital Status Corder From Order From Or Service Main Data Record Personnel File .ption Ish) Descril (Engl Entities Nomor Prinlak Prinlak Dari Jabatan Penyangkatan dalam Jab. Pendidikan Umum Terakhir Le Status Rumah Tinggal Rode Rumah Dinas Nomor Ukuran Tepi Nomor Ukuran Tepi Tingyi Badan Tingyi Badan Penat Badan (Kg) Nomor Personil Nama Personil Fangkat/Golongan Corps/Cabang-Fungsi Satminkal Kotama dan Satker Status Personil Kateyori TMT Rateyori TMT ABRI/Pegawai Tanggal Lahir Jenis Kelamin Status Kawin TMT Prinlak Jabatan Data Application H File Personil Description (Indonesian TABLE Data Pokok Personnel (1) ERSFIL NOPRIN FRDARI STJAARI STJAARI STJAARI STARUAR FRORDIN FRORDIN FROGATIO FROGATIO FROGATIO FROGATIO Entity Name NOPERS FORNDERS FORNDERS FONTERS FONTERS SCONTERS SCONTE DAPOK 1 Entity Type Record File Α. NO. н. 1

Ċ

1

۰.,

Tribe Esychotest Classification Fhysical Fitness Test Payroll Number Number of Sons Number of Sons Number of Daughters Unit Giving Ration Unit of Administration Unit of Administration Unit of Payroll Eighest Mil. Fducation Health Classification Physical Ability Number of Date of Gccupation Cccupation Code Rank at the Beginning o the Occupation Occupational Order Numb Order From Occupation Status Job Tmplementation Hair Flood Medals Focal Language Foreign Language Religion Date of Promotion as an Officer Fiction Syc Years Fact Service Years Occupation Record (cont'd.) Masa Dinas Pa Fiktif Masa Kerja Yang Diperhitungkan Sukubangsa / Keturunan Klasifikasi Psychotest Kesamaptaan Nomer DPP Jumlah Anak Laki Jumlah Anak Perempuan Kesatuan Kaporlap Kesatuan Posir Personil Kesatuan Posir Personil Kesatuan Posir Personil Kesatuan Posir Personil Kesatuan Pingasan Nomor Jubar Subjubar Dik Militer Tertingyi Kemampuan Fisik Entities q Jab. setiaa n Dalam J Jabatan Jabatan Ð TMT Jabatan Kode Jabatan Pangkat Saat Itu ы Rambut Golongan Darah Satya Lencana K Bahasa Daerah Bahasa Asing Agama TMT Perwira Application Data I Nomor Skep Jal SKEP Dari Penyanykatan l Pelaksanaan Jo Table Jabatan Personnel SUKU KLASIF SERAPTA NDPP ANAKL ANAKL SAATCB VANNNO FUANFIES FUANFIES FUANFIES JABATAN NOKJAB KDRJAB STJAP IAKJAB TGJAB KOJAB PKTJAB RAMBUT CERTAN SETTA FAHDA FAHDA AGAMA TMTPA MDPFIK MKFIK Element Flement ment Pecord Eler 337 -c1m JUION щ

-

rler Record ec. Date of Occupation Rank at the Beginniny of Occupation Order Number Order From Occupation, Description Occupation Status Implementation of sned Date of Education Education Code Place of Education Result of Education Duration of Education Number of Student Graduated as Number p: 0 Rank Code Date of the Rank Number of Promotion C Promotion Order From Kind of Promotion History Rank/Cateyory Record Date of Operation Kind of Operation Name of Operation Flace of Operation Cperation Order Crder From Fini Military Education **Operational** Record **Occupational** Entities (cont'd.) an Tanggal Mulai Pendidikar Kode Pendidikan Tempat Pendidikan Hasil Pendidikan Lama Pendidikan Jumlah Siswa Nomor Urut Lulus Jab. Kode Pangkat TMT Pangkat Nomor Skep Pangkat Skep Pangkat Dari Macam Kenaikan Pangkat vi Opera. s. Nomor Skep Jabatan SKEP Dari Jabatan Pengangkatan Dalam Ja Pelaksanaan Jabatan ð Operi Pendidikan Militer Itu angkat/Golongan Tanygal Mulai Op Macam Operasi Nama Operasi Lokasi Operasi Sprin Operasi Sprin Dari Tanggal Selesai (Application Data Uraian Jabatan H TMT Jabatan Pangkat Saat rable Operasi a, Personnel NOKJABU KDRJABU JABATU STJABU STJABU IAKTABU TGJABU FKTJABU E RDIKNIL TGOPS MAOPS NAMOPS KEPOPS KEPSAT SELCPS FKAT TGKAT KEPKAT KDRKAT MKKAT TGDIK KODIK HSLDIK HSLDIK JANDIK NORUTSIS URAJAB FANGKA CPERAS Element Flement Flement Flement Flement Elegent Flegent Flegent Flegent Flegent Flegent Flegent Element Element Record Fecord Record Record -10 M F M - ~ -antinor -UNTRUN-: : . ш • म्म Q

States and the states of the s

9

1

Q

R

1

Č

			Personnel	Table II Application Data Entities	(cont'd.)
	о ^в .	Element Element	KORBAN TJA SA	Korban Tanda Jasa	Victims Operation Medals
с.		Fecord	KLUARGA	Keluarga	Family Record
	-0103109-00-00	E C C C C C C C C C C C C C C C C C C C	HUBKEL HUBKEL Adamaker Klaintnk Mersta Agamak	Hubunyan Keluarya Tangyal Lahir Keluarga Nama Keluarga Tempat Lahir Keluarga Kelamin Keluarga Pekerjaar / Status Tanggal Berlakunya Status Ajama Keluarga	Family Relationship Date of Birth Name Place of Birth Sex Occupation Spate of Occupation Religion
н.		Record	KETRKEL	Keterangan Keluarga	Detail of Family Record
	-010-3	Element Element Element	LIKKEL TGKAW IOKKAW ALAMAT	Pendidikan Keluarga Tanggal Kawin Tempat Kawin Alamat Lengakp	Family Member's Education Date of Marriage Place of Marriage Address
Ι.		Record	ADRES	Alamat Personil Bujanyan	Eachelor Address Record
	-	Element	ALAMAT	Alamat	Personnel Address
ц.		Record	PENGLN	Pengalaman Luarnegeri	Foreign Experience Pecord
	-010-3-00	Element Element Element Element	TGBRK NEGARA MTUGAS UTUGAS TGKEMB	Tanggal Berangkat Kode Negara Tujuan Macam Tugas Uraian Tugas Tanggal Kembali	Date of Departure Code of Foreign Country Kinl of Duty Detail of Duty Date of Return
М		Record	FGARAN	Pelangyaran	Ereach Record
	-0m	Element Element Flement	TGGAR MAGAR TPGAR	Waktu Kejadian Macam Pelanggaran Tempat Kejadian	Late of Breach Kind of Breach Flace of Breach

