PS/OP/Min. 87-30
21.10.1987

TECHNICAL BOARD ON PROCESS CONTROL AND ELECTRONICS FOR ACCELERATORS

CHAPITRE ASPECTS OPERATIONNELS

Compte rendu de la deuxième réunion du 6.10.1987

Présents : M. Chanel/PS, G. Daems/PS, L. Evans/SPS, A. Faugier/SPS, G. Guignard/LEP, B. Kuiper/PS, J.P. Potier/PS, G. Shering/PS
Excusé : M. Bouthéon/PS

* * * * *

1. Commentaires sur le compte rendu de la séance du 1.9.1987. Dans le chapitre "prochaines étapes" en page 2, une erreur s'est glissée; il faut lire "le 6 octobre, G. Guignard exposera les activités AAWG dans le cadre du LEP".

2. Services généraux

   J.P. Potier rapporte les contacts avec H. Laeger, responsable de l'accès aux services généraux du LEP.

   - Il apparaît qu'une étude devrait être lancée immédiatement pour définir des standards pour les représentations et les interactions avec ces systèmes.

   - Côté PS, la nécessité d'une telle étude apparaît également du fait de la réduction du nombre de personnes en roulement prévue pour la génératrice principale (Siemens), le refroidissement et les alimentations des faisceaux.

   Du fait de cette urgence, il est proposé de mettre sur pied un sous-groupe spécifique, composé de membres du PS, du SPS, du LEP et du ST, qui aura pour tâche :

   - de rechercher s'il existe des standards internationaux ou nationaux susceptibles de s'appliquer au CERN; à défaut ou en complément, d'établir des contacts avec des industries ou services ayant des activités similaires à nos services généraux;

   - de définir les standards applicables dans notre cadre;

   - finalement, de spécifier - en accord avec les responsables du contrôle - un ensemble de logiciels qui permettrait de faciliter et de renforcer la standardisation des accès aux systèmes généraux.

G. Daems a fait remarquer qu'il peut y avoir de fortes conséquences pour le Groupe Contrôles du PS et que celui-ci devrait être représenté dans le sous-groupe.

J. Boillot, côté PS, a été pressenti pour cette étude et des discussions sont en cours avec les différentes personnes concernées.

3. <u>Commentaires sur la conférence Contrôles de Villars</u>

B. Kuiper rapporte brièvement les points essentiels de la conférence : sessions essentiellement consacrées à des "Workshops" ou des "Tutorials", occasion donnée aux utilisateurs d'exprimer leur opinion sur les systèmes de contrôle.

Un point important soulevé lors de notre réunion est la grande incertitude sur l'évaluation des coûts en main d'oeuvre des logiciels. A ce sujet, il est fait remarquer que la pratique courante consiste à évaluer en hommes-années indépendamment de la qualification du personnel requise: accepterait-on dans d'autre domaines de totaliser main d'oeuvre, niveau universitaire et niveau technique ?

4. <u>Exposé sur les activités AAWG - G. Guignard</u>

G. Guignard rappelle l'approche de AAWG.

- C'est une approche liée essentiellement à la physique machine pour établir des modèles logiques des programmes d'application. La technique utilisée est le "structured analysis".

- AAWG travaille sur des requêtes faites par le LEPCC (LEP Commissioning Committee) au niveau des méthodes.

- Une autre équipe dirigée par R. Keyser fera l'étude détaillée et la réalisation des programmes.

On trouvera en annexe le Status Report de AAWG du 3.2.1987 d'où proviennent les transparents projetés au cours de notre réunion.

Au cours de la discussion, il a été soulevé le fait que cette étude est faite en supposant l'existence des outils software nécessiares. Au moment de l'étude détaillée, on pourra tomber sur quelques difficultés, en particulier avec l'accès en RT aux data banks.

Lors de la prochaine réunion, le 20 octobre 1987, I. Wilkie exposera l'approche SPS, qui est de même style mais descend jusqu'au design détaillé. M. Chanel exposera les méthodes employées à LEAR.

J.P. Potier

Distribution

Chapitre OPAS	:	M. Bouthéon
		M. Chanel
		G. Daems
		L. Evans/SPS
		G. Guignard/LEP
		G. Shering
		J.P. Potier
c.c. Membres TEBOCO	:	J. Altaber/SPS
		G. Baribaud/PS
		L. Burnod/SPS
		A. Daneels/PS
		B. Kuiper/PS
		M. Rabany/SPS
		R. Rausch/SPS
Pour information	:	Y. Baconnier/PS
		J. Boillot/PS
		V. Hatton/SPS
		P.G. Innocenti/SPS
		F. Perriollat/PS

Lep Note 565
31 july 1986

CONTROL, DATA & STORAGE MODELS FOR LEP OPERATION

J.-P. Koutchouk

1. AIM

It is customary to describe a control system in terms of its implementation (computer network,...) with emphasis on the ability to communicate information between a control center and various equipment in the field. This paper is concerned with another view of the control system which emphasizes **the data handling for accelerator operations.** At this level of description, efficient data exchange is assumed to be somehow provided and data processing and organization become of concern.

2. INTRODUCTION

In order to avoid any confusion, data are here defined to be the information created or used by applications programs (i.e. the programs which are not systems programs); dormant processes like programs on disk are therefore not considered as data.

Given this distinction, it is known that the data necessary to operate a storage ring or collected during its operation are numerous and multiform; in ordinary language, they are often referred to as being the "accelerator database", encompassing precalculated, measured or predicted information with varying storage requirements: read-only, dynamic or logged.

There is some arbitrariness in the choice of a data model; data are indeed only well defined when the algorithms (the actions) are themselves well defined, or even better when the operation has started; it is however too late to enforce a logical data organization. It seems therefore worth proposing an initial data model; to avoid a too arbitrary choice, the data model is derived from a control model which is easier to depict; the storage of the data is a separate requirement which deserves another separate 'model'.

Albeit important, this view of the accelerator data must be complemented by requirements of the user access to the data (symbolic names,access speed and protection,...), which emerge as well [1] .
In an attempt to clarify the model descriptions, a non-dogmatic use is made of SASD notations.

3. THE CONTROL SYSTEM MODEL

With respect to the user, an accelerator control system has essentially two aims:

1. the remote operation of individual components or subsystems, for commissionning, fault finding or system development.

2. the operation of the accelerator as a whole for operations and machine studies.

Each of these aims entails distinct requirements. The first one has guided the design of the SPS control system that can be regarded as a reference in this respect. This paper addresses the second aim, which is much more related to the end-user (accelerator operator or phycisist).

The control system model must put emphasis on the data which are exchanged by processes; we choose therefore to exclude considerations on timing (the sequence of accelerator states) that may hide the unity of the data, and data storage considerations which may confuse the issue at this stage. The so-called SASD data-flow diagram (DFD) is appropriate, in that it is time independant and makes no hypothesis on whether the data are permanently stored or not.

One may distinguish on figure 1 the major logical phases of accelerator operations : (the operator is not represented as he interacts with all the actions):

1. the operator provides the specifications for a complete run (successive optics, magnetic insertions,...)

2. in 1, this information is used, together with suitable archived information, to built a "Reference Data Set" (Twiss parameters, expected beam characteristics, power supply excitations for the various acceleration phases...); part of this processing may be skipped if a previous reference model was saved.

3. in 2 the relevant part of the Reference Data Set is selected, possibly trimmed (systematic trims) and transferred to the "Operational Data Set", which is supposed to be an accurate model of the accelerator.

4. use of these data is made in 3 to set the accelerator components according to operator requests or automatic sequencing.

5. machine adjustments (3) and measurements (4,5) are not allowed to modify the Operational Data Set but are rather collected in a separate set called "Current Data Set"; predictions of the model algorithms (6) are collected in a "Target Data Set" (closed orbit correctors, predicted orbit,...) and copied into the Current Data Set in case of action. Upon validation by the operator, data may be transferred from the Current to the Operational Data Set (7).

6. an autonomous task surveys the accelerator status, issues alarms and takes care of harware and software interlocks (8).

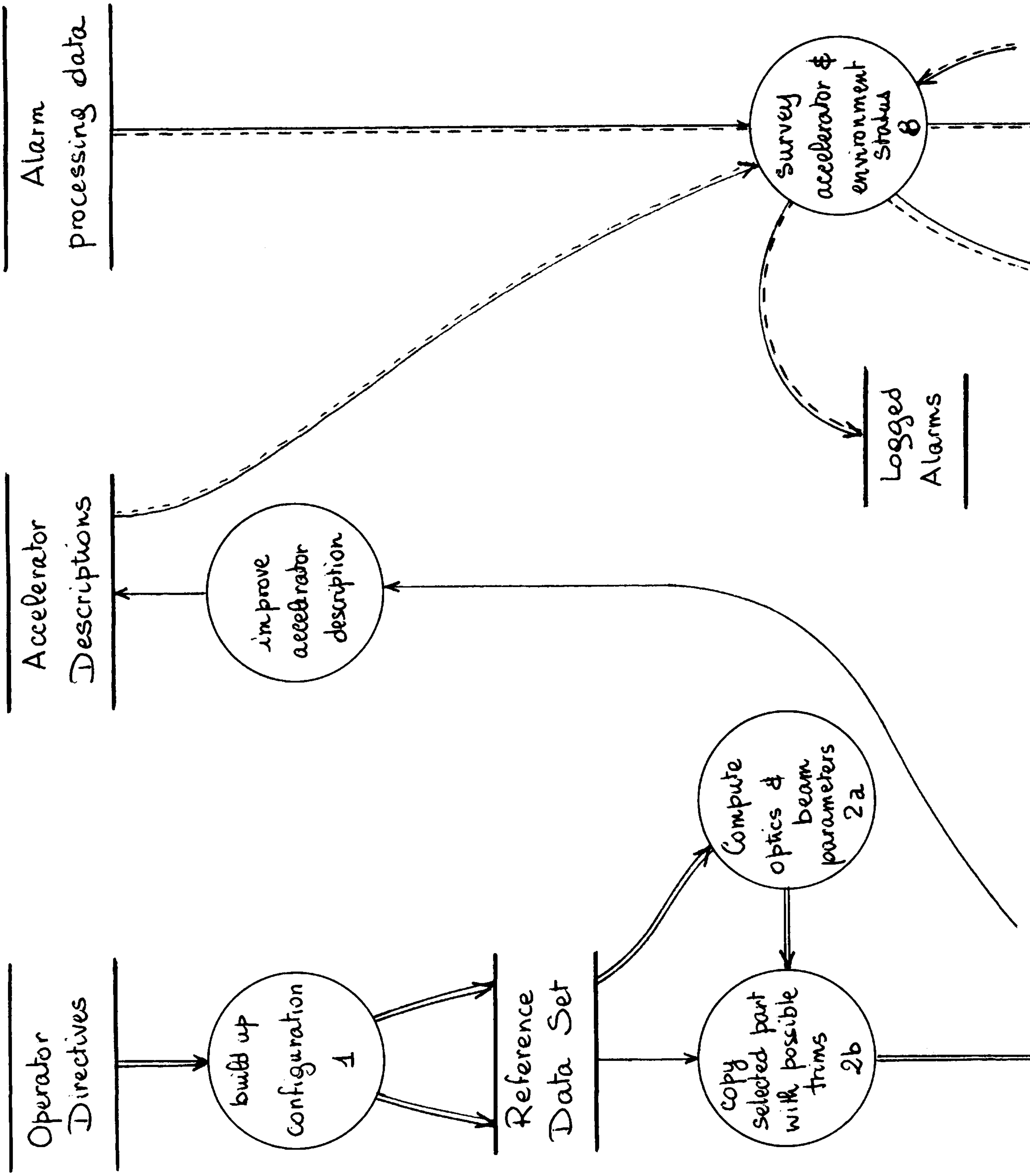
These phases may be grouped into three categories, that we have attempted to show using the stroke thickness:

1. the operation of a perfect and stable LEP

2. the correction and improvement process

3. the accelerator state control

The various data sets isolate these parts from each other, so as to enforce clarity and reliability.



