INSTITUTO DE PESQUISAS ESPACIAIS
MINISTÉRIO DA CIÊNCIA E TECNOLOGIA

AUTORIZAÇÃO PARA PUBLICAÇÃO
AUTHORIZATION FOR PUBLICATION

PALAVRAS CHAVES/KEY WORDS
CONFORMANCE TEST/SFDU/CCSDS/STANDARD DATA INTERCHANGE STRUCTURES/SPACE DATA SYSTEMS

AUTORIZADA POR/AUTHORIZED BY
Clóvis Solano Pereira
Dir. Eng.e Tec. Espacial

AUTOR RESPONSAVEL / RESPONSIBLE AUTHOR
Eduardo W. Bergamini

DISTRIBUIÇÃO/DISTRIBUTION
- [ ] INTERNA / INTERNAL
- [x] EXTERNA / EXTERNAL
- [ ] RESTRITA / RESTRICTED

REVISADA POR / REVISED BY
Juan Suñe Perez

CDU/UDC
681.324

DATA / DATE
Julho 1989

PUBLICAÇÃO Nº / PUBLICATION NO
INPE-4868-PRE/1479

ORIGEM / ORIGIN
DEL/VSS

PROJETO / PROJECT
ENCOMP

Nº DE PAG. / NO OF PAGES
76

ULTIMA PAG. / LAST PAGE
66

VERSÃO / VERSION
01

Nº DE MAPAS / NO OF MAPS

TÍTULO/TITLE

CONFORMANCE TEST CONCEPTS FOR DATA ENTITIES BASED ON SFDU STRUCTURE & CONSTRUCTION RULES

AUTORES/AUTHORSHIP
Eduardo Whitaker Bergamini

RESUMO - NOTAS / ABSTRACT - NOTES

The concepts introduced and, to some extent, developed in this work are intended to form a basis for what could, possibly, become a methodology for Conformance Test of Data Entities observing the CCSDS Protocol for SFDU Structure & Construction Rules. These rules form the CCSDS recommendations for Standard Data Interchange Structures, under development by its Panel 2 working group.

OBSERVAÇÕES / REMARKS

This concept paper was prepared in response to an action item established by CCSDS Panel 2 (on Standard Data Interchange Structures) and was presented at its 15th Plenary Workshop held at Grumman/PSC Facilities, Reston, VA, USA from May 08th to 12th, 1989.

INSTITUTO DE PESQUISAS ESPACIAIS

AUTORIZAÇÃO PARA PUBLICAÇÃO
AUTHORIZATION FOR PUBLICATION

PALAVRAS CHAVES/KEY WORDS
CONFORMANCE TEST/SFDU/CCSDS/STANDARD DATA INTERCHANGE STRUCTURES/SPACE DATA SYSTEMS

AUTORIZADA POR/AUTHORIZED BY
Clóvis Solano Pereira
Dir. Eng.e Tec. Espacial

AUTOR RESPONSÁVEL / RESPONSIBLE AUTHOR
Eduardo W. Bergamini

DISTRIBUIÇÃO/DISTRIBUTION
[ ] INTERNA / INTERNAL
[X] EXTERNA / EXTERNAL
[ ] RESTRITA / RESTRICTED

REVISADA POR / REVISED BY
Juan Suñe Perez

CDU/UDC: 681.324

DATA / DATE: Julho 1989

PUBLICAÇÃO Nº / PUBLICATION NO: INPE-4868-PRE/1479

ORIGEM / ORIGIN: DEL/VSS

PROJETO / PROJECT: ENCOMP

Nº DE PAG. / NO OF PAGES: 76

ULTIMA PAG. / LAST PAGE: 66

VERSÃO / VERSION: 01

Nº DE MAPAS / NO OF MAPS:

TÍTULO/TITLE

CONFORMANCE TEST CONCEPTS FOR DATA ENTITIES BASED ON SFDU STRUCTURE & CONSTRUCTION RULES

AUTORES/AUTHORSHIP
Eduardo Whitaker Bergamini

RESUMO-NOTAS/ABSTRACT-NOTES

The concepts introduced and, to some extent, developed in this work are intended to form a basis for what could, possibly, become a methodology for Conformance Test of Data Entities observing the CCSDS Protocol for SFDU Structure & Construction Rules. These rules form the CCSDS recommendations for Standard Data Interchange Structures, under development by its Panel 2 working group.

OBSERVAÇÕES/REMARKS

This concept paper was prepared in response to an action item established by CCSDS Panel 2 (on Standard Data Interchange Structures) and was presented at its 15th Plenary Workshop held at Grumman/PSC Facilities, Reston, VA, USA from May 08th to 12th, 1989.

RESUMO

Os conceitos introduzidos e, em certa extensão, desenvolvidos neste trabalho destinam-se a formar uma base para o que poderia, possivelmente, se tornar uma metodologia para Teste de Conformidade de Entidades de Dados que observam o Protocolo CCSDS para as Regras de Estruturação e construção de SFDUs ("Standard Formatted Data Units"). Estas regras formam a espinha dorsal para as recomendações do CCSDS para a Troca Padronizada de Dados Estruturados, sendo desenvolvidas pelo grupo de trabalho do seu Painel 2 .

ACKKNOWLEDGEMENTS

The author of this paper wishes to thank all those who helped him in the preparation of this work, with special mentioning for: Ms. Natalia Santos Gomes for her extensive support in the word processing, typing and copying and Ms. Ana Lucia S. de Castro for her extensive drafting work of the figures. Thanks are also extended to Ms.Ana Maria da Silva Vadô for the final, partial typing of this paper as an internal publishing, in INPE, as a technical report.

CONTENTS

LIST OF FIGURES

1. INTRODUCTION

The **concepts** for Conformance Test of data introduced in this work are intended to form, partially, the **basis for a methodology.** In this context a methodology would be characterized by a set of concepts and structures that would be defined to permit, hopefully, an uniform approach to conformance test. This approach would desired y lead to unique or, at least, equivalent results concerning the compatibility evaluation of **data products** which are supposed to comply with **structure and construction rules requirements** defined by the **CCSDS Protocol** and destinated to form Standard Data Interchange Structures.