C

T

Q

Method of Breach Motivation of Breach Result/Effect of Breach CO action Main Punishment Total of Punishment Additional Punishment Late of Punishment Place of Punishment Record Graduation Date as Off. Personnel Name SSN plus Selfcheck Digit **File** Officer Conduct hecord Evaluation Semester Date of Graduation Name of Elucation Title of Education General Education Cfficer Personnel Record Source of Officer Year of the Test Classification Grade Classification Psychotest (cont'd.) Cara Yang Digunakan Motivasi Pelanggarar Akibat Kejadian Tindakan Dan Sat Bukuman Pokok Hukuman Tambahan Waktu Dijatuhkannya Fuk. D Tempat Pelaksanaan Huku. P **Application Data Entities** Fa dan Tahun Selesai/Lulus Nama Pendidikan Gelar/Keterangan Tahun Lulus Sebagai Nama Personil Nomer Personil Tahun Penilaian d Periode ke Nilai Klasifikasi Personil Perwira Pendidikan Umum Table II Sumber Perwira Dapen Perwira Tahun Test Klasifikas Psychotest Personnel PERWIRA САРА МОТТV АМСАГV ТОКОАR ЕЕКРОК НКРОК ТСОНК ТРОНК TALULS NDIKUM GELAR KLAPEN KLAPEN THNPSY KLAPSY RDIKUM SUMBPA TIMNHI **ESYCHO** THNPA KEYPA NOPRS LAPEN RICCE RELECTE RELECTE RELECTE RELECTE RELECTE RECTE RE Element Record Record Record **Fecord** File 10.00 00 00 10 00 E F F -4n. -... -~· н ບໍ່ ~ 25

۰.

•

.

6

46

tities	Payroll File Payroll File Payroll File Main Data Record Finance Officer Unit of Command Finance Officer Disbursing Number Payroll Group Fayroll Status Personnel Namc Parroll Status Personnel Namc Ratk Parroll Status Payroll Discount Date of Strvice Year Main Salary Marital Status Number of Childrén Fayroll Discount Pousing Discount Fousing Discount Pousing Discount Fousing Discount Fousing Discount Fousing Discount Fousing Discount Fayroll Number Selfcheck Digit Selfcheck Digit Sex Code
TABLE III oll Application Data En	Pescription File DPP Data Pokok Ku Kotama Juru Bayar Sub Jubayar Sub Jubayar Sub Jubayar Sub Jubayar Sub Jubayar Status Penghasilan NRP/NBI/ NIP Status Penghasilan NRP/NBI/ NIP Status Penghasilan Nama Pangkat/Golongan Rasa Kerja Gaji Gaji Pokok Status Keluarya Status Recria Gaji Pot Sewa Pumah Kateyori Skorsing Bulan Wisus Status Record Status Record Fangkat Tituler Nomer Tambahan Kode Kelamin
Рауго	E T L E C C C C C C C C C C C C C C C C C C
	0 200000000000000000000000000000000000

l

r

2

(cont'd.)	Description (English)	Transaction Record Finance Command Unit of Payroll Finance Officer Dispursing Number Unit of Command S.S.N. Selfcheck Digit Personnel Name Rank Payroll Number Payroll Number Date of Transaction Date of Transaction Post Transaction Data Post Transaction Code Date of Transaction Post Transaction Data
Table III Application Data Entities	Description (Indonesian)	Laporan Mutasi Ku Kotama Pekas/NA Juru Bayar Sub Jubar Sub Jubar Kesatuan/Satker NRP/NBL/NIP Nomor Tambahan Nomor Tambahan Nomer DPP Jenis Mutasi Tan Mutasi Keadaan Baru Kelompok DPP Macam Perubahan
Payroll	Entity Name	I XNUSSINNAGUHXXX A HAGUAXINNAGUHXXX A AGGUAXINNAGUHXXX A AGGUAXINNAGUHXXA A AGGUAXINNAGUHXXA A AGGUAXINAGUHXXA A AGGUAXINAGUHXXA A A A A A A A A A A A A A A A A A A
	Entity Type	налагалалалалалалала палагалалалалалала палагалалалалала палагалалалала палалалалалала палалалалала палалалал
	No.	あ - つうせららののつううますう

(1

O

Nama Pangkat Pangkat Pangkat Pangkat Nomer Personil Jabatan Satminkal/Kesatuan Satminkal/Kesatuan Satminkal/Kesatuan Satminkal/Kesatuan Satminkal/Kesatuan Satminkal/Kesatuan Socypation Occupation Command Aloamat Rumah/Keluarga Pome add Command Pome add Pome add Command Pome add Po File Rec. ŝ Fers Description (Fuglish) Pers Personnel Application Data Entities Intell. Intell Intel Intel Personil ΔI Personil Description (Indonesian TABLE Record File Intelligence NAMA FANGKAT CORPS CORPS SABATAN SABATAN ALAMAAT ALAMAAT AGAAAAT AGAAAAT AGAAAAT ANAKLK ANAKLK DIKKUM DIKKUM CIKKUL MADIKDN IAMDIKDN IULUSDN TAHUNDN TEMPATDN DIKINLN FILINTEL z FERSINTL TEMPATIN LULUSLN TAHUNLN LIKINDN Entity Name Element Flement Flement Flement Group Element Element Element Entity Type Record Group Ð Fil 5000000 252 . 8 ¥. No.