Alarm processing data
Accelerator Descriptions
Operator Directives
build up configuration
1
Reference Data Set
improve accelerator description
Compute optics & beam parameters
2a
copy selected part with possible trims
2b
survey accelerator & environment status
8
Logged Alarms

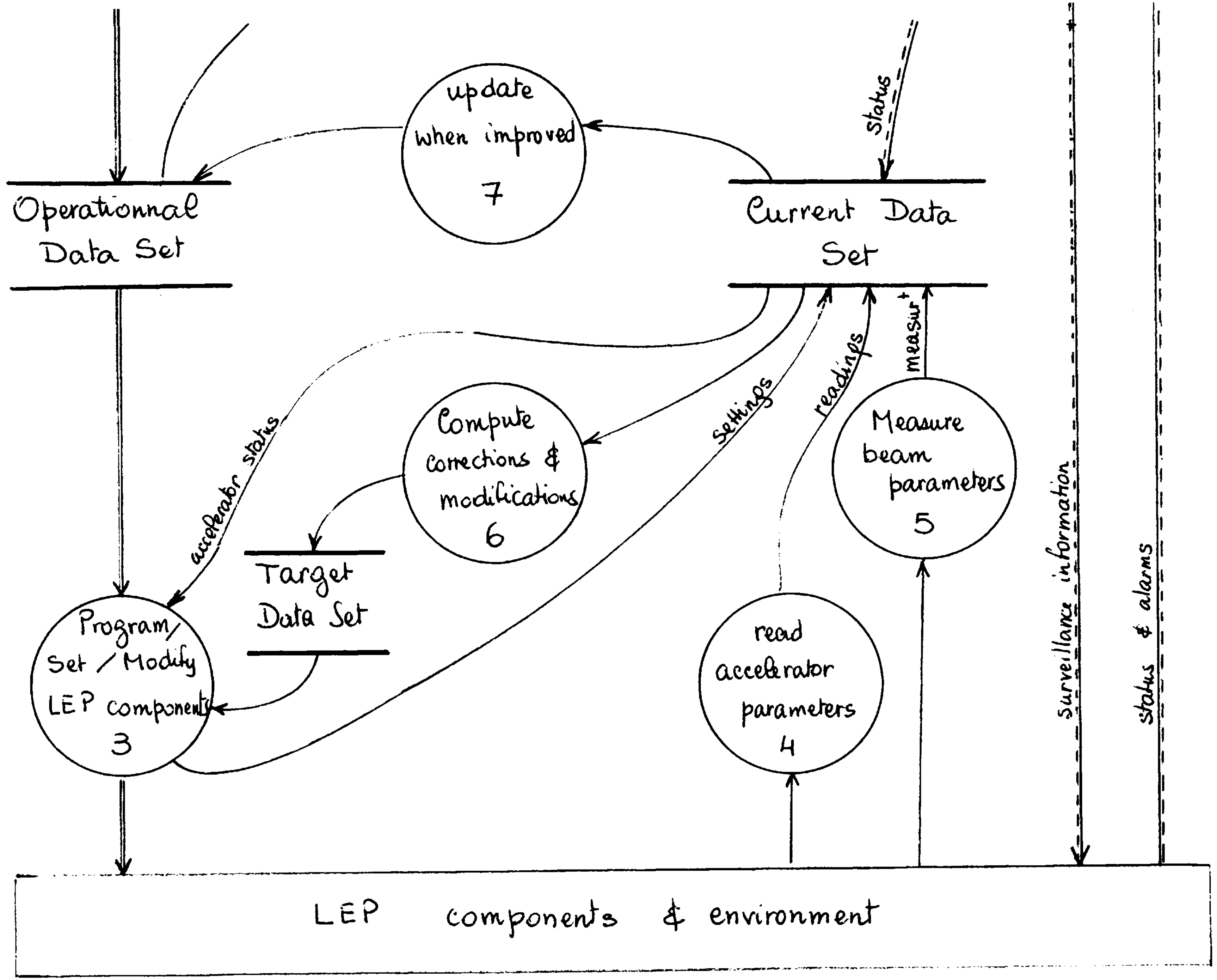


figure 1 - Control System DFD

4. THE DATA MODEL

The focus is put here on the semantical content of the data and not on their storage. From the control system diagram, the LEP data may be split into three categories represented on the following ERD (fig 2) where the boxes represent data and the diamonds relationships.

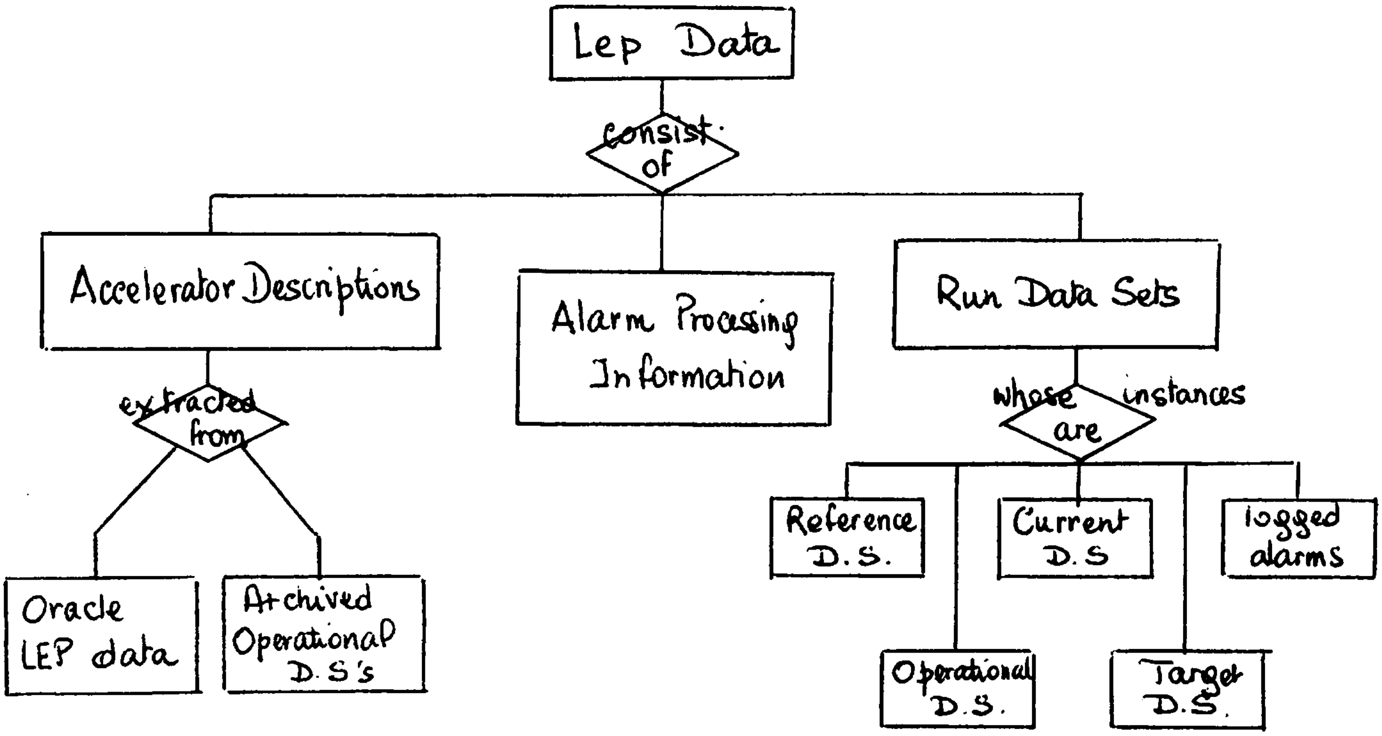


figure 2

Suggestions for the Accelerator Descriptions have been made elsewhere [2] [3] ; the definition of the Alarm Processing Information is not really the purpose of this paper; as regards the Run Data Sets, it should be observed that the data are not organized according to their types (optics,closed orbit,...) but rather according to a phase of operation. The expected advantage is to introduce naturally the physics relations between the data (eg. orbit measurement corresponding to some corrector settings) and to facilitate the storage management (what is to be stored). The relations between the Run Data Sets are shown on figure 3.