Conformance Test Systems could be devised as "on line" or "off line" tools, depending on the specific application. A Conformance Test System could be incorporated to a typical Data Capture System (Ref. [3]), as illustrated in Figure 1.1.

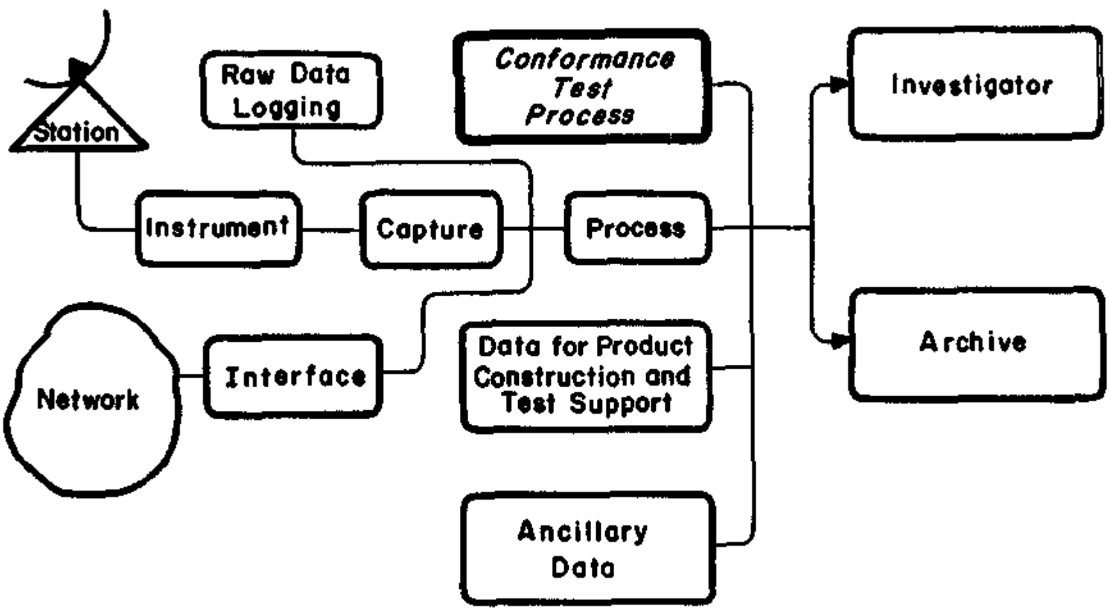

Figure 1.1. Typical Data Capture System.

REFERENCES

[1] "Standard Formatted Data Units - Structure and Construction Rules", Recommendation CCSDS 620.0-B-1, Blue Book, Consultative Committee for Space Data Systems, February 1988.

[2] "Standard Formatted Data Units - Structure and Construction Rules: Extensions", Recommendation CCSDS 620.1-W-3a, White Book, Consultative Committee for Space Data Systems, January 1989.

[3] "Standard Formatted Data Units - Product Aggregation Aspects", Recommendation CCSDS 610.1-G-2a, Green Book, Issue 2a, Consultative Committee for Space Data Systems, January 1989 (Draft of February 1989).

[4] "Draft Recommendation X.290-OSI Conformance Testing Methodology and Framework for CCITT Applications (Final Version)", CCITT Study Group VII - Contribution 257 (COM VII - 257 - E), December 1987.

Although this work is not intended to develop concepts associated to real time constraints of conformance test systems, it can be considered that the more precise and compact a methodology for conformance test is devised, the more time independent will possibly be its applications; simply because different levels of complexity could be devised for convenient implementation in real or non-real time systems.

Concepts and Terminology proposed for Conformance Test are introduced in **Chapter** 2 of this work. More specifically, a basic terminology for conformance test defined, mainly, by ISO/CCITT (Ref. [4]) is presented (Sec. 2.2) after being condensed and adapted to the context of this work, whenever it was judged applicable. A concept and a formulation leading to conformance test and report tables are also introduced (Sec. 2.3). Basic Schemes for testing Events and for executing the flow of resulting Test Steps are also introduced (Sec. 2.4).

A modelling concept for Conformance Test is proposed in **Chapter** 3. Inspired on the requirements (Ref. [3]) for SFDU structuring and construction, a fairly extensive proposal for breakingdown and testing the rules applied to construct a data product is also given (Sec. 3.3) departing from a basic model of CONFORMANCE TESTING MACHINE (Sec. 3.1). Still in Chapter 3, the characterization of Data Items is commented with a brief example (Sec. 3.3). Another topic also covered in the same chapter, concerns a typical reporting structure that could be expected (Sec. 3.4) from the execution of the Conformance test scheme introduced in Section 3.2.

In **Chapter** 4, under the denomination of "Case Study", a simulation resulting from the application of the Conformance test scheme (introduced in Section 3.2) on a data entity, which is composed with aggregation by SEQUENCE, is given, as far as the resulting structure of the conformance test reports is concerned.

Comments and suggestions for future developments of the conformance test concept are given in the conclusive **Chapter (5)**.

2. CONCEPTS AND TERMINOLOGY

2.1. Introduction

The **CCSDS** concept for standard data interchange and storage is based on well defined structure and construction rules inspired on the TLV Objept concept [1]. The CCSDS data entity standard resulting from the TLV concept is the SFDU. Therefore, data entities structured and constructed in accordance to the SFDU concept are said to observe the **CCSDS Protocol**, defined for this purpose.

CCSDS recognizes that other types of protocol are in use for data interchange and storage in space data systems of many space agencies. The data entities so defined are said to observe **Non - CCSDS Protocol** in their structure and construction rules.

A **Data Entity** is a logical collection of data that, for some reason, has a separate and distinct existence and objective. There are several types of data entities ([2] and [3]), like: **Data Unit**, a **Data Product**, a **Data Object** or, even, a **Data Element.** In the context of this work the Data Entity is the definition used for data to be interchanged or stored in accordance to a specific protocol. Therefore, it is understood that, depending on the context, the data under consideration is a Data Unit or a Data Product or other.