	ities (cont'd.)	<pre>tescription (English)</pre>	Foreiyn Language Ability Name of Foreign Language Active or Passive Local Languaye Ability Name of Local Language Active or Passive History of Occupations Order Number Date of Occupation Cocupation Rank
Table IV	onnel Application Data Ent	Description (Indonesian)	Kemampuan Bahasa Asing Bahasa Asing Aktif/Pasif Kemampuan Bahasa Daerah Bahasa Daerah Aktif/Pasif Riwayat Jabatan Nomer Surat Keputusan Tangal Tangal MT Jabatan Pangkat
	igence Perso	Fntity Name	EAHASING AKPASING EAHADAER EAHADAER KRDARAH AKRDARAH AKRDARAH NOKEP IAMTUGGAL JABAJAB FANGKATAN FANGKATAN
	Intell	Entity Type	жыты по стана с така в с с с с с с с с с с с с с с с с с с с
		NO.	800

C

6

G

50

ļ

			TABLE V	
		Territorial	Personnel Application Dat	a Entities
No.	Frtity Type	Frtity Name 	Description (Indonesian)	<pre>Description (English) </pre>
	File	FILETERA	File Personil Terr.	Terr. Pers. File
Α.	Fecol 1	FERSTERF	Record Personil Terr	Terr. Pers. Record
-UMANNADARCEUMANRAQODENMAR NNNNNNA EEEEEEEENNANNNN		KKKODER KKKODER KKKODER KKKODER KKKA KAABAAAAA KAABAAAAAAAAA KAAAAAAAAA	Eselon Kolem Kodemil Kode Jabatan Kode Jabatan NRP/NBI Tangyal Lahir Kode Pendidikan Ter Kode Pendidikan Ter Kode Pendidikan Ter Kode Pendidikan Ter Kode Rendaran Dalam Apter Kode Rumah Dalam Apter Kode Rumah Tinggal. Kode Kesanggupan Kerja Jumlah Arak	Cryanization Level Code Military Resort Command Military Resort Command Military District Command Occupation Code Name Fank Corps/Specialty Ses.N. Date of Firth Military Education Code Part. Experiences Course Code Terr. Course Code Date of Accupation Terr. Experiences Date of Resignation Terr. Experiences Date of Resignation Terr. Experiences Date of Resignation Terr. Course Code Terr. Experiences Date of Resignation Terr. Experiences Date of Resignation Terr. Course Code Date of Resignation Terr. Experiences Date of Resignation Terr. Date of Resignation Terr. Experiences

C

5

e

G

O

Entities (cont'd.)	Description (English)	Transaction Record Military Resort Command Military Rayon Command S.S.N. Transaction Code First Field Second Field Third Field Personnel/Material Code	
Table V onnel Application Data	Description (Indonesian)	Laporan Mutasi Korem Kodim Koramil NRP/NBI NRP/NBI NGde Mutasi Field Kedua Field Kedua Field Ketiga Kode Pers/Mat	
tcrial Pers	Fntity Name	MU TTER KOREM KORIM NRP NUTASI FIELD1 FIELD1 FOELD2 KODE	
Terri	Entity Type	Reconstruction Recons	
	NC .	- Nr 3000000	

ſ

ſ

C

C

	Description (English)	List all perschel eligible for rank promotion (each semester)	Print all individual personnel data as feed- fack to the lowest unit level (each semester)	Fersonnel Master File Up- Dating (monthly)	Corversion of Current Fayroll Master File irom Old to New Version (monthly)	Payroll Transaction Data Editing in Pullalta Kotama (monthly)	Payroll Transaction Data Editing in Dispullantal (monthly)	Payrol, Transaction Lata Sorted By Payroll Number (monthly)	Cli Payroll Master File Sorted by Payroll Number (monthly)
TABLE VI Process Entities	Description (Indonesian)	Membuat daftar personil yang eligible untuk kenaikan parykat (tiaf semester)	Cetak data personil perorangan, untuk feed- back ke kesatuan (tiap semester)	Update Master Tile Fer- sonil (tiap bulan)	Konversi AF DPP Bulun Ini dari Versi Lama ke Versi Baru (tiap bulan)	Edit Laporan Mutasi di Pullahta Kotama (tiap bulan)	Edit Laporan Mutasi di Dispullahtad (tiap bular)	Sort Laporan Mutasi by No. DPP (tiap bulan)	Sort Master File DFP Bulan Lalu by No. DFP (tiap bulan)
	Entity Name	ELIGIBLE	FEEDBACK	ISVINW	LPPR01	CP P0 1 A	LPPS01	DPPS02	LPPS03
	Pertinent To Appl.	Personnel	Fersonel	Personnel	Payroll	Payroll	Favroll	Payroll	Fayroll
	• • • •	-	•	m	- t	ي	6 .	7.	8

Ţ

7

f

1

C

Ö

s Data Number Sort Merge by Payroll Number between Newest Master File with Matched Main Data (monthly) đ Ø -d n Fayroll Main Data's Dat Editing in Disfullahtad (monthly) Dat. Double Check of Payroll
Main Data (monthly) Matching between New Master File and Merged Main Data (monthly) Sort Merge by Payroll Number between Payroll Main Data with Number and with New Number (monthly) Create Payroll Number to Main Data (monthly) l Main Data's j in Pullahta (monthly) Master File (monthly) Payroll Main Data': Sorted By Payroll | (monthly) Payroll Editing Kotama (Payroll Dating (Pokok (cont'd.) Sort Merge antara Dapok Bernomer DPP dengan Dapok Bernomer DPP Baru di-create (tiap bulan) Pemutakhiran Master File DPP (tiap bulan) Sort Merge antara New Master II dengan Matched Dapok New dì di Data DPP Matching antara Master 1 dengan Gabungan Dapok (tiap bulan) Edit Data Pokok Pullahta Kotama (tiap bulan) Edit Data Pokok Dispullahtad (tiap bulan) Sort Data Pokok by No. DPP (tiap bulan) Table VI Pemberian No. (tiap bulan) Entities Check Double (tiap bulan) (tiap bulan) Process DPPS09 DPPS10 LP PS04 **DPPO5A DPPS05 LPPS06 LPPS07 LPPS08** LPPS11 Payroll **Payroll** Payroll Payroll Payroll Payroll **Payroll Payroll** Payroll 10. 11. 13. . ហំ ٠ 12. 14 16. 17 σ