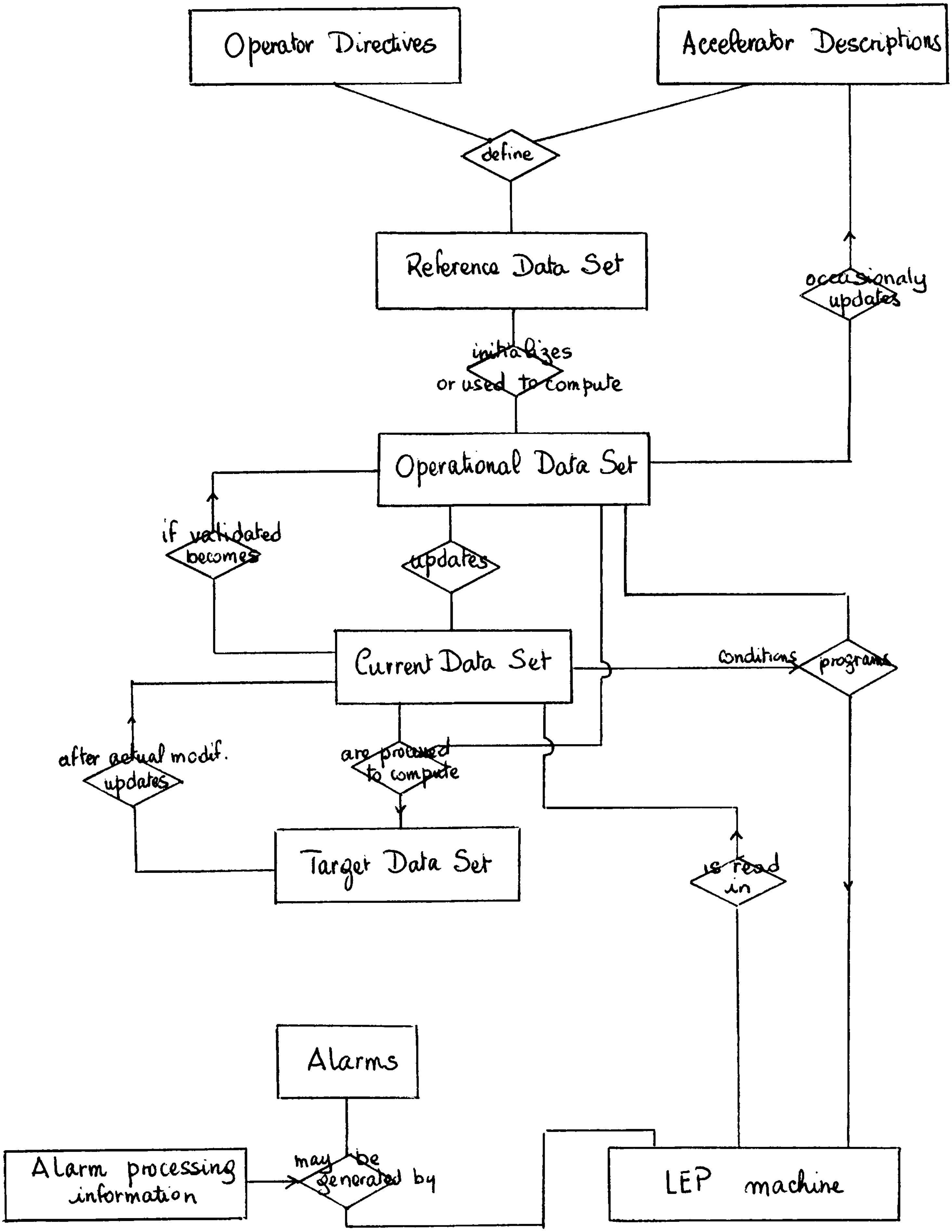
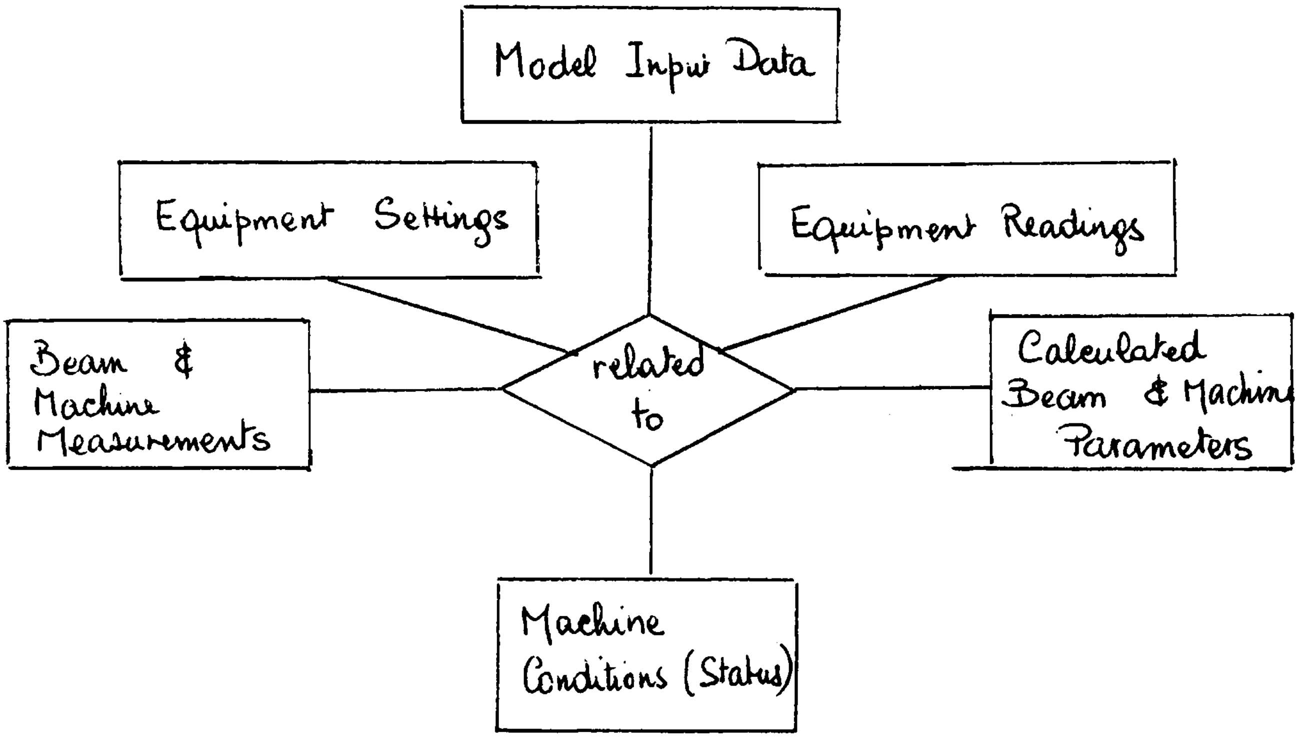

Operator Directives
Accelerator Descriptions
define
Reference Data Set
occasionaly updates
initializes
or used to compute
Operational Data Set
if validated
becomes
updates
Current Data Set
conditions
programs
after actual modif.
updates
are processed
to compute
Target Data Set
is read
in
Alarms
Alarm processing
information
may be
generated by
LEP machine

figure 3

Most of the data entities are implicitly or explicitly defined in the DFD comments; one may add some definitions or clarifications:

1. DATA SET: A data set is composed of interrelated information modules that could be decomposed in the following way:



   Some data sets may contain only part of this information; the general structure should however be retained. It is not yet clear whether all the data pertaining to a complete run should be organized in a single structure or whether a data set should consist of several structures corresponding to injection, ramping and flat top.

2. REFERENCE DATA SET: It is a transient data store for the initial calculations of the model algorithms; its essential aim is to provide a reliable starting point for machine cold starts and in case of difficulty and/or confusion.

3. OPERATIONAL DATA SET: This set is carried from one run to the other in case of stable operation.

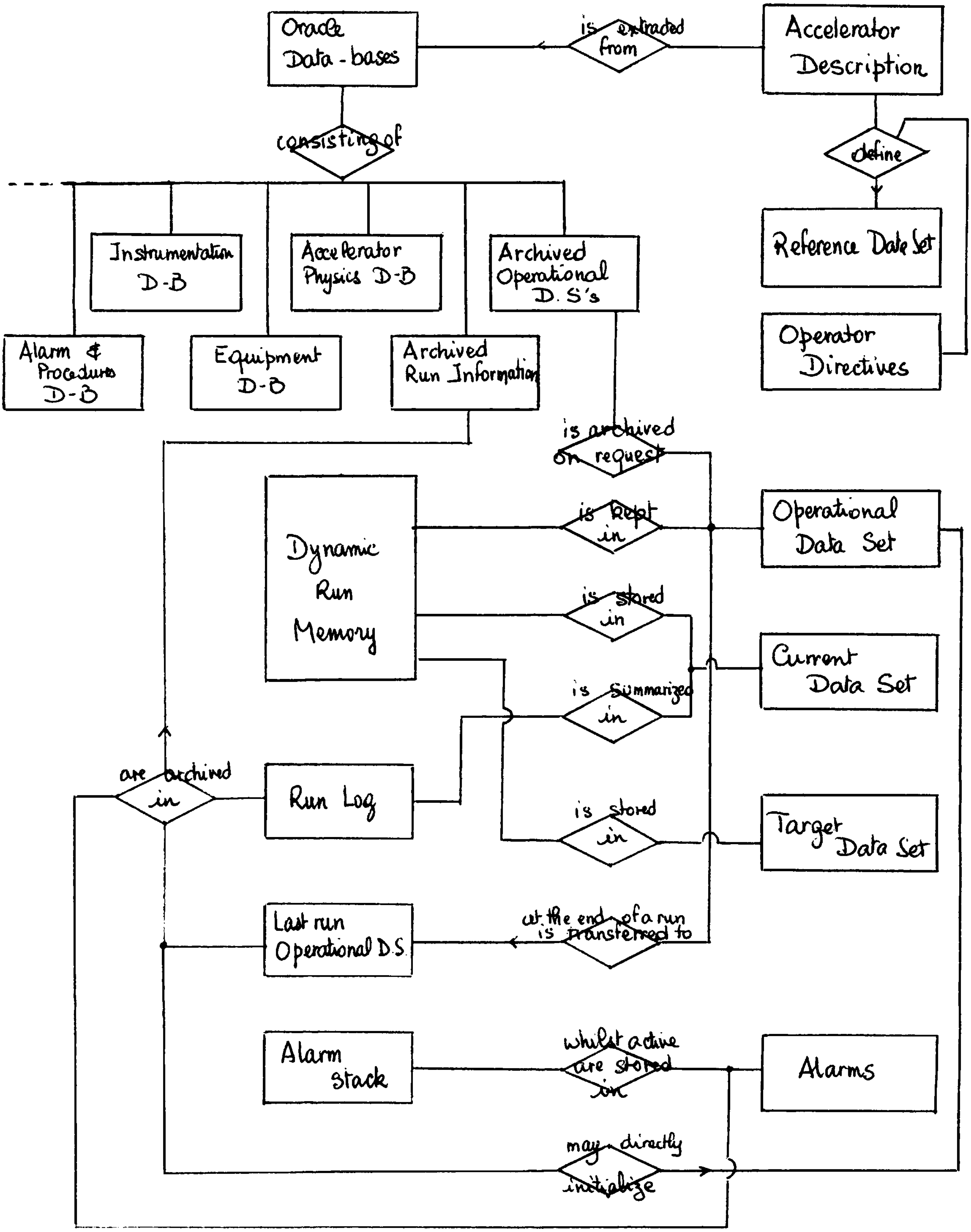
4. CURRENT DATA SET: It contains information on the successive accelerator states as far as they have been measured and not overwritten; it is the "working memory" of the system. It contains in addition the present status of the accelerator complex, which is used to enable or forbid certain actions.

The data model emphasizes three data loops which should provide a reliable accelerator operation:

- the correction, tuning and trimming cycle, where the machine is improved, taking into account the constraints of the day.
- the gradual improvement of the Operational Data Set, a well-known characteristic of accelerator performances improvement
- the occasional upgrade of the Accelerator Model, following a better knowledge or understanding of the machine.

5. THE STORAGE MODEL

The performance improvement is based on learning and post-mortem analysis. This process may be done on a short time scale (within a run) or on a long time scale (run to run or even over periods). It obviously shows the requirement of storing data in such a way that efficient use can be made of them. There seems to be a natural emergence of a storage model from the data model:


Oracle Data-bases
is extracted from
Accelerator Description
consisting of
define
Reference Date Set
Operator Directives
Instrumentation D-B
Accelerator Physics D-B
Archived Operational D.S's
Alarm & Procedures D-B
Equipment D-B
Archived Run Information
is archived on request
Dynamic Run Memory
is kept in
Operational Data Set
is stored in
Current Data Set
is summarized in
are archived in
Run Log
is stored in
Target Data Set
Last run Operational D.S.
at the end of a run is transferred to
Alarm stack
whilst active are stored in
Alarms
may directly initialize

The relevant storage units are:

1. The Oracle accelerator physics data base, which mainly contains the LEP structure and excitation sets presumably in MAD input language.

2. The archived Operational Data Sets that have been judged to be worth keeping.

3. The archived run information, classified by run number, which may be partly offline. Whether these last two storage units are within Oracle or simply known to it is not settled nor important at this stage.

4. The alarm processing information, which is simply transferred from Oracle to the alarm system.

5. The run log file collects essential information (eg. active commands, summary of results,...) that could be used to find correlations with faults (in fact, it is a replacement of the old typewriter).

6. The last run operational data set serves to pass information from a run to the next in case of stable operations, allows comparison of run to run performances and permits stable LEP operations to be independant of Oracle in case of necessity.

7. The dynamic run memory contains past, current and future values for machine and beam parameters, together with the machine state and interlocks. Memory stacks would be interesting to keep track of the past without creating too vast amounts of data files; the depth of the stacks remains to be defined.

6. DISCUSSION

The proposed models are inspired by the experience of running the ISR; in fact, the concepts developed were often not actually present as such in the control system; they were rather hidden behind operational procedures which gradually emerged over the years. The need to develop such facilities was essentially triggered by the search of better reliability, flexibility and higher automation [4] .

These factors are equally important for LEP, although its somewhat shorter cycle may modulate their relative importance.

The SPS experience has led to the decision of redefining the datastructures [5] ; the project follows a similar aim of improving the clarity and flexibility of the data; the analysis and proposed organization is somewhat different, being oriented towards the short and flexible SPS supercycles.

The databases described in MCCM's notes [6] [7] [8] are all included in the proposed model as far as they are of concern to the user; their logical organization is however different in that the classification criterion is not a data type but the logical link between the data. The other databases are equally important, but they find a more natural place in the description of the network operating system.