The concept of **Data Aggregation** plays a fundamental role to formalize the definition of new data entities to be constructed from the aggregation or composition of other data entities. Of course, the CCSDS protocol for data entities permit the aggregation of other data entities which observe the same protocol [3]. Furthermore, the CCSDS protocol estabilishes sufficient (ecumenical) rules that permit also the aggregation of CCSDS and Non-CCSDS protocol data entities, provided that the resulting data entity observes the CCSDS protocol. The CCSDS estabilishes four basic techniques for data aggregation [3]:

- 1. **Aggregation by ENVELOPE;**
- 2. **Aggregation by FLAG;**
- 3. **Aggregation by SEQUENCE;**
- 4. **Aggregation by REFERENCE.**

The compliance of a data entity with the CCSDS protocol should be obtained with the application of tools that can execute **Conformance Testing** on the mentioned data entity. The conformance testing shall be based on the requirements that must be observed for implementing a data entity which shall comply with CCSDS protocol. These requirements are strictly related to the structure and construction rules for data interchange and storage, as recommended by CCSDS in Refs. [1] and [2] and explained in Ref. [3]. The following sections are dedicated to the formalization of the basic concepts of Conformance Test to be applied in the context of Data Entities which observe the CCSDS protocol.

2.2. Conformance Test and Report

Some basic concepts and terminology, inspired on Ref. [4], are defined to structure the concept of **Conformance Test** and **Conformance Report (or both)** of data entities based on CCSDS protocol.

If a **Test Event** is defined as an indivisible unit of test specification, a **Test Step** is constructed from an ordered set of Test Events. An **Outcome** is defined as the **observable result** of a Test Event. A **Test Case** is composed of an **hierarchical structure** of Testing Steps that define a **complete** set of actions required to achieve an objective, specified by a **Test Purpose.** A set of Testing Cases observing a logical ordering is denominated a **Test Group.** A hierarchical structure with a **complete** set of Test Groups is defined as a **Test Suite.** The **Verdict** of a Test is the final **statement** resulting from an Outcome or a Set of Outcomes, depending on the scope of the Test. The **set of Test Cases that are essential** in order to achieve the Test Purpose and assign Verdict statements to the possible Outcomes is defined as a **Test Body.**
The **Statement of a VERDICT** can be expressed by one of the following three basic assignments:

1. **PASS:**

   A verdict given when the observed outcome satisfies the test purpose and is valid with respect to the relevant recommendations;

2. **FAIL:**

   A verdict given when the observed outcome is syntatically or semantically invalid or inopportune with respect to the relevant recommendations;

3. **INCONCLUSIVE:**

A verdict given when the observed outcome is valid with respect to the relevant recommendations but prevents the test purpose from being accomplished.

The **Statement of an Event OUTCOME** can be expressed by one of the following three basic assignements:

1. **VALID:**

   A Test Event is Valid when it is allowed by the protocol recommendation, being both semantically and syntactically correct and ocurring in an allowed context of observed outcome;

2. **INVALID:**

   A Test Event which is not semantically and/or syntactically valid by the protocol recommendation;

3. **INNOPORTUNE:**

   A Test Event which, although syntactically and semantically correct, produces an observed Outcome which is not allowed, at that point.

Test Cases, Groups, Suites and Bodies can be **defined** in three different forms:

1. **GENERIC**

Observing a general specification of the test purposes with all predictable steps, cases or groups to cover the pertinent testing events;

2. **ABSTRACT:**

Particular, formal specification of the test purpose (s);

3. **EXECUTABLE:**

Real implementation of an abstract test specification.

The following three definitions are considered fundamental for characterizing the execution and recording of results obtained with conformance testing practice:

CONFORMANCE TEST (CT):

A Sequence of testing steps concatenated in cases or groups , that, when applied to process a specific data entity permit to assess the degree of its conformance with a reference data entity or with a set of requirements and purposes with which it must comply.

CONFORMANCE TEST REPORT (CTR):

The sufficient structured, logged information, stored or documented, which permits the verification and assessment of the results obtained with the application of a conformance test to a data entity, including the verification of: all executable test cases and, the testing events or purposes, their observable and foreseen data outocomes and related information and by means of one or more logged verdicts which testity the compliance of the same data entity with pre-established protocol for the entity structure and construction rules.

CONFORMANCE TEST & REPORT (CT&R):
The combined act of applying a conformance test to a product and reporting its results.

The Conformance Testing **Requirements** can be **classified** in **three categories**:

1. **MANDATORY:**
   The requirements that **must** be observed in **all** cases;

2. **CONDITIONAL:**
   The requirements that have to be observed **if** the conditions set out in the recommendation apply;

3. **OPTIONAL:**
   The requirements that can be **selected** to satisfy the implementation, provided that when applicable they are observed.

The Conformance Testing **requirements** can be **stated in two different ways:**

1. **POSITIVELY:**
   The requirement indicate what shall be done;

2. **NEGATIVELY:**
   The requirement indicate what shall **not** be done.

The **Analysis of Conformance Test (CT)** based on Conformance Test Report (CTR) should rely on **results** which, desirebly, observe the following **predicates:**

1. **REPEATEABILITY:**
   Whenever a CT is performed, the results should be the same;

2. **COMPARABILITY:**
   Whenever a CT is performed in **different** environments (supplier, user or "test house") the results should give **comparable reports** with **equivalent results.**

3. **AUDITABILITY:**
   Whenever a CT is performed it may be expected that, for legal reasons, a consistent and acceptable review and assessment of the logging, obtained from the observed inputs, events, outcomes and verdicts may be necessary to

make sure that all procedures of the Conformance Test Suite have been correctly followed.

2.3 - Conformance Test Report Table

It seems to be convenient to **report** the conformance test of a data entity or of any of its data items of interest by some standard description which may facilitate the analysis, review or assessment of the results and permit its standard logging, as an additional result.

One of the main benefits of a standardized conformance test reporting practice would be the commonality of interpretation of results among the members or control authorities of the user community who would need it to evaluate or to have assurance on the data products being exchanged among different parties (eg. different agencies, disciplinary areas, projects, etc.).

The **familiarity** with a common (standard) Conformance Test report procedure would possibly encourage the exchange of data products, while promoting increased confidence between the data supplier and the data user parties.