Į

•

ſ

ſ

Ţ

SSN Sort/Yerge to combine the Separated Master File for Ration Report (monthiy) Create Master for Payroll Lists for each Kotama (monthly) SSN's selcheck γd hγ th€ doubled (monthly) Sort Separated Master File by SSN (monthly) File Frint Payroll Lists Report double (monthly) Master File Sorted
(monthly) Separate Master Kotama/Lakpus (monthly) Frint Ration (monthly) Verifying Main Data Calculate (monthly) Check for Main Jata (monthly) (cont'd.) Kotama Sort/Merge Master per Kotama untuk Cetak Ke-kuatan Pangan (tiap bulan) Create Master Hitung per Kotama/Lakpus (tiap bulan) Perbaikan DPP double (tiap bulan) per NRP Cetak DPP (KU-107, KU-108, Rekap Pot: Persit KCK) (tiap bulan) Pemisahan Master Kotama/Lakpus (tiap bulan) DPP double bulan) per γd Table VI Entities Sort Master p by NRP (tiap bulan) KU-102 bulan) Create Notam (tiap bulan) Sort Master (tiap bulan) Check (tiap Cetak (tiap Process **LPPS12** DPPS13 DPPS15 DPPS16 **DPPS18** DPPS19 DPPS20 **LPPS14 DPPS17** DPPS21 Payroll Payroll Payroll Payroll Payroll **Payroll Payroll** Payroll Payroll Payroll 19. 21. 22. 23. 24. 25. 26. 20. 27. 18

l

٢

C

C

<u>_</u>___

55

.

.

. .

	Sort/Merge to combine Bad Transaction Data (monthly) Frint Bad Transaction	uata zor reel uatk (monthly) Sort/Merge to combine Bad Main Data (monthly)	Print Bad Main Data for Feed back (monthly)	Updating the Intelligence Personnel Record (any time)	Frint the Intelligence Personnel Record for Feed-back (each semester)	Updating the Territorial Personnel Record (any time)
Table VI cocess Entities (cont'd.)	Sort/Merge Laporan Mu- tasi Ditolak (tiap bulan) Cetak Laporan Mutasi	uttolak untuk reeu-pack (tiap bulan) Sort/Merge Dapok Ditolak (tiap bulan)	Cetak Dapok Ditolak untuk Feed-back (tiap bulan)	Updating Data Perorangan Apintel (tiap saat)	Cetak Data Perorangan Apintel untuk feed- back (tiap 6 bulan)	Updating Data Peroranyan Apter (tiap saat)
ΡI	LPPS22 LPPS23	DPPS24	DPPS25	UPDATE	FEEDBACK	UPDATE
	Payroll Payroll	Payroll	Payroll	Intell.	Intell.	Terr.
	28. 29.	-0E	25.	30 .	31.	32.

r

1

(<

Ō

		TABLE VII	
		User Entitles	
No.	Entity Name	Description (Indonesian)	Description (English)
~	DISPUL	Dinas Pengumpulan dan Pengolahan Data TNI-AD (DISPULLAHTAD)	Indonesian Army Data Col- lecting and Processing Service (DISPULLAHTAD)
7	SPAM	Staf Pengamanan TNI-AD (SPAM SUAD)	Indonesian Army Intelligence Staff (SPAM SUAD)
m	SPERS	Staf Personil TNI-AD (SPERS SUAD)	Indonesian Army Pérsonnel Staff (SPERS SDAD)
4	STER	Staf Teritorial TNI-AD (STER SUAD)	Indonesian Army Territorial Staff (STER SOAD)
ſ	JANKUAD	Jawatan Keuangan TNI-AD (JANKUAD)	Indonesian Army Finance Service (JANKUAD)
Q	MINPERS	Jawatan Administrasi dan Personil TNI-AD (JANMINPERSAD)	Indonesian Army Administra- tive and Personnel Service (JANMINPERSAD)
٢	DALKAR	Dinas Penyendalian Karier TNI-AD (DISDALKARAD)	Indonesian Army Carrier Centrol (DISDALKARAD)
ω	MDMD	Semua eselon organisasi pada TWI-AD	General, all user entities in Indonesian Army Organization

C

1

9

Q

2. <u>Relationships</u>

The relationships can be represented by the following relations:

RELATIONSHIPS (relation_name, key, composite_key, secondary_key, other_entity_names) ---- key ----

a. Relation Name

In pre-DFMS situation, this may be filled with the record entity name to represent the relationships between element data entities. In the case where a DBMS has been implemented, it should be filled with the relation name since a logical record may consist of some relations in order to reduce complexity and/or fulfill the five normal forms. t. Key

This is an entity name that is used as the key for both storing and accessing the relation. If the relation uses composite keys, this will be the first part of the composite keys where the second part will be stored as composite-key attribute.

c. Composite Key

This is the entity name used as the second part of a composite key. If primary key is not composite, this attribute will be filled with "NONE".

d. Secondary Key

This is the entity name used as the secondary key. If the relation has no secondary key, this attribute will be filled with "NONE". e. Other Entity Names

The names of other entities (except keys) in the relation will be filled by this attribute. An ampersand sign (5) will be used between two entity names and a sentence of "REPETITIONS OF" will be written in front of a repeating group of elements.

3. <u>Attributes</u>

ſ

1 9

Every entity has information attached to it called attributes. In the following, all information that may be included as attributes will be discussed.

a. Data Entities

There will be several pieces of information included for each data entity (either file, record, or element entity). Since the DD typically be implemented as one of the DBMS application, these can be represented as a relation of:

FILE_ENTITY (entity_name, block_size, access_method, logical_ record_size, physical_storage_device) --- key ---RECORD_ENTITY (entity_name, length, fixed_variable_code, key, composite_key, secondary_key, updating_time, updating_mode) --- key ---

ELEMENT_ENTITY (entity_name, length, code, source, user, definition) --- key ---

(1) Entity Name.

The name of the entity is limited to a maximum of eight characters, this may be a combination of alphatetic and numeric, but should contain an alphatetic as its first character.

(2) <u>Block Size</u>.
 Self explanatory.

(3) Access Method.

This may be encoded as:

- S Sequential
- I Indexed Sequential
- D Direct Access
- V Virtual Storage (VSAM)
- (4) <u>Logical Record Size</u>. This is equal to record length.
- (5) <u>Physical Storage Device</u>.