7. CONCLUSION

The proposed control system, data and storage models provide a top view of the data organization and a framework within which requirements may be better defined and implementation ideas may be tried. The essential requirements aimed at were:

1. the need to interrelate the accelerator data

2. the ability to tune the accelerator on a run to run basis and retain the interesting information

3. the ability to cope with the gradual improvement of the experimental and theoretical understanding of the accelerator behaviour, which requires flexibility and data analysis capabilities.

We have attempted to express the user needs, and thus no discussion on implementations is proposed; before doing so, the various data entities require a finer description.

8. ACKNOWLEDGMENTS

I would like to thank John Poole for carefully reading and commenting this paper.

REFERENCES

[1] J.P. Koutchouk, J. Poole The LEP dictionnary LEP controls note 69
[2] L.Tausch The distributed database for the LEP control system LEP controls note 66
[3] J.Poole Using Oracle databases in a particle accelerator system LEP controls note 68
[4] J.Gamble et al Towards full automation of accelerators thru computer control 11th Int.Conf. on HEA CERN 1980
[5] R.Bailey et al Ideas on data storage requirements for the new SPS control system SPS/AOP/SASD/note 18, may 1986
[6] M.Crowley-Milling Data bases in the LEP control system LEP note 375
[7] M.Crowley-Milling Data bases for LEP control LEP note 476
[8] M.Crowley-Milling Some notes on databases for LEP MCCM/LEP/2-86 Rev April 1986

3. 2. 1987
G. Guignard

Status Report from the Application Analysis Working Group

Composition of AAWG:

C. Fischer	BI	
G. Guignard	TH	(Chairman)
J.P. Koutchouk	TH	
J. Poole	DI	(Secretary)
A. Verdier	TH	
G. von Holtey	BI	
R. Wolf	MA	

with the help from (participation in all meetings)

R. Bailey SPS-AOP link with SPS work
R. Keyser SPS-ACC leading the design team

Analysis team is operational since september 1986 and had 13 meetings up to now.

Contacts are being developed with

P.G. Innocenti and SPS-ACC ; AAWG work presented in January
J. Miles, R. Giachino, I. Wilkie, J. Ulander ; working meeting foreseen on some particular analysis.
Hardware groups when necessary ; BI, MA, ...

The present reports includes

- Summary of the terms of reference: definitions and objectives
- Logical model of the control system and data sets agreed upon
- First list of procedures identified from LEP CC minutes and related documents
- Examples of logical models and analyses made for "measure" and "correct" procedures DFD's, Data Dictionary, Minispec, StD's

Interpretation of the Terms of Reference for
The LEP Application Analysis Working Group (AAWG)

1. Definition of the Application Programs

The application programs considered are high level software procedures relevent to control room operation; they generally involve co-ordinated action of several accelerator subsystems and interact with various data structures, equipment interfaces, operators etc.

These applications contain treatment of high level parameters, i.e. storage ring physical quantities. Using accelerator physics knowledge, they transform these high level parameters into measurable and adjustable quantities.

Many of the applications correspond to simple processes and logic. In some of them, mathematical models may be present in order to describe or change the behaviour of the circulating beams.

The following are the main tasks of the application programs:

- Set-up: e.g. generating ring configurations
- Initiating and supervising actions: e.g. injection and accumulation, ramping in energy, flat top and beam collision
- Measuring and controlling parameters:
  - Treatment and analysis of data (data handling)
  - Application of corrections (trimming)
  - Performance optimisation
  - Closed loop adjustment

LEP operation will also require utility programs (e.g. injection mode selection, software interlocks) which would be better specified by the future operations team.

3. Definition of the Objectives of the AAWG

The working group will analyse the needs or requirements expressed by the LEP Commissioning Committee (LEPCC) and its working groups (including representatives of the experimentors and the future operations team). The aim of this analysis is to identify the application programs and related data structures necessary for commissioning and operating LEP, and to define the processes and data flows involved, using physical criteria.

The starting point of this analysis is the summary of the needs written in the LEPCC minutes and related documents. These will be studied and reviewed in order to identify the common parts and to ensure coherence.

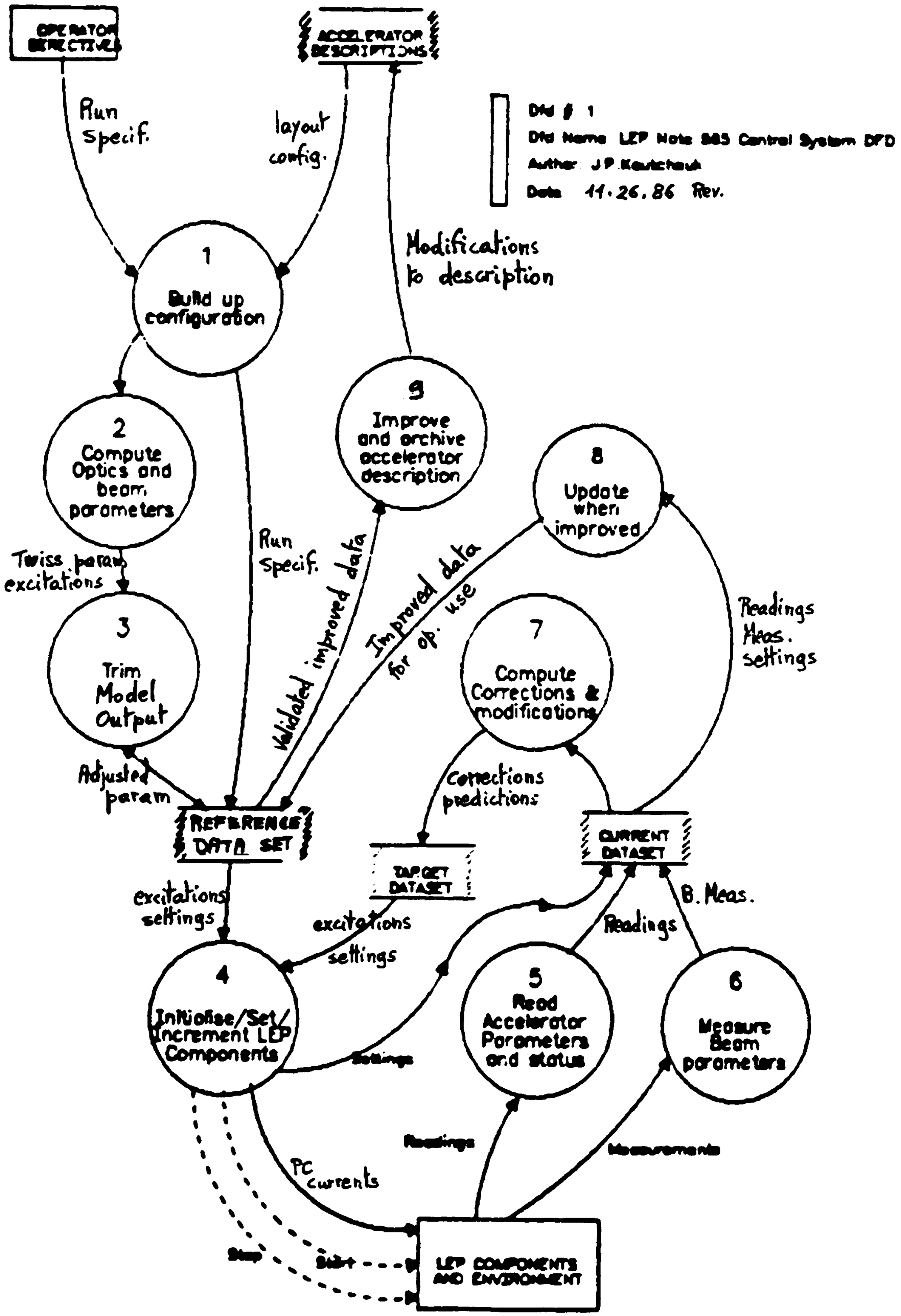
The result of the analysis will be in the form of specifications to a group of programmers who will design, code and test the relative applications.

Structured Analysis is the recommended technique because it will produce documentation in advance of the code and it provides a common language for discussion of the problems. However, it is not imposed a priori. Consequently the AAWG will use these techniques, provided that it considers that they are appropriate to the problem at hand.

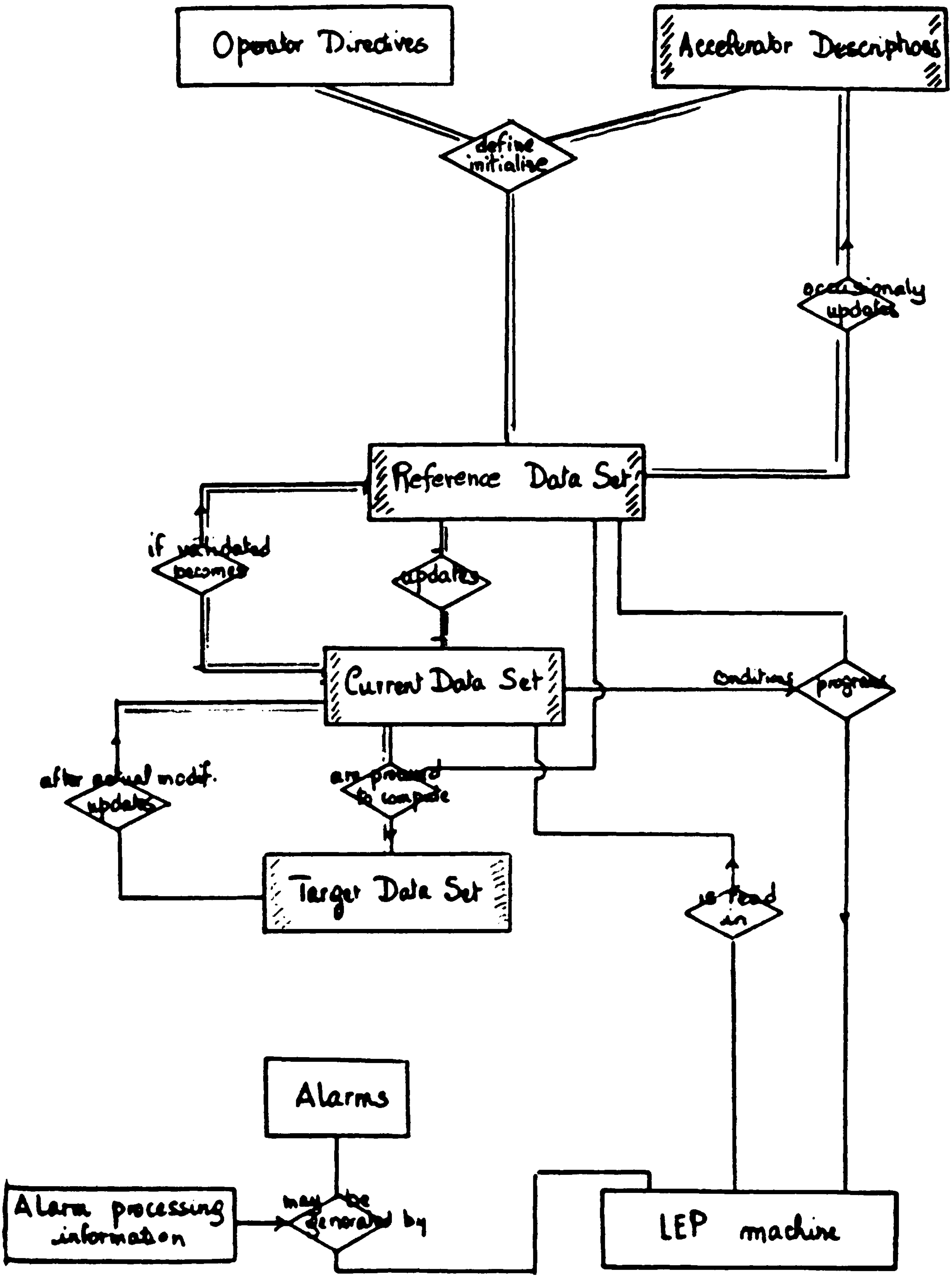
IBM PC_AT computers together with Personal Computer Structured Analysis (PCSA) packages from the SPS and the Yourdon Analyst Toolkit will be made available.