Furthermore, standard conformance test report procedures would also facilitate and encourage the development of automated Conformance Test procedures or, even, systems. Predicates like: repeateability, comparability and auditability would be directly benefited in the analysis process of Conformance Tests results.

A report scheme denominated by **CONFORMANCE TEST REPORT TABLE (CTRT)** is proposed to be used as an output of the conformance test applied to a specific **DATA ITEM.** The basic template for the Conformance Test Report Table is represented in Figure 2.1. A **Data Item** is defined as any data entity passive of conformance testing. If a TLV object must be submitted to conformance testing procedures, its basic components: T, L and V fields should also be passive of conformance testing, individually, as **separated** Data Items.

It can be observed in Figure 2.1 that for each **Test EVENT (TEE)** there is a set of **eight REPORT Items** associated to it. Each of these REPORT items have the following definitions:

- **Instance of EVENT (INE):**

  Specifies the actual contents of the Event or, alternatively, their addressing, depending on the convenience of each representation if any. The following convention is proposed to assign a value to INE:

  - **INE = C/ <actual content>;**
  - **INE = P/ <pointer to actual content>;**
  - **INE = N/A (Not Applicable).**

- **Observed Event OUTCOME (OBO):**

  Specifies logical (binary) information stating the **actual, observed** OUTCOME of the conformance test applied to the corresponding Test EVENT, with the following alternatives:

  - **OBO = (00)** if **VALID (VA);**
  - **OBO = (01)** if **UNVALID (UV);**
  - **OBO = (10)** if **INNOPORTUNE (IP);**
  - **OBO = (11)** if **NOT APPLICABLE (NA).**

- **Expected Event OUTCOME (EXO):**

  Specifies logical (binary) information stating the **expected** OUTCOME of the conformance test applied to the corresponding Test EVENT, with the following alternatives:

- **EXO** = **(00)** if **VALID (VA)**;
- **EXO** = **(01)** if **UNVALID (UV)**;
- **EXO** = **(10)** if **INNOPORTUNE (IP)**;
- **EXO** = **(11)** if **NOT APPLICABLE (NA)**.

- **Overall Event OUTCOME (OVO)**:

  Specifies logical (binary) information stating a **final, conclusive** (overall) result for the event OUTCOME of the conformance test applied to the corresponding Test EVENT, with the following alternative meanings:

  - **OVO** = **(00)** if **VALID (VA)**;
  - **OVO** = **(01)** if **UNVALID (UV)**;
  - **OVO** = **(10)** if **INNOPORTUNE (IP)**;
  - **OVO** = **(11)** if **NOT APPLICABLE (NA)**.

The **Overall Event OUTCOME (OVO)** of a Conformance Test applied to a Test EVENT is defined from the observed (OBO) **and** from the Expected (EXO) Event OUTCOMES and can be stated by a (Boolean logic) Karnaugh Map, as represented in Figure 2.2.

CONFORMANCE TEST REPORT TABLE									
Data Item Under Conformance Test :<DAI>									
Pointer to Originating Report: <P O R>					Pointer to Following Report:< P F R>				
Part No.	REPORT Item / Test EVENT	Instance of EVENT	Observed Event OUTCOME	Expected Event OUTCOME	Overall Event OUTCOME	Local Conformance VERDICT	External Conformance VERDICT	Overall Conformance VERDICT	Obser - vation
1	<TEE$_1$>	<INE$_1$>	<OBO$_1$>	<EXO$_1$>	<OVO$_1$>	<LOV$_1$>	<EXV$_1$>	<OVV$_1$>	<OBS$_1$>
2	<TEE$_2$>	<INE$_2$>	<OBO$_2$>	<EXO$_2$>	<OVO$_2$>	<LOV$_2$>	<EXV$_2$>	<OVV$_2$>	<OBS$_2$>
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• •	• •
n-1	<TEE$_{n-1}$>	<INE$_{n-1}$>	<OBO$_{n-1}$>	<EXO$_{n-1}$>	<OVO$_{n-1}$>	<LO			
n	<TEE$_n$> <DAI>	<INE$_n$>	<OBO$_n$>	<EXO$_n$>					

Figure 2.1 General Format of the CONFORMANCE TEST REPORT TABLE (CTRT).

EXO \ OBO	00 (VA)	01 (UV)	11 (NA)	10 (IP)
00 (VA)	00 (VA)	01 (UV)	00 (VA)	10 (IP)
01 (UV)	01 (UV)	01 (UV)	01 (UV)	01 (UV)
11 (NA)	00 (VA)	01 (UV)	11 (NA)	10 (IP)
10 (IP)	10 (IP)	01 (UV)	10 (IP)	10 (IP)

Figure 2.2. Karnaugh Map of Overall Event OUTCOME (OVO) States.

- **Local Conformance VERDICT (LOV):**
  Specifies logical (binary) information stating the VERDICT affecting the Test EVENT **internal** to the event data item under conformance test. The following values can be assigned to the VERDICT:

- **LOV = (00)** if **PASS (PA)**;
- **LOV = (01)** if **FAIL (FA)**;
- **LOV = (10)** if **INCONCLUSIVE (IC)**;
- **LOV = (11)** if **NOT APPLICABLE (NA)**;

- **External Conformance VERDICT (EXV)**:

  Specifies logical (binary) information stating the VERDICT affecting the Test Event considering its possible relation to events **external** to the current data item under conformance test. The following values can be assigned to the VERDICT:

  - **EXV = (00)** if **PASS (PA)**;
  - **EXV = (01)** if **FAIL (FA)**;
  - **EXV = (10)** if **INCONCLUSIVE (IC)**;
  - **EXV = (11)** if **NOT APPLICABLE (NA)**;

- **Overall Conformance VERDICT (OVV)**:

  Specifies logical (binary) information stating a **final, conclusive** (overall) VERDICT result of the Conformance Test for a specific Test EVENT, with the following alternative values that can be assigned:

  - **OVV = (00)** if **PASS (PA)**;
  - **OVV = (01)** if **FAIL (FA)**;
  - **OVV = (10)** if **INCONCLUSIVE (IC)**;
  - **OVV = (11)** if **NOT APPLICABLE (NA)**;

The **Overall Conformance VERDICT** (OVV) of a Test EVENT is defined from the Local (LOV) **and** External (EXV) Conformance

VERDICTS, and can be expressed by a (Boolean logic) **Karnaugh Map**, as represented in Figure 2.3.