This way be coded as the following:

- TAPE Magnetic Tape
- DISK Magnetic Disk
- DRUM Magnetic Drum
- FICP Floppy Disk / Diskette
- CARD Punched Card
- PAFR Paper Tape

(6) <u>Fixed / Variable Code</u>.

- The codes used are:
 - FIX Fixed Length Record
- VAR Variable Length Record
- (7) <u>Key</u>.

5

This is an entity name used as the key for both storing and accessing the relation. If the relation uses composite keys, this will be the first part of the composite keys where the second part will be stored as composite-key attribute.

(8) <u>Composite Key</u>.

This is the entity name used as the second part of composite key. If a primary key is not composite, this attribute will be filled with "NONE".

(9) <u>Secondary Key</u>.

This is the entity name used as the secondary key. If the record has no secondary key, this attribute will be filled with "NONE". (10) Updating Time.

This attribute contains a description about how often this record will be updated. This will be a number of days.

(11) <u>Updating Mode</u>.

This attribute contains description about how the updating is done. It is encoded as follows:

> BATCH - Batch Processing ONLIN - Online Processing BOTH - Both of Batch and Online

(12) Entity Length.

This denotes the length of an entity (may be record, or element). In the case of variable record, this information will be filled with zeroes, since the length of each record will be attached within the record itself.

(13) Entity Code.

This is a code for the character type:

- A Alphabetic
- N Numeric
- AN Aphanumeric.

(14) Source.

This denotes the organizational entity responsible for providing, updating, and deleting the entity.

(15) <u>Users</u>.

This denotes the organizational entity (entities) allowed to retrieve and use the entity. If subschemas are applied to the DD, this attribute will not be necessary.

(16) <u>Definition</u>.

This provides a detailed narrative describing the entity. This may include the information about frequency_of_update, range_of_acceptable values, and so on.

(17) <u>Alias</u>.

At the present time, every entity name (especially element entity) has no alias attribute attached to. In the future, this attribute sure will be needed. One way to accomodate this need is by add another relation called alias relation in which has at least two attributes: entity-name and its alias-name.

t. Process Entities

 \mathbb{C}

The information included as attributes in the process entity are: name, input_entity, output_entity, and the description of the process. These can be represented as the relation of:

PROCESS (process_name, input_entity, output_entity, description_ of_process, descr_of_input, input_media, descr_of_output, output_media)

But, since a process may have more than one of either input or output entity, this relation should have a composite keys rather than having only the name as its single key. Because only one input/output entity is allowed in every instance of FROCESS [Ref. 6], processes having more than one input/ output entity may waste storage since all attributes will appear unnecessarily more than once. In the case that this relation has a composite key, a query asking which data entities are input (or output) to a given process also possesses a difficulty since this relation can not be retrieved using only the process_name (it must be retrieved using its composite key, instead). In order to make a better payoff, those attributes may be arranged using the following three relations:

PRCCESS (process_name, description)

PROCESS_INPUT (process_name, input_entity, description, media) PROCESS_OUTPUT (process_name,output_entity,description, media)

(1) <u>Process Name</u>.

17

Process name will be limited to maximum of eight characters as applied to data entities.

(2) <u>Description of Process</u>.

Self explanatory.

(3) Input Data.

The input data may be either a file, record, group of elements, data element, or data entered from console.

(4) <u>Description of Input</u>.

This attribute may be filled with a description such as the input data is "sorted by FANK" for instance.

(5) <u>Input Media</u>.

As to physical storage device attribute, this attribute may be filled with input storage device, or data entered via console.

TAPE	-	Magnetic Tape
DISK	-	Magnetic Disk
DRUM	-	Magnetic Drum
FLCP	-	Floppy Disk / Diskette
CARD	-	Punched Card
PAFR	-	Paper Tape
CCNS	-	Data Entered via Console

(6) <u>Output Data</u>.

The output data may be magnetic-data, displayed data, or printed material.

(7) <u>Description of Cutput</u>.

This attribute may be filled with a description such as the output data is "sorted by KOTAMA" for instance.

(8) <u>Input Media</u>.

This attribute may be filled with output storage device or printed output material and these are encoded as follows:

> TAFE - Magnetic Tape DISK - Magnetic Disk DEUM - Magnetic Drum FICP - Floppy Disk / Diskette CAED - Punched Card PAER - Paper Tape PEIN - Printed Material

c. Usage Entities

ſ

1

The information pertinent to these entities have been included and can be derived from data entities in terms of who is responsible for update operations and who is allowed to retrieve a data entity. Here, this information will be stated again from the reverse point of view, that is, which data are the responsibility of this entity, an đ which data are allowed to be retrieved by this entity. Information that will be included as the usage entity's attributes are: user_name, description of the user, entity_name, and type_of_access. For reasons similar to those discussed concerning the process entities, since a given user may have more than one data-entity as its resposibility and/or to be retrieved to, these attributes may be represented by the following three relations:

(1) <u>User Name</u>.

As with the data entity and process entity names, the usage entity name will be limited to a maximum of eight characters, toc.

(2) <u>Entity Name</u>.

Entity_name here means a data_entity_rame.

(3) <u>User Responsibility</u>.

This may be a list of file, record, group of element, or data element entities that are this user's responsibility. This can be viewed as subschema.

(4) <u>User Access</u>.

This may be a list of file, record, group of element, or data element entities that may be retrieve by this user. This can be viewed as subschema, and may or may not be same as the list of user responsibility items.

(5) <u>Type of Access</u>.

The type_of_access is either:

- F Read only
- U Update only
- E Both R and U
- N No access
- (6) <u>Description of Entity</u>. Self explanatory.

d. Summary of Relations

Table VIII summarizes the relations of relationships, data entities, process entities, and usage entities.

e. Example of Relations

Table IX presents examples of file, record, and element entity relations. Examples of process entity relations and user entity relations are presented as Table X. Finally, an instance of the relationship relation is shown in Table XI.