The duration of the mandate to the AAWG is 6 to 8 months, after which the work will be reviewed. Given the limited time and effort, priority will be given to the applications considered as essential for operation and commissioning.

Model of LEP control system



Logical data organisation



Procedures for LEP

Introduction

Processes marked with an asterisk are proposed as being primitives but it should be noted that they still need to have their data flows specified.

1. Build up Configuration

1.1 Collect operator intents
1.2 Extract data from Oracle
1.3 Define ramp (energy vs time)

2. Compute optics and beam parameters

2.1 Compute machine and beam parameters
2.2 Compute optics parameters for ramp and transfer matrices
2.3 Compute strengths for all magnetic elements at all energies
2.4 Pre-define sequence of events

3. Trim Model Output

At this point one can apply fudge factors found in the Accelerator description.

4. Initialise / Set / Increment LEP Components

4.1 Increment/Set magnetic elements •

- Dipoles
- Main Quads
- Transfer lines
- Insertion Quads
- Tilted Quads
- Sextupoles
- Dispersion matchers
- Beam seperators
- Wigglers
- Steering magnets
- Orbit correctors

4.2 Increment/Set collimators, scrapers and stoppers •
4.3 Increment/Set injection bump, kickers •
4.4 Set/Request beam parameters from SPS •
4.5 Increment/Set RF •
4.6 Initialise beam instrumentation •
4.7 Cycle magnetic elements

5. Read Accelerator Parameters and Status (Read hardware)

5.1 Read magnetic elements •

- Dipoles
- Main Quads
- Transfer lines
- Insertion Quads
- Tilted Quads
- Sextupoles
- Dispersion matchers
- Beam seperators
- Wigglers
- Steering magnets
- Orbit correctors

5.2 Read collimators, scrapers and stoppers •
5.3 Read kickers •
5.4 Read beam parameters from SPS •
5.5 Read RF •
5.6 Read beam instrumentation •

6. Measure Beam Parameters

6.1 Twiss Parameters and Phase Advance
6.2 Dispersion
6.3 Chromaticity
6.4 Tune •
6.5 Coupling
6.6 Energy
6.7 Orbit, beam separation
6.8 Luminosity (LEP monitors and experiments' monitors)
6.9 Background (LEP monitors and experiments' monitors)
6.10 Emittances and transverse profiles
6.11 Bunch intensities •
6.12 Bunch length •
6.13 Damping partition numbers
6.14 Synchrotron tune
6.15 Lifetime
6.16 Current, dI/dt •
6.17 Trajectories

7. Compute Corrections and Modifications

7.1 Correct Twiss parameters and phase advance
7.2 Correct dispersion
7.3 Correct chromaticity
7.4 Correct tune •
7.5 Correct coupling
7.6 Correct energy
7.7 Correct orbit, beam separation
7.8 Optimise luminosity (LEP monitors and experiments' monitors)
7.9 Optimise background (LEP monitors and experiments' monitors)
7.10 Optimise emittances and transverse profiles
7.11 Equalise bunch intensities •
7.12 Correct bunch length •
7.13 Correct damping partition numbers
7.14 Correct synchrotron tune
7.15 Correct (Compute) collimator position for aperture
7.16 Make local orbit bump
7.17 Optimise injection bump
7.18 Correct emittance ratios
7.19 Interpolate between programmed and actual machine
7.20 Analyse measurements for source of defects
7.21 Signal analysis

8. Update Reference Dataset

Transfer proven corrections to Reference dataset from the Current dataset.

9. Improve and Archive Accelerator Description

This means improving the model, fudge factors and correcting calibration factors etc. It is a permanent update to the system.

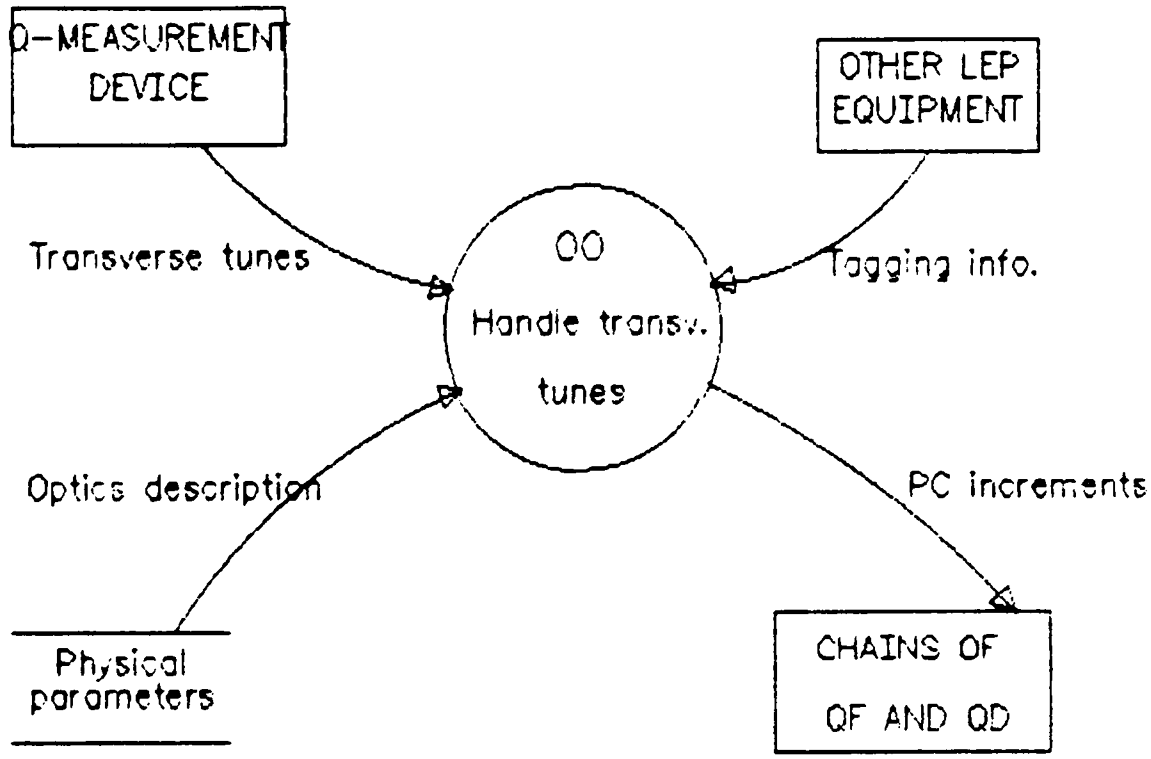
N.B. This list is not exhaustive

Specific analysis of measured data
Interpolate
Distribution of information to physicists
Procedures associated with collective effects
⋮

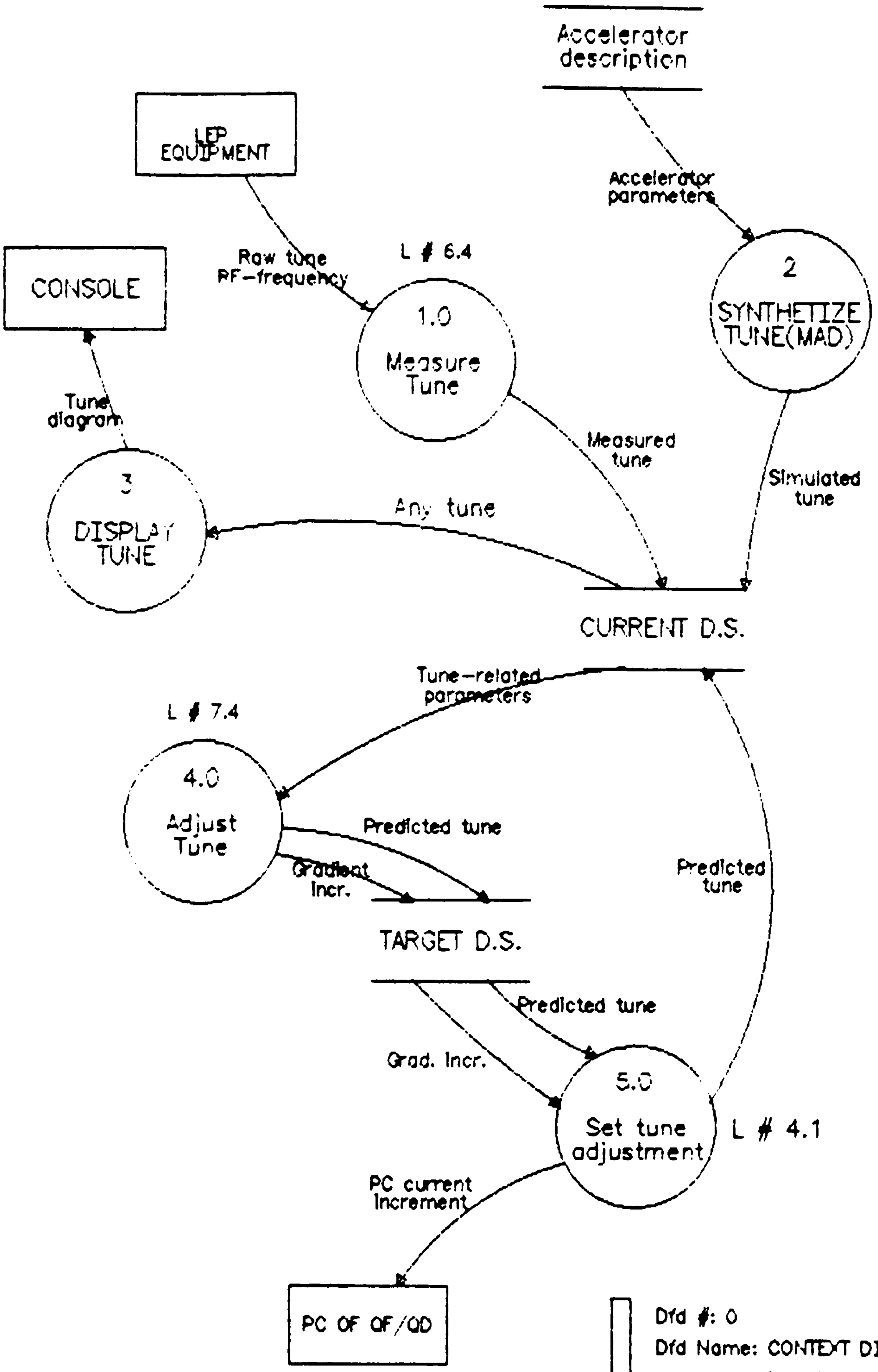
and will be completed, following the progress made by both LEPCC and AAWG.