LOV / EXV	*00 (PA)*	*01 (FA)*	*11 (NA)*	*10 (IC)*
00 (PA)	00 (PA)	01 (FA)	11 (NA)	10 (IC)
01 (FA)	01 (FA)	01 (FA)	01 (FA)	01 (FA)
11 (NA)	11 (NA)	01 (FA)	11 (NA)	10 (IC)
10 (IC)	10 (IC)	01 (FA)	10 (IC)	10 (IC)

Figure 2.3. Karnaugh Map of Overall Conformance VERDICT (OVV) States.

- **OBSERVATION (OBS):**

  Specifies information (if any), in RESTRICTED ASCII **text form**, concerning the test report of the specific Test EVENT. If may be a simple message. The following convention is proposed for specifying this information:

  - **OBS = T<text>**
  - OBS = **P<pointer to text>**
  - **OBS = N/O (no observation);**

A **Conformance Test Report Vector (CRV)** for a Test EVENT is defined as the concatenation of the eight REPORT items defined in the Conformance Test Report table. Therefore, as a result, the following expression can defined:

CRV(TEEi) = (INEi, OBOi, EXOi, OVOi, LOVi, EXVi, OVVi, OBSi)

It can also be drawn and defined, from the Conformance Test Report Table, the concept of **Test EVENT Vector (TEV)**, which is the concatenation of EVENTS associated to a Data Item, as follows:

TEV (DAI) = (TEE1, TEE2, ..., TEEn)

Based on the same reasoning, still referred to the Conformance Test Report Table, the following vectors are also defined, for **each** one of its REPORT items:

1) **Instance of Test EVENT Vector (IEV):**
   IEV (DAI) = (INE1, INE2, ..., INEn);

2) **Observed Event OUTCOME Vector (OOV):**
   OOV (DAI) = (OBO1, OBO2, ..., OBOn);

3) **Expected Event OUTCOME Vector (EXV):**
   EXV (DAI) = (EXO1, EXO2, ..., EXOn);

4) **Overall Event OUTCOME Vector (OTV):**
   OTV (DAI) = (OVO1, OVO2, ..., OVOn);

5) **Local Conformance VERDICT Vector (LCV):**
   LCV (DAI) = (LOV1, LOV2, ..., LOVn);

6) **External Conformance VERDICT Vector (ETV):**
   ETV (DAI) = (EXV1, EXV2, ..., EXVn);

7) **Overall Conformance VERDICT Vector (OLV):**
   OLV (DAI) = (OVV1, OVV2, ..., OVVn);

8) **Observation Vector (OBV):**
   OBV (DAI) = (OBS1, OBS2, ..., OBSn);

The last EVENT of the Conformance Test Report Table (**Part No.** equal **n**) is, by definition, the **Data Item** Under Conformance Test (DAI). Therefore, the **Conformance Test Report Vector (CRV) of the Data Item (DAI)** is, by definition, given by the expression:

CRV (DAI) = (INEn, OBOn, EXOn, OVOn, LOVn, EXVn, OVVn, OBSn)

A **criteria** is defined **to determine** the OUTCOMES and VERDICTS of the CRV (DAI) Vector, as a function of the preceding events:

1) **An Observed or Expected OUTCOME** of Test EVENT DAI (the n-th) is defined by: The logical (Boolean) OR-ing of **all non -(11) states** of the preceding (n-1) OUTCOMES, with the following interpretation of the logical (Boolean) results:
   OUTCOME = **(00)**, is **VALID;**
   OUTCOME = **(01)**, is **INVALID;**
   OUTCOME = (10), is **INNOPORTUNE;**
   OUTCOME = (11), is **INVALID** and, therefore, is transformed to **(01)**.
   Otherwise, if **all** preceding (n-1) logical (Boolean) OUTCOMES are equal to (11), then, the Test EVENT DAI has the following OUTCOME assignment:
   OUTCOME = (11), NOT APPLICABLE.

2) **A Local or External Verdict** of Test EVENT DAI (n-th) is defined with a the **same** logic to be used in defining the OUTCOME of the same Test EVENT (DAI).

However, where INNOPORTUNE is the assignment for an OUTCOME, it must be replaced by INCONCLUSIVE in the VERDICT assignment.

The structure and construction rules for the Conformance Test Report Table (CTRT) were given. Moreover, a set of "state" vectors was defined and drawn from the Table structure.

The **fill-up** of a **CTRT** like that of Figure 2.1 has a **practical meaning** only with:

1. Definition of **a Data Item (DAI)** and of its corresponding **Test EVENT Vector (TEV)**;

2. Definition of an **Executable TEST GROUP** to be applied to **Instance Event Vectors (IEVs)** of the Data Item;

3. Definition of a **pointer to** the so called "**Originating Report**" (if any) to which the current CTRT is related to or is derived from;

4. Definition of a **pointer to** the so called "**Following Report**" (if any) to which the current CTRT is related to or is to be its generator;

5. Definition of an **Instance of Test EVENT Vector (IEV)** which will be the **actual data input, besides** all the other Data for Testing Support, for the Executable TEST GROUP;

6. The **execution** of the Executable TEST Group which will fill-up the rest of the CTRT, besides, possibly, generating other logs of data which may be of interest for analysis and assessment of the Conformance Test.

At this point it is opportune to remind that the Executable Test Group may, in fact, be applied as an **Executable Test Body,** if only the essential tests which will generate the CTR (which includes, possibly, only the CTRT) are to be executed. An open question to the concept being criated in this work is the careful investigation of the CTRT as, possibly, a **minimum and or sufficient set of output data** to be accepted as **the** CTR of an Executable Test Body, when applied to a Data Item. Otherwise, besides the CTR, other additional output data should be defined to guarantee the **completeness** of a minimized CTR, expected from the execution of a Test Body. Possibly, the CTRT itself may be expanded or modified to comply with this minimum set of output data expected to characterize a ("complete")CTR, resulting from the execution of a Test Body.