Ś

(

66

Ì

			Exanp	TABL les of Data	.E IX 1 Entity Re	elations		
Element	Entiti	0 I						
Relatior ELEMENT-	ENTITY	(enti)	ty_rame,14 (ey	ength,code,	source, use	er,đefinitio	(u	
Entity- name	Entity length	Char. tyre code	User- respons. (source)	User- allow to access	Definiti(0 h		
NC FERS N A MA	14 26	A N A	MINPERS MINPERS	иии И по и	SSN plus duplicati Personne the only	selfcheck-d ion number l nare birt formal sour	igit and h certific ce locumen	ate is t
Record 6	entitie.	ທ່າ						
RECORD-I	SNTITY (entit] ke	/_name,len sec	ngth,fixed condary_key	variable.	code,key,com _time,updati	posite key hg_modë)	
Entity- name 	Pecor lengt		Fixed/ variable code	Primary- key 	Composite key	Secondary U key	pdating U ime	plating ode
PERSINTI DAFCKDPI	295	,	11.12	NOPERS DPP	NONE NONE	NONE NONE 3	0	NLIN

Relation: FILE-ENTITY (entity_name,block_size,access_method,logical_record_size, Examples of Data Entity Relations (cont'd.) Physical-storage-device DISK DISK Logical-record-size Table IX 295 128 Access-method 1 5 Block-size 10 20 File Entities. FILETERR FILEINTL Entity-name

•

5

J

68

. . . .

Process	Exam]	ples of Pı	TABLE X Cocess & User 1	Entity	Relations		
PROCESS PROCESS PROCESS PROCESS	ENTITY (process INPUT (process UTPUT (process	name, desc 	rtiption) ==etity,descri out_entity,descri	iption, criptic	media) :n, media)		
Prccess name	Process description	Input name	Input description	Input media	Cutput name	Output description	Output media
DPFS11	Sort merge by Payroll Numb. between New Masterfile E Matched Lata	DA POKD P P	Matched Main Data (output of DPPS10)	FLOF	LA FOK DPP	New Master- file ready as the in- put to DPPS12	TAPE
		DA PCKDPF	New Master- file pre- merge (output of Dpps10)	TAPE			
Responsible for entities FERSFILE PERSINTL User Entity Relations (cont'd.) 1 access Type of ы П П П മല്ല Allowed access to entities _access) PERSFILE PERSINTL PERSTERR PERSFILE PERSINTL PERSTERR Table X Relation: USFR_ENTITY (user_name, description) USFR_ACCESS (user_name, entity_name, type_of-USFR_ACCESS (user_name, entity_name, type_of-USFR_FESPOSIBILITY (user_name, entity_name) Indonesian Army Intelligence Staff (SPAM TNI-AD) ω Indonesian Army Personnel Staff (SFERS TNI-AC) Examples of Process Description of User entity User entities. 1 User entity 1 SPERS SPAM 1

•

4

Q

Les and a second

Ł

TABLE XI Examples of Relationship Relation Relationship Entities. RelationSHIPS (relation_name, key, composite_key, secondary_key, other_entity_names) RELATIONSHIPS (relation_name, key, composite_key, secondary_key, other_entity_names)	Relation Primary Composite Secondary Other Entity Names PEFSINTL NOPERS NONE NONE NAMA E PANGKAT E CORPS E NOPERS E TPTLAHIR E PANGKAT E CORPS E NOPERS E STAMAT E TPTLAHIR E PANGKAT E CORPS E NOPERS E ABBATAN E PANGKAT E CORPS E NOPERS E TPTLAHIR E ANAKER E DIKUM E DIKMIL E NAMA E TAMUNK E TAMANA E FICHANA E TAMANA REPETITIONS OF : REPETITIONS
---	---

4. Example of Data Dictionary's Queries

Meta-data contained within data dictionary may be used to answer questions asked by top managers, users, or technical staff (such as system analysts, programmers, and operators). In the following, several queries and the corresponding responses will be presented.

a. Top Management Queries

Top management may ask a question like: "How often is the personnel masterfile updated ?"

Possible answer is:

PERSFILE is updated once every 30 days.

t. User Queries

The user responsible for personnel management may ask the following question: "I need a list of personnel having rank of captain who speak French fluently and are experienced in the intelligence field. Is DISPULLAHTAD atle to provide these data ?".

Possible answer is:

See entities:

1

1.	PANGKAT	(rank)	in	relation	PERSINTL
2.	ASING	(foreign language)	in	relation	PERSINTL
3.	AKPASING	(active/passive code)	in	relation	PERSINTL

c. Technical Staff Queries

"What are the inputs and/or outputs of process DPPS17 ?", is one possible question asked by an operator for instance.

Possible answer is: Process DPPS17 Input is/are: 1. DAPOKDPP (sorted by DPP), media: TAPE Output is/are:

DAPOKDPP (sorted by KOTAMA), media: TAPE
 KEKPANGN (sorted by KOTAMA), media: DISK

d. Unanticipated Queries

(€.

One of the advantages of designing the dictionary using the relational model is it can accommodate unanticipated queries. As one of the DBMS's application, this dictionary supports any query expressible in a relational query language (e.g. SEQUEL).

V. THE INPLEMENTATION OF DATABASE AT DISPULLAHIAD

A. DATA DICTIONARY DESIGN AS A STEPPING-STONE

ŕ

The implementation of a database at DISPULLAHTAD may take advantage of the design of Data Dictionary in the preceding chapter in many ways:

• The designed DD may be used as the first DBMS application. Then, the experience gained here can be used in the future application of the real database implementation.

• DD as a repository of all meta-data will provide a full specification and description of all entities and relationships between them. Given this, the implementation of DBMS will be faster due to fewer changes and iterations in database development.

• In the case where an automatic database design tool is used, such as DATA LESIGNER, it may be interfaced with the DDS in order to take an advantage of the DDS contents. The database designer may benefit from the full descriptions of each entity contained within the data dictionary.

In this regard, the design of DD for current DISPULLAFTAD applications can be considered as a steppingstone to the DBMS implementation.

B. DISPULLAHTAD'S DATABASE DESIGN

From Tables II through V, it can be seen that there are many data element redundancies within the four applications. These redundancies need to be eliminated by designing a database fulfilling the five normal forms. One possible method is by gathering the common data elements in one relation, and other specific data pertinent to each of the four applications in separate relations. The specific data model

for each application must likely consist of more than one relation.

In designing the database, availability of data description may be exploited to make this work easier. For example, suppose the following relationship is designed in the personnel database:

MAINPERS (nopers, nama, pangkat, corps, jabatan, satminkl) --key-

Here, the possible descriptions contained within the dictionary that may be extracted are:

• Which files and records would have to be accessed in crder to establish an instance of this relation ?