Ex. 1 Transverse Tunes

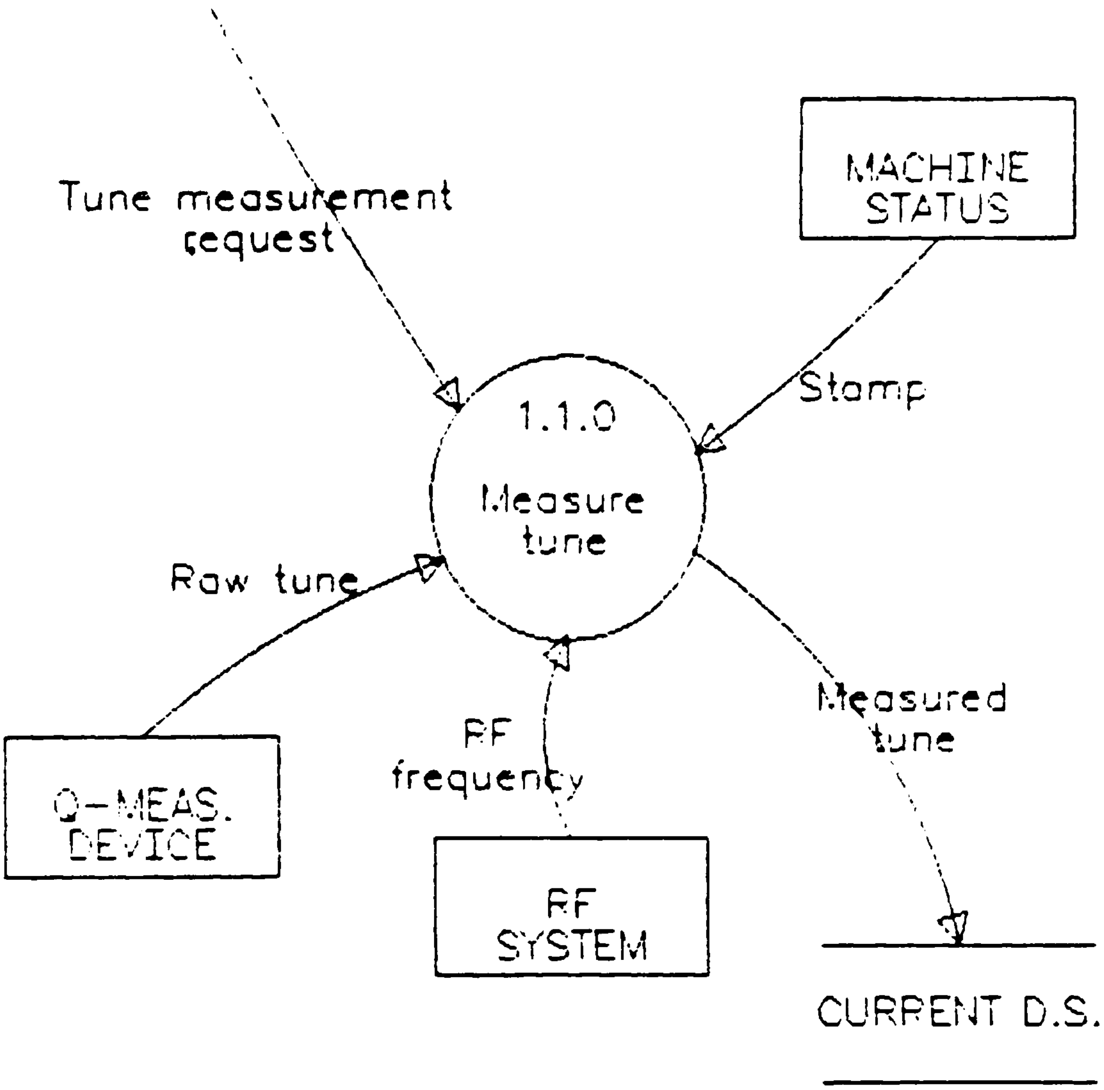
Complete series of Data Flow Diagrams
and list of Dictionary Entries