2.4 - **Basic Schemes for Test Flow**

Based on the definitions given in the preceding sections and observing the natural **hierarchy of tests,** the following Figures 2.1, 2.2 and 2.3 give a **basic scheme** devised for structuring abstract and executable procedures for TEST STEPS, TEST CASES and TEST GROUPS, respectively.

It should be noticed that: 1) a TEST STEP is, by definition devoted to test an EVENT; 2) a TEST CASE is structured over a logical set of TEST STEPS to achieve a specific test purpose; 3) a TEST GROUP is to be defined with a logical hyerarchy of TEST CASES. These and other related concepts for conformance testing were introduced in Section 2.2.

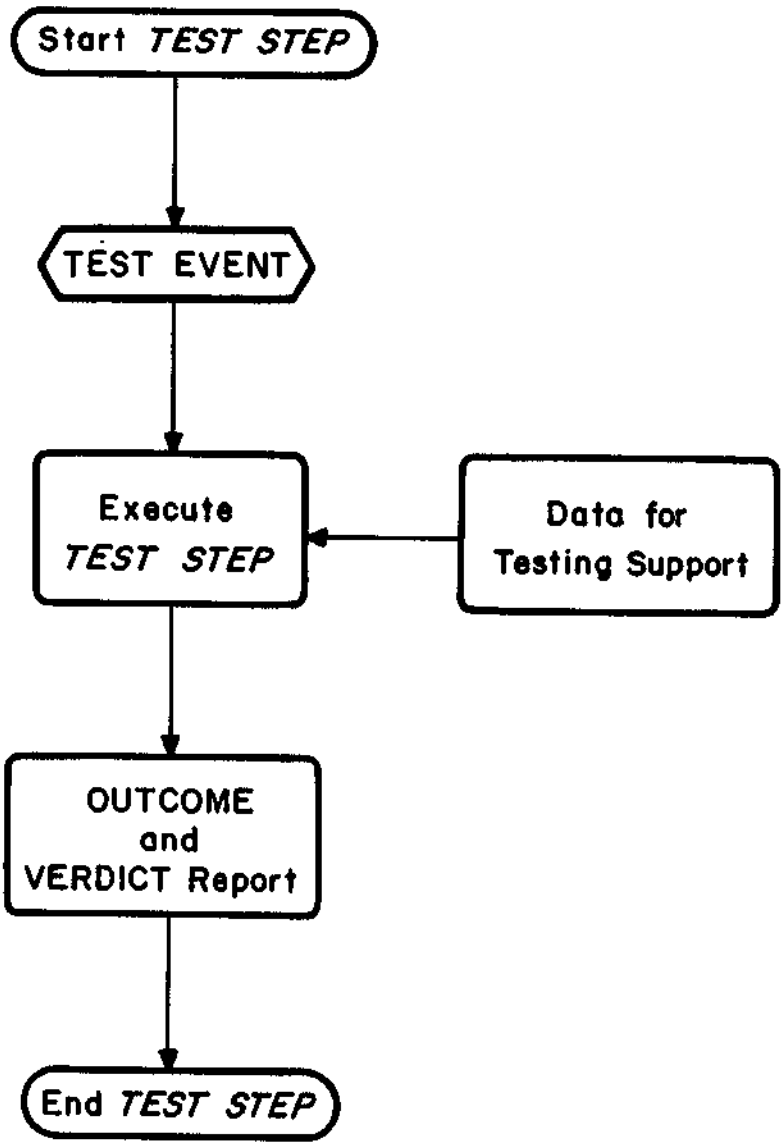

Figure 2.4. Basic Scheme for Execution of a TEST STEP.

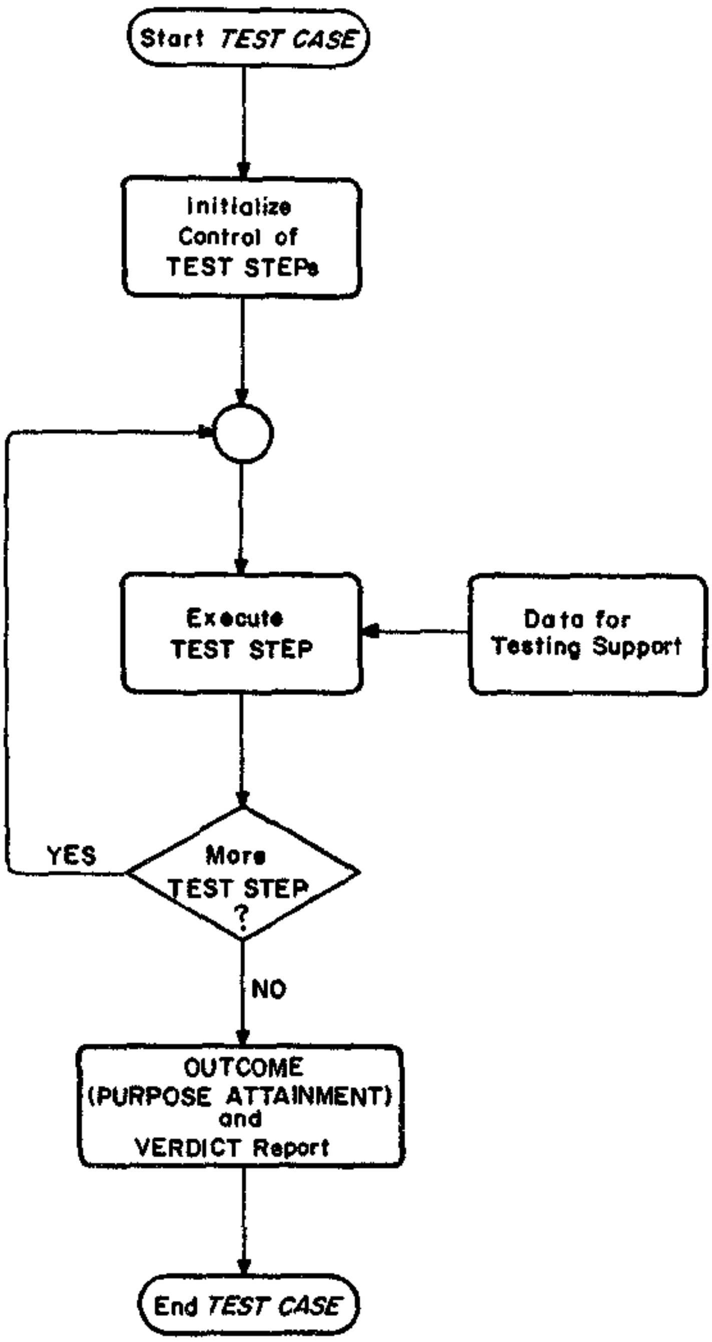

Figure 2.5. Basic Scheme for Execution of a TEST CASE.