• What is/are the key/composite keys of each records derived from the preceding query ?

• What is the length of each of those entities ?

Furthermore, in order to satisfy the five normal forms a full and clear description of each entity is needed. These descriptions are contained within the data dictionary. The issue of actual database design is beyond the scope of this thesis and is left for possible follow-on thesis work.

C. CHOOSING THE DATA DICTIONARY SYSTEM (DDS)

1. Features

5

(:

The available commercial DDS have most of the following features (see Figure 5.2 for more detail).

a. Dictionary Schema

This is a feature used to generate a manufacturer's standard schema, such as entities, relationships, and attributes. 1. Schema Extensibility

This is a feature whereby an installation is able to customize the manufacturer's standard schema by adding to it new entity-types, relationship-types, and attribute-types.

c. Dictionary Maintenance

This is a feature that enables an installation to create and maintain its data dictionary.

d. Reports and Queries

An installation may generate reports using metadata contained within the data dictionary. A DDS provides these abilities via these features.

e. Bridge/Interface Facility

This feature generates descriptions from the data dictionary needed by other systems, typically an application development tool such as DATA DESIGNER.

f. Program Access Facility

This enables an installation to extend the functionality of a DDS, i.e., the preparation of programs able to access data dictionary contents.

g. Status Facility

This feature provides the status of each entity, especially when the data dictionary is used in the system life cycle, by providing aids in application development.

h. Security Facility

An installation may restrict access to the data dictionary to authorized personnel only. This requirement is fulfilled by this feature.

2. <u>DISPULLAHTAD's DDS Recuirements</u>

a. Data Dictionary Features

1

There are five data dictionary features required in order to implement DISPULLAHTAD's data dictionary. These are:

• Dictionary Schema which is needed in order to generate the entities, relationships, and attributes.

• Schema Extensibility which will be needed to accommodate specific needs that cannot be fulfilled by the dictionary schema.

• Dictionary Maintenance which is used to create and maintain the data dictionary.

• Reports and Queries which are required in order to generate reports from and to extract data contained within data dictionary.

• Security Facility, even though the data dictionary has no actual instances of "secured" data, it will contain information about such data that can be used to access it, i.e., entity_name, access_method, where such data are stored, etc. Therefore, a security facility is required to add and strengten the level of security.

Other features are optional. These can be considered as "nice to have".

b. Active Versus Passive Data Dictionary

A full-active or partially-active system is very desirable because it provides features such as enforcing standards, range_of_value auditing, transaction monitoring, etc. On the other hand, an active system possesses much overhead, suffers in terms of longer turn-around time, and requires more complex processing algorithms. Therefore, at this time, a passive data dictionary system is more appropriate for DISPULLAHTAD. In the future, after enough

experience has been gained, a partial-active system may be applied in order to take fuller advantage of the data dictionary.

c. Free-standing Versus DBMS-dependent System

A free-standing system is very appropriate for an installation having different DBMSs. This may happen in an installation with databases using network or hierarchical structures in conjuction with newer technology such as the relational system. In this case, all systems may access the data dictionary independently, since the usage of the lata dictionary is not limited to any one system.

On the other hand, a data dictionary may be implemented as one of the DBMS applications (some DDSs are implemented in this manner) [Ref. 8]. This approach is appropriate for an installation implementing databases using a single DBMS, and DISPULLAHTAD falls into this category. Therefore, a DBMS-dependent system will provide more advantages for DISPULLAHTAE, e.g., it can be used as a training tool in implementing the database. Another possible advantage is that if DISPULLAHTAD should change from a passive to an active system, there will not be too many modifications required because the data dictionary and the databases are already compatible.

d. Make or Euy

ſ

Buying an available commercial system may provide a high quality and ready-to-use system. Figure 5.2 summarizes features of current commercial systems that may be used in choosing the best system fulfilling the required features. A commercial system used by many installations without much trouble may be an indication that it has a certain quality. Criteria listed in Figure 5.1 may be used in selecting the best system for DISPULLAHTAD.

 It should have at least the following five features that may be considered as the primary criteria:

 a. Dictionary Schema.
 b. Schema Extensibility.
 c. Dictionary Maintenance.
 d. Reports and Queries.
 e. Security Facility.

 The following fcur criteria may be considered as the secondary criteria:

 a. Compatible with current hardware.
 b. Compatible with applied DBMS.
 c. High quality assurance.
 d. Low Acquisition and Set-up Costs.

Figure 5.1 Criteria for Choosing Commercial Data Dictionary System.

On the other hand, one significant advantage of designing a DDS in-house is that the system can be fitted to specific requirement of the installation. Furthermore, given that DISFULLAHTAD will implement a DBMS, the relational dictionary presented in the previous chapter may be a good candidate for the first application. By implementing the dictionary as a relational database, only one system needs to be acquired (DBMS) instead of two (DBMS and DDS).

3. <u>Recommendation</u>

.

(

5

1

s.

For the reasons discussed in the previous section, it is better for DISPULLAHTAD to implement the data dictionary model described in the previous chapter as the first application of the DBMS. The dictionary will be a DBMS-dependent system. Initially, the system should be passive, and later, if appropriate, it may be changed to an active system.

) 9 1 1 1			8 0 0 8 8	
* *	* *	¥ ¥	* *	* *	N /A	* !!
*	*	¥	*	*	*	*
*	¥	¥	*	*	*	¥
#	¥	*	*	*	*	*
¥	* *,	#	*	*	¥	*
¥	¥	¥	¥	¥	¥	NZA
cess *	N/A	NZA	N/A	N/A	N / A	NZA
rean m N/A	*	N/A	ИЛА	N/N	N/A	N/A

.

Z

5

4

C

Q

ŀ

Pigure 5.2 Features of Current Commercial DDS
(summarized from Lefkovits et al [Ref. 1]) .

and a second second

Ę

VI. CONCLUSION

Implementing a DEMS is a "must" for DISPULLANTAD in order to control the proliferation of its applications that in turn raises problems of data redundancy and data inconsistency. Primarily, a DBMS provides data manipulation capabilities whereas a data dictionary provides management and control. Applying management and control (by implementing DDS) first will make the job of database design and implementation easier in term of lessening the difficulties and the time and effort required to develop databases. In this regard, designing a data dictionary may be considered as a stepping-stone to the implementation of a DBMS.