Dfd #: 00
Dfd Name: GENERAL TUNE CD
Author: GG/JPK/RW
Date: 01/08/87



Dfd #: 0
Dfd Name: CONTEXT DIAGRAM
Author: GG/JPK/RW
Date: 01/08/87

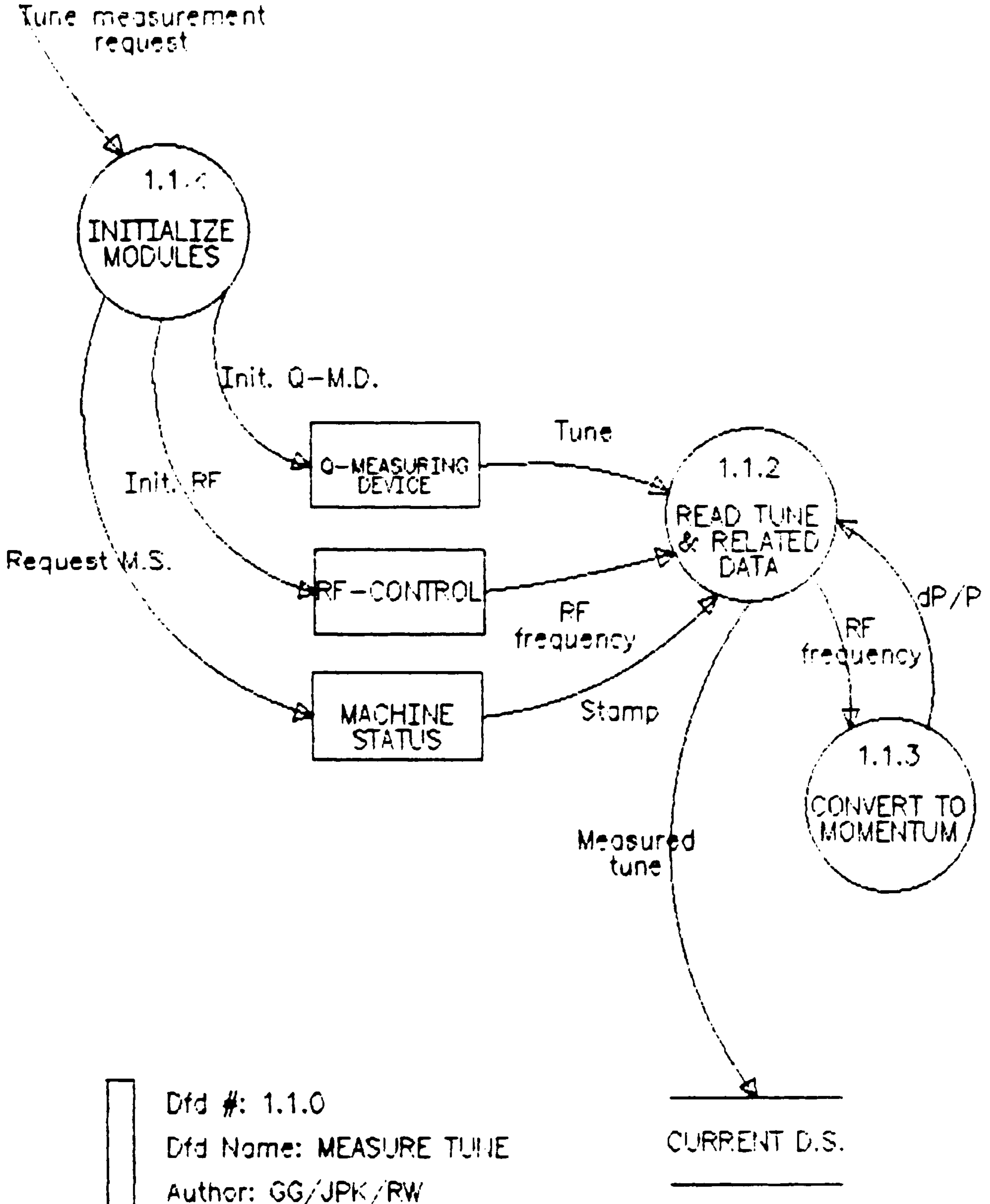


Dfd #: 1.0

Dfd Name: Measure Tune

Author: GG/JPK/RW

Date: 01/16/87

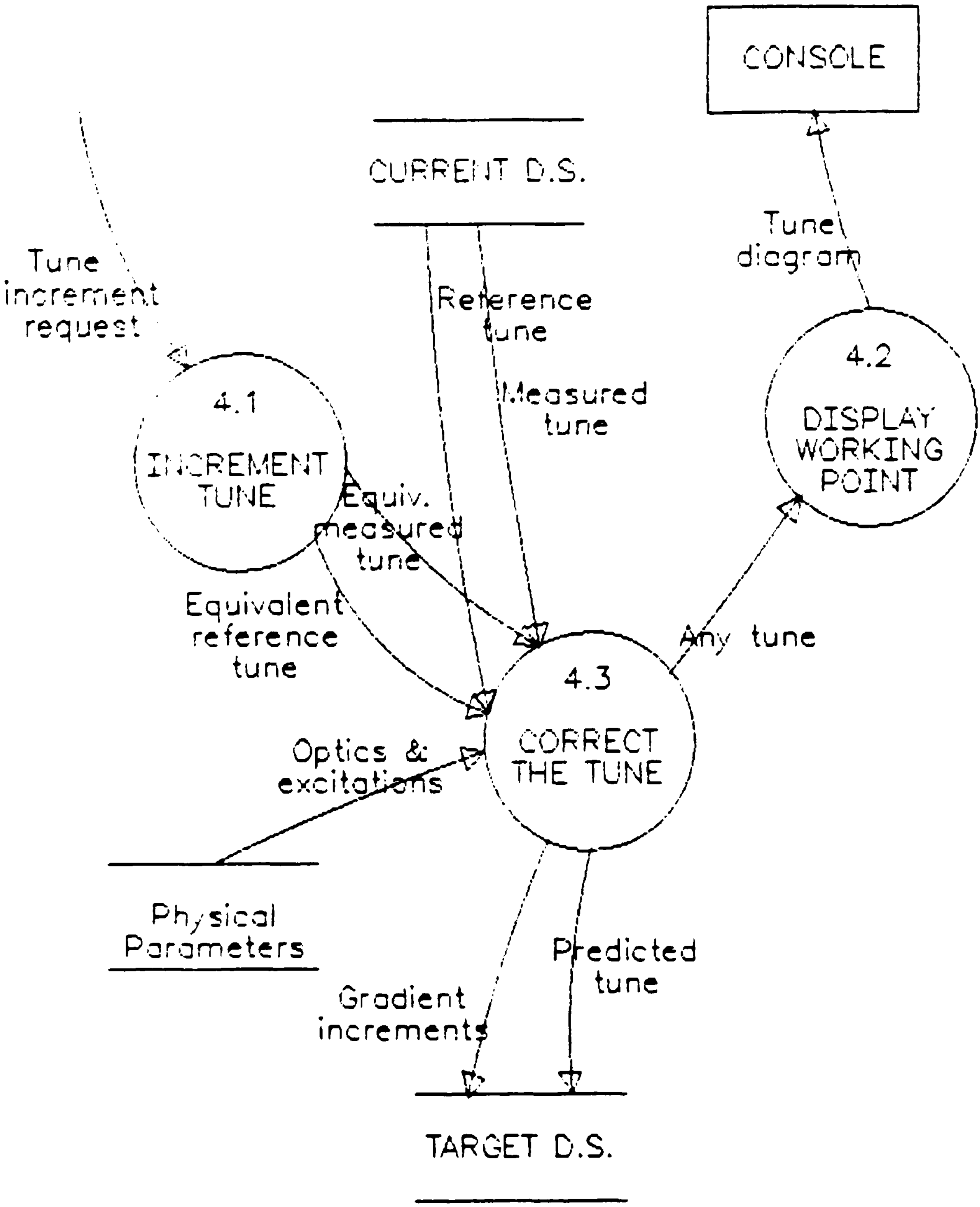


Dfd #: 1.1.0

Dfd Name: MEASURE TUNE

Author: GG/JPK/RW

Date: 01/12/87

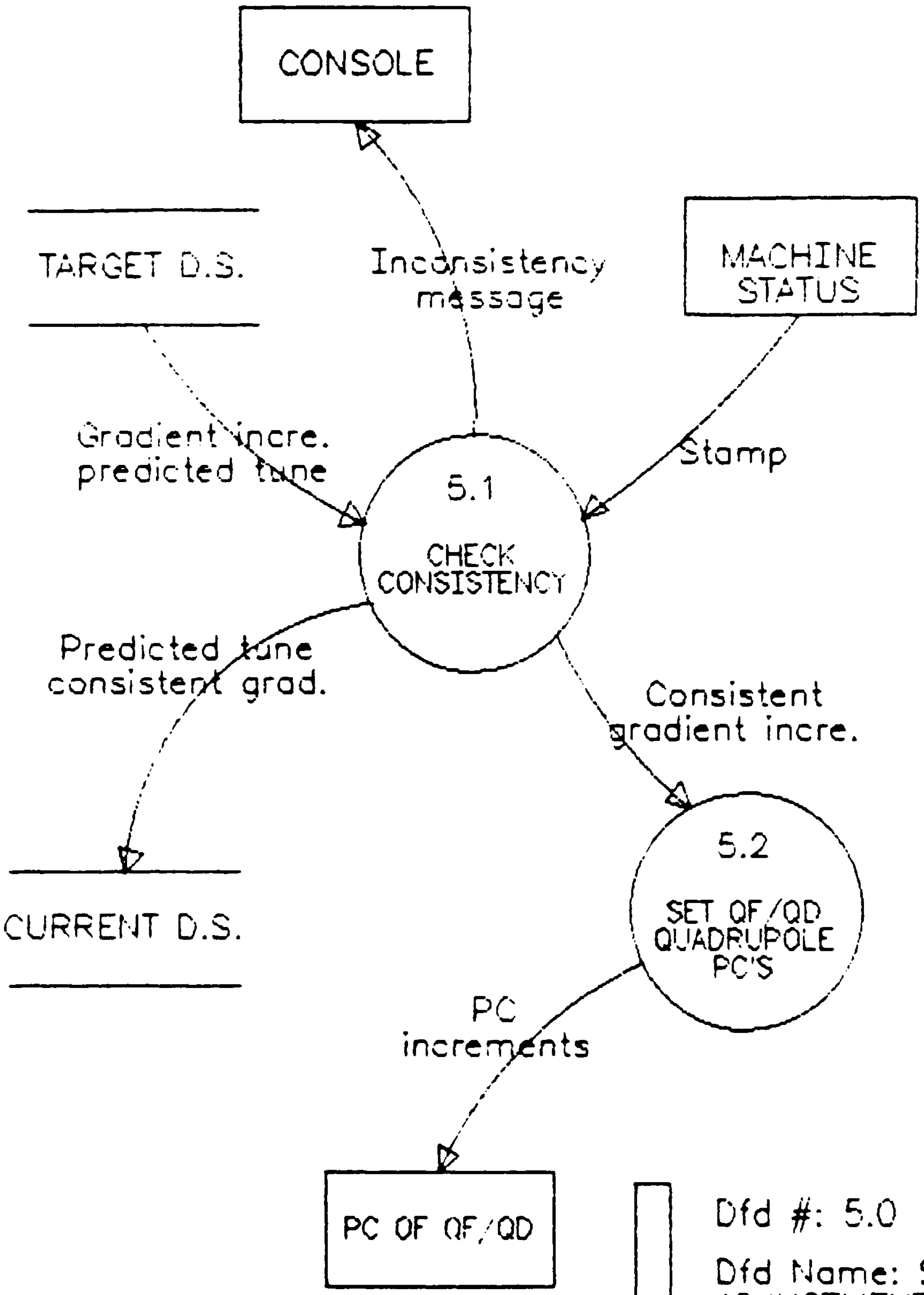


Dfd #: 4.0

Dfd Name: Adjust Tune

Author: GG/JPK/RW

Date: 01/15/87



Dfd #: 5.0

Dfd Name: SET TUNE ADJUSTMENT

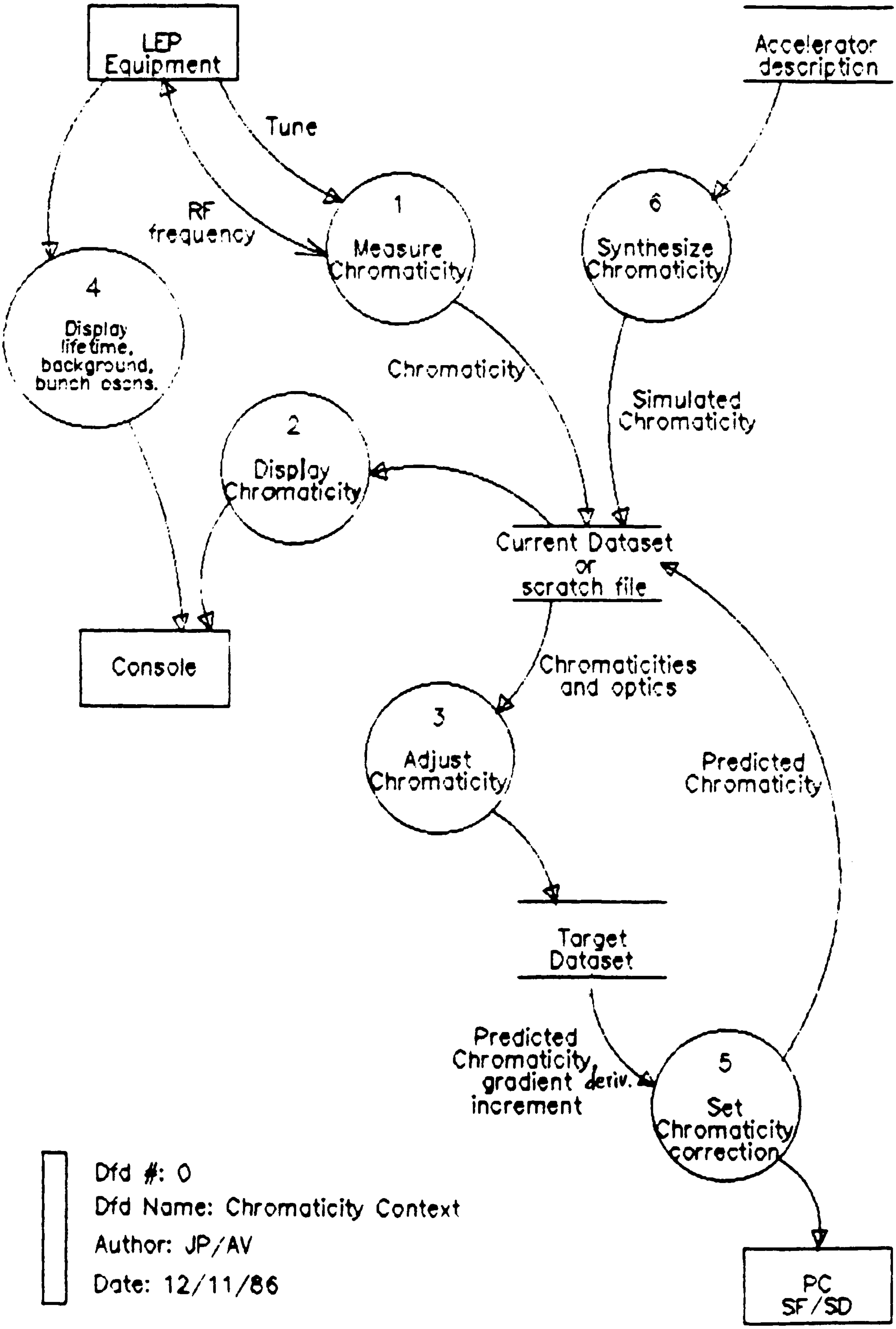
Author: GG/JPK/RW

Date: 01/16/87

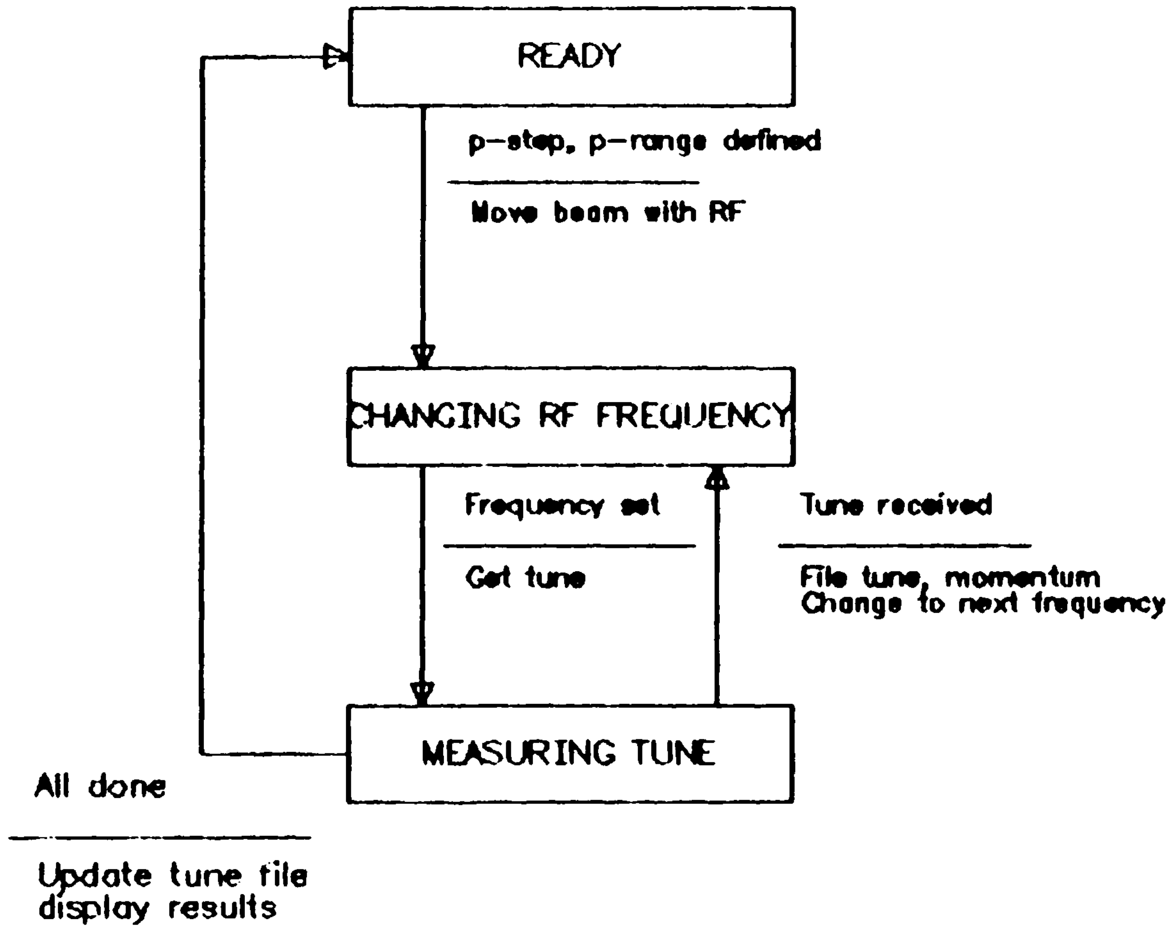
UNDEFINED DICTIONARY ENTRIES
============================
01/26/87 17:26:54