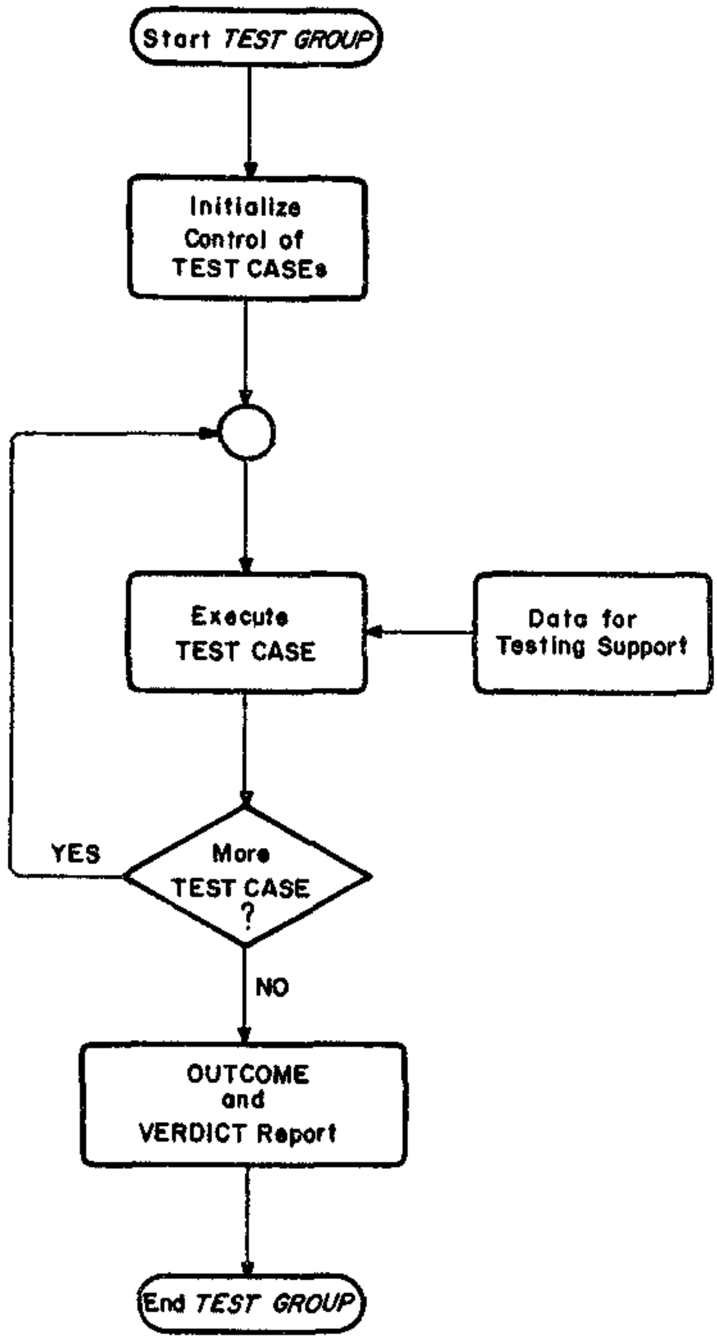

Figure 2.6. Basic Scheme for Execution of a TEST GROUP.

3 - THE MODELLING OF CONFORMANCE TEST FOR DATA ENTITIES

3.1 - Introduction

The Conformance Test (CT) of a Data Entity which o serves the CCSDS Protocol must be structured and formalized by incorporating the construction rules and, therefore, the requirements established by the mentioned protocol.

The execution of Conformance Test and Reporting of its results requires a rationale relying on procedures that should, desirebly, incorporate a modular structure. The **modularity** is of particular interest, considering that a data entity is expected, in general, to be the aggregation of data products with similar structures, not only by means of its simmple concatenation, at a same level of authority, but also, at inner levels of nesting. To some extend, this principle of modularity **by decomposition of Conformance Testing and Report schemes** is treated in the next section of this chapter.

However, all the **basic schemes** which will be introduced for Conformance Test rely on a **basic model** which may be implemented as a **system**, considering the different types of resources involved in its structure. The basic model for a **Conformance Test System** is represented in Figure 3.1. The so called **"CONFORMANCE TEST MACHINE"** represents all the operations, automated or not, necessary to execute the Conformance Test on a **DATA ENTITY**, considered as an input. It is also understood that the conformance test relies on the so called **DATA FOR TESTING SUPPORT** which, also as an input, allow the Conformance Test Machine to be "fully" **executable.**