This thesis has presented a relational model of a dictionary which can satisfy the needs of DISPULLAHTAD. The advantages of this model are:

1) it is compatible with any relational DBMS that DISPULLAHTAD may procure.

2) it obviates the need to buy a DDS in addition to the IBMS.

3) it can be tailored specifically to DISPULLAHTAD's needs because of the flexibility of the relational model.

4) it satisfies the criteria for a DDS (see Figure 5.1)

In order to attain the objective of managing and controlling data rescurces, the following implementation policy is recommended:

• All personnel involved in software development (such as system analysts, programmers, etc.) should use the lata dictionary extensively in doing their jcbs.

• Only the data administrator staff may update the data dictionary, others may access it in read-only mode.

• All suggestions concerning the data dictionary may be addressed to the data administrator staff.

This thesis has stopped short of suggesting a database design for DISPULLAHTAD's personnel application. Follow-on work could be done using the dictionary model suggested herein as a foundation.

C

C

Į.

C

Ē

•

82

Ł

LIST OF REFERENCES

1

1. Dolk, Daniel Roy, <u>A Dictionary/Directory System</u> (<u>CDS</u>) for <u>SPLICE</u>, Naval Postgraduate School, 1934

Ľ

Statistics and Statistics in the

- 2. Indonesian Army HQ, <u>Memorandum on the Project of</u> Indonesian Army Data <u>Collecting</u> and <u>Processing</u> Service, dated 9 May 1976, the office of Assistant of Flanning, Research, and Development to the Army Chief of Staff, Indonesian Army HQ, 1976
- Indonesian Army Data Collecting and Processing Service, <u>Indonesian Army Management Information</u> Systems - <u>Draft</u>, undated, Indonesian Army Data Collecting and Processing Service
- 4. Sprague Jr, Balp H., and Eric D. Carlson, <u>Building</u> <u>Effective Decision Support Systems</u>, Prentice-Hall, Inc., 1982
- 5. Allen, Frank W., Mary E.S. Loomis, and Michael V. Marnino, <u>The Integrated Dictionary/Directory System</u>, Computing Surveys, Vol. 14, No. 2, June 1982
- 6. Kroenke, David M., <u>Database Processing</u>: <u>Fundamentals</u>, <u>Design</u>, <u>Implementation</u>, Second Edition, Science Research Associates, Inc., 1983
- 7. Goldfine, Alan H., Editor <u>Database Directions</u> Information <u>Resource Management</u> - <u>Strategies</u> and <u>Tocls</u>, NBS Special Publication 500-92, ACM, National Bureau of Standards, Issued September 1982
- 8. Lefkovits, Henry C., Edgar H. Sibley, and Sandra L. Lefkovits, <u>Information Resource/Data Dictionary</u> Systems, QED Information Sciences, Inc., 1983
- 9. Indonesian Army HQ, Indonesian Army Chief of Staff Instruction, Nc.: INS/7/VII/1980, dated 17 July 1980, Indonesian Army HQ, 1980
- 10. Indonesian Army H2, <u>Indonesian Army Chief of Staff</u> <u>Instruction</u>, <u>1st Amendment</u> No.:INS77a/VII/1930, dated 17 February 1983, Indonesian Army H2, 1980
- 11. Indonesian Army HQ, <u>Technics Directive of Pavroll</u> <u>Application</u>, No.:JJKNISZU9ZIXZT987, dated 29 September 1987, Indonesian Army HQ, 1981
- 12. Indonesian Army HQ, <u>Technics Directive of Payroll</u> <u>Application</u>, <u>1st Amendment</u>, No.:JUKNIS/09a/1X/1981, dated 5 January 1983, Indonesian Army HQ, 1983

13. Indonesian Army HQ, <u>Technics Directive of Intelligent</u> <u>Fersonnel Application</u>, No.:JUKNIS7057V1/1975, Cated 15 June 1978, Indonesian Army HQ, 1978

- 14. Indonesian Army HQ, <u>Technics Directive of Intellijent</u> <u>Personnel Application</u>, No.:JUKLAK/16/IX/1982, dated 23 September 1982, Indonesian Army HQ, 1982
- 15. Indonesian Army HQ, <u>Technics Directive of Territorial</u> <u>Personnel Application</u>, Indonesian Army HQ, 1976

INITIAL DISTRIBUTION LIST

ſ

ł

		No.	Copies
1.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943		2
2.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314		2
3.	Department Chairman, Code 54 Department of Administrative Science Naval Postgraduate School		1
4.	Computer Technology Department, Code 37 Naval Postgraduate School Monterey, California 93943		1
5.	Assistant Professor Daniel Roy Dolk, Code 5 Department of Administrative Science Naval Postgraduate School	4DK	1
6.	Associate Professor Norman R. Lyons, Code S Department of Administrative Science Naval Postgraduate School	54LB	1
7.	Bamhang Sutedjo, Kapten Chb Nrp 26701 Dispullahta TNI-AD, Jl. Veteran No. 5 Jakarta Pusat, INDÓNESIA		1
8.	library of Dispullahtad Dispullahta TNI-AD, Jl. Veteran No. 5 Jakarta Pusat, INDÓNESIA		2
9.	Tony A Rumbayan, Kolonel Czi Dispullahta TNI-AD, Jl. veteran No. 5 Jakarta Pusat, INDÓNESIA		1
10.	Djoko Sardjono, Kolonel Cpl Dispullahta TNI-AD, Jl. Veteran No. 5 Jakarta Pusat, INDONESIA		1
11.	Triyono, Letnan Kolonel Czi Dispullahta TNI-AD, Jl. Veteran No. 5 Jakarta Pusat, INDÓNESIA		1
12.	Dadat Kustiwa, letnan Kolonel Chb Dispullahta TNI-AD, Jl. Veteran No. 5 Jakarta Pusat, INDÓNESIA		1
13.	Syafei Djamil, Ietnan Kolonel Chb 1121 Seventh Street Monterey, CA 93540		1
14.	Moch. Subekti, Kapten Laut (E) 1121 Seventh Street Monterey, CA 93940		1
15.	Widhya B Prawiraatmadja, Kapten Inf 1121 Seventh Street Monterey, CA 93940		1

END

FILMED

5-85

DTIC

.