ACCELERATOR DESCRIPTION
ACCELERATOR PARAMETERS
ANY TUNE
CONSISTENT GRADIENT INCRE.
CURRENT D.S.
DP/P
EQUIV. MEASURED TUNE
EQUIVALENT REFERENCE TUNE
GRAD. INCR.
GRADIENT INCR.
GRADIENT INCRE. PREDICTED TUNE
GRADIENT INCREMENTS
INCONSISTENCY MESSAGE
INIT. O-M.D.
INIT. RF
MEASURED TUNE
OPTICS & EXCITATIONS
PC CURRENT INCREMENT
PC INCREMENTS
PHYSICAL PARAMETERS
PREDICTED TUNE
PREDICTED TUNE CONSISTENT GRAD.
RAW TUNE
RAW TUNE RF-FREQUENCY
REFERENCE TUNE
REQUEST M.S.
RF FREQUENCY
SIMULATED TUNE
STAMP
TARGET D.S.
TUNE
TUNE INCREMENT REQUEST
TUNE DIAGRAM
TUNE MEASUREMENT REQUEST
TUNE-RELATED PARAMETERS

Ex. 2 Chromaticity
Example of State Transition Diagram



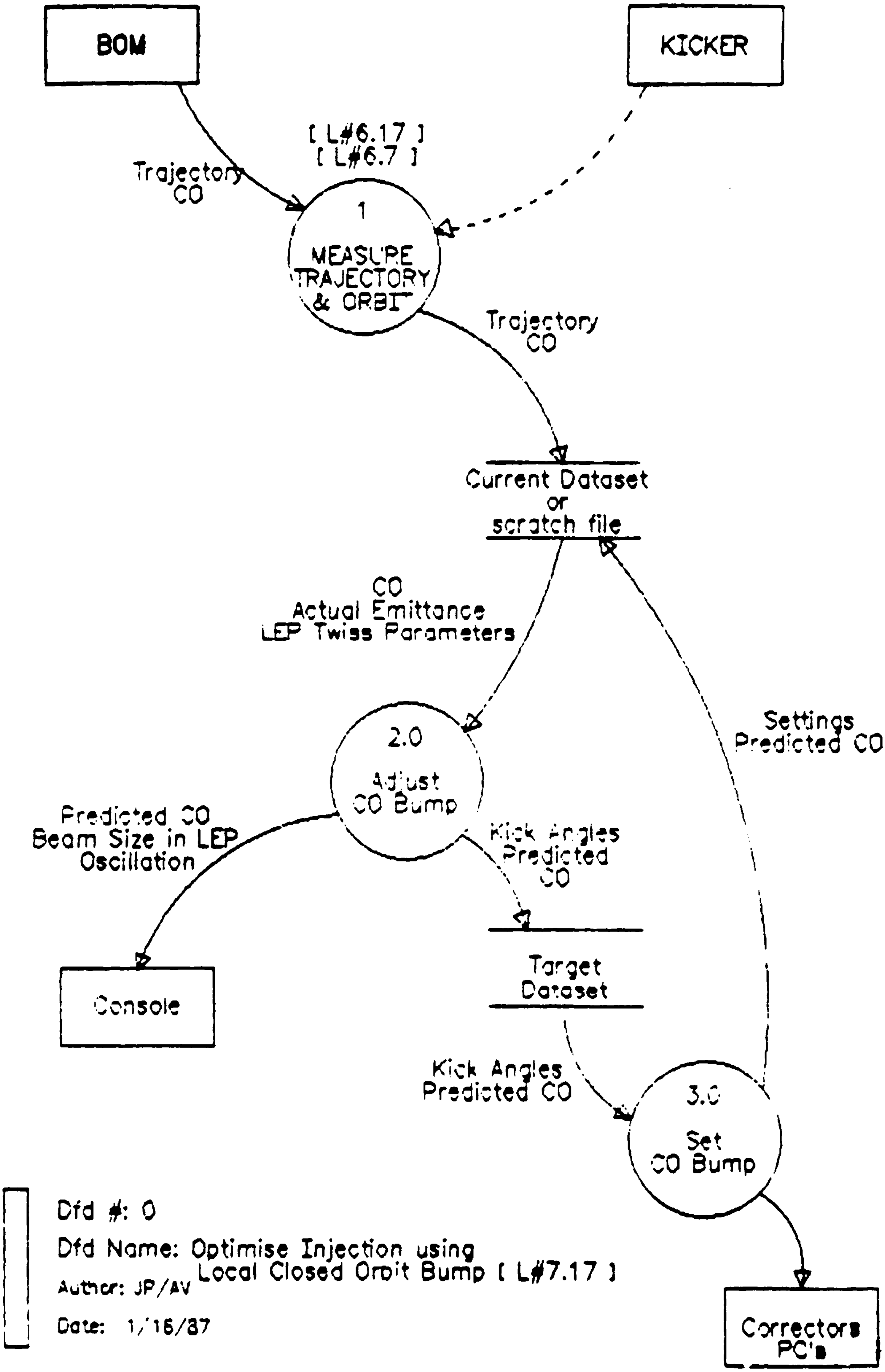
Dfd #: 0
Dfd Name: Chromaticity Context
Author: JP/AV
Date: 12/11/86



State Transition Diagram

Ex. 3 Injection Optimisation

Example of Dictionary Content : entries description



ABBREVIATED DICTIONARY LISTING
================================
01/19/87 12:29:22

ACTUAL EMITTANCE
COMMENTS: Measured emittances of LEP Beam

BEAM SIZE IN LEP
COMMENTS: Beam size at 4 sigma computed with actual emittances in both planes at each point in the bump region.

CO
COMMENTS: One takes "Measured Average Positions" from the Current Dataset (No smoothing or corrections for energy effects).

CO ACTUAL EMITTANCE LEP TWISS PARAMETERS
COMPOSITION: CO + ACTUAL EMITTANCE + LEP TWISS PARAMETERS
COMMENTS:

CO BUMP
COMMENTS: CLOSED ORBIT BUMP

CORRECTOR PC INCREMENTS
COMMENTS: Values sent to PC's

CURRENT DATASET
COMPOSITION: LEP TWISS PARAMETERS + CO + TRAJECTORY + KICKER SETTINGS + PC CURRENTS + VACUUM CHAMBER POSITIONS + ACTUAL EMITTANCE + KICK ANGLES + PREDICTED CO + OSCILLATIONS
COMMENTS:

CURRENT DATASET OR SCRATCH FILE
COMPOSITION: CURRENT DATASET + SCRATCH FILE
COMMENTS:

INJECTION BUMP
COMMENTS: Displacement generated by the injection kickers

KICK ANGLES
COMMENTS: Angles required from the correctors generating the closed orbit bump which minimises the injection oscillation

---

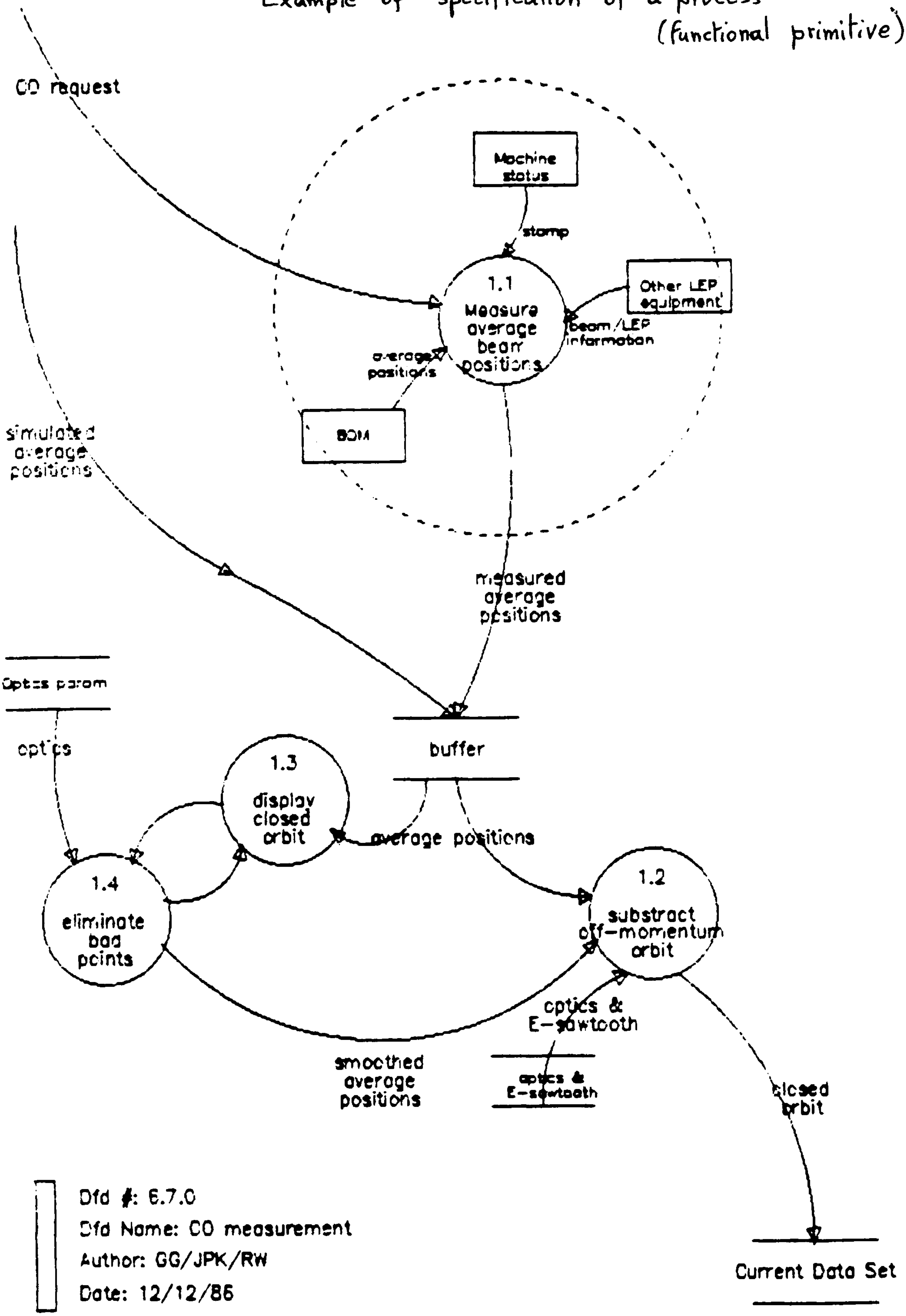
KICK ANGLES CO BUMP
COMPOSITION: KICK ANGLES + CO BUMP
COMMENTS:

KICK ANGLES PC CURRENTS PREDICTED CO
COMPOSITION: KICK ANGLES + PC CURRENTS + PREDICTED CO
COMMENTS:

KICK ANGLES PREDICTED CO
COMPOSITION: KICK ANGLES + PREDICTED CO
COMMENTS:

Ex. 4 Closed Orbit

Example of specification of a process (functional primitive)



MINISPEC FOR PROCESS 1.2
"SUBTRACT OFF-MOMENTUM ORBIT"
****

```
EXTRACT <dP/P> FROM HEADER;
if <dP/P>RF NOT SPECIFIED
   then
   begin
   <dP/P>=SUM(X)/SUM(Dx) FOR ALL NON-FAULTY MONITORS;
   SET <dP/P>AV IN THE HEADER;
   end;
for ALL NON-FAULTY MONITORS do
   begin
   Xp=X-Dx*<dP/P>AV;
   Zp=Z;
   end;
SET FLAG TO "CORRECTED FOR AVERAGE dP/P" IN HEADER;
if E-SAWTOOTH CORRECTION REQUESTED
   then
   for ALL NON-FAULTY MONITORS do
      begin
      Xco=Xp-Dx*dP/P(s);
      Zco=Zp;
   end;
SET FLAG TO "CORRECTED FOR E-SAWTOOTH";
end.
```