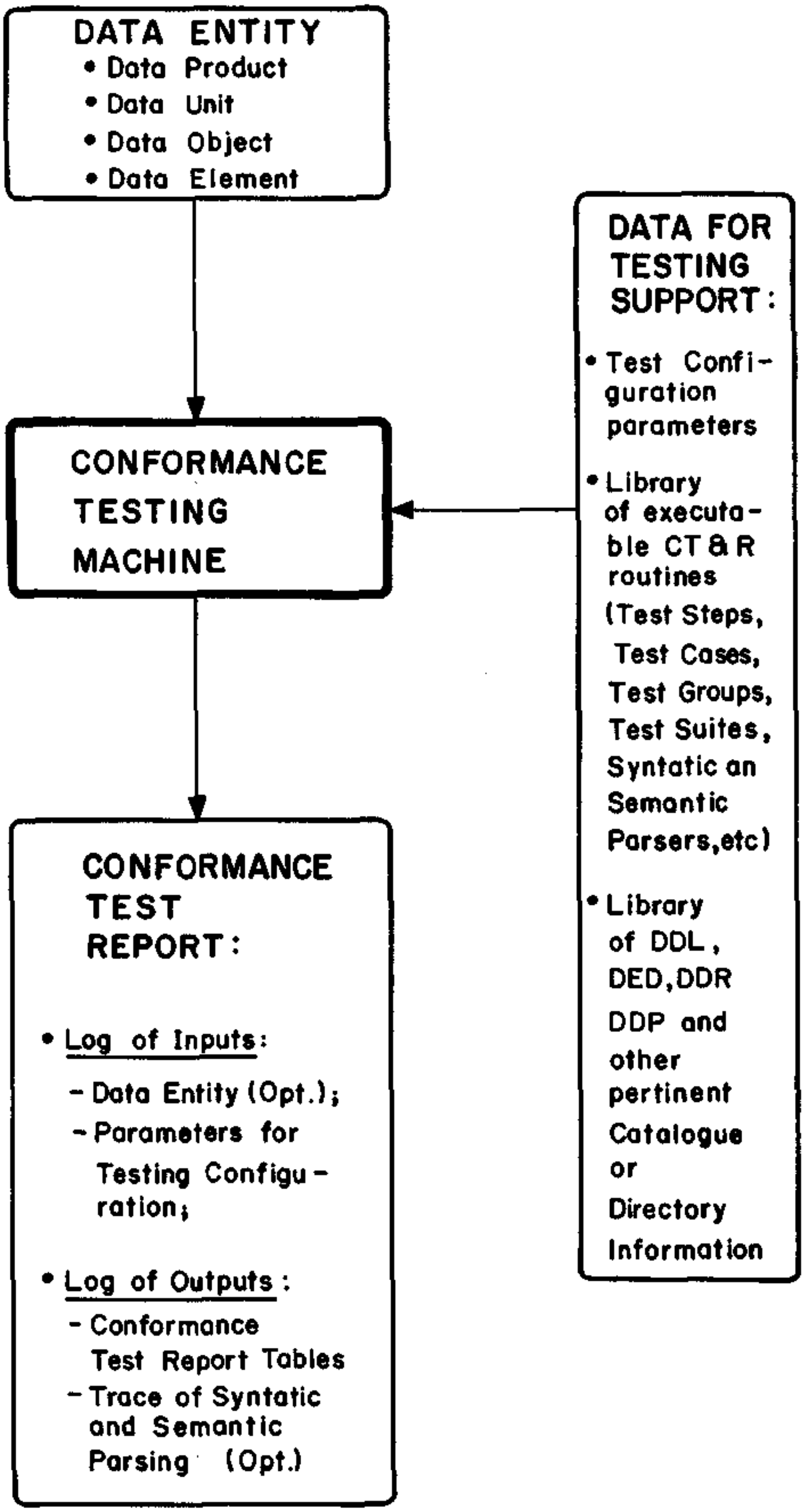

Figure 3.1. Basic Model for Conformance Test System.

3.2. **A Structure Breakdown for Conformance Test**

The Standard Formatted Data Unit Structure and Construction Rules and its related Aggregation Aspects, as defined or exemplified in **References [1],[2] and [3]** permit the structuring of procedures or schemes that can form a basis or **a rationale for conformance test** of the data entities which observe the CCSDS Protocol.

The following Figures 3.2 up to 3.21 represent an attempt to structure **a modular, basic concept for conformance testing procedures.** As a matter of fact, this basic concept is **structured with an hyerarchical set of schemes** which are intended to **breakdown** in (apparently) **basic modules all** the main **construction rules** which must be observed in the construction of data entities that must conform with the CCSDS Protocol. The **requirements** for constructing data entities based on CCSDS Protocol are explicit or implied in the concatenated sequence of hierarchically modular schemes represented from Figures 3.2 to 3.21. This sequence of schemes are intended to be in a **"natural" order** of **depthness or level** of **Conformance Testing** of a data entity which is assumed to be constructed in compliance with the CCSDS Protocol. Each level of conformance test **breakdown structure** is followed by the corresponding CT&R scheme(s), be it recursive or not, that characterizes a basic procedure according the required construction rules for each partial structure. Therefore, it would be expected that if an executable Conformance Test and Report System is applied to a data entity and if it is in functional accordance to the breakdown structure presented in Figures 3.2 to 3.21, **Syntatic and Semantic Parsing** procedures, complying with the mentioned structure, would be executed by the Testing and Reporting System. The modular schemes represented in Figures 3.2 to 3.21 are divided in the so called **"CASES" (00,0 and 1 to 8).** The characterization of these "CASES" was found to be convenient in the breakdown process which was applied, having in mind the different **requirements**, so far, being established for SFDU structuring under its construction rules. Futhermore, the breakdown in "CASES" allowed a more clear separation of breakdown points where **inner or lower levels of authhority** may be **nested** in the data units which are aggregated in a data entity. For instance, this is what may occur in CASE 00 itself, where a **new**, inner (lower) level (of authority) Class-Z data product might be (**recursively**) encountered. The same inner level of recurrence may occur within CASE 6, where a Z-Class data unit may be pointed, in aggregation.

In the Conformance Test Terminology it can be said that the schemes represented in the following figures (2 to 21) of this section would form the basis for constructing a **Test Suite** which would, desirebly, be complete to test any data entity supposed to be in conformance with the CCSDS protocol. Therefore, a whole set of **Test Groups**, with their respective **Test Cases** would have to be derived after deriving, as exhaustively as possible, a complete set of **Test Steps** associated to their corresponding **Test Events**. It could, therefore, be expected that a Conformance Test and Report System would have Syntatic and Semantic Parsers **invoking** the components of an executable Test Suite, in a proper order, when it is executed.

The basic work introduced in Section 2.3 and which is lightly illustrated in Section 3.3 and Chapter 4 would form a basic rational for a complete, in depth, breakdown structure for CT&R (Conformance Test and Report) to, supposedly, cover all the construction rules and requirements of the CCSDS Protocol for SFDUs.

A most careful definition of **Test EVENTS** would be needed **to assure completeness** in the structuring of the many Test Steps which would have to be observed for CT&R of each Data Item. Other issues, like the definition and structuring of what should be meant by **Data for Testing Support** is open for discussions and conclusive results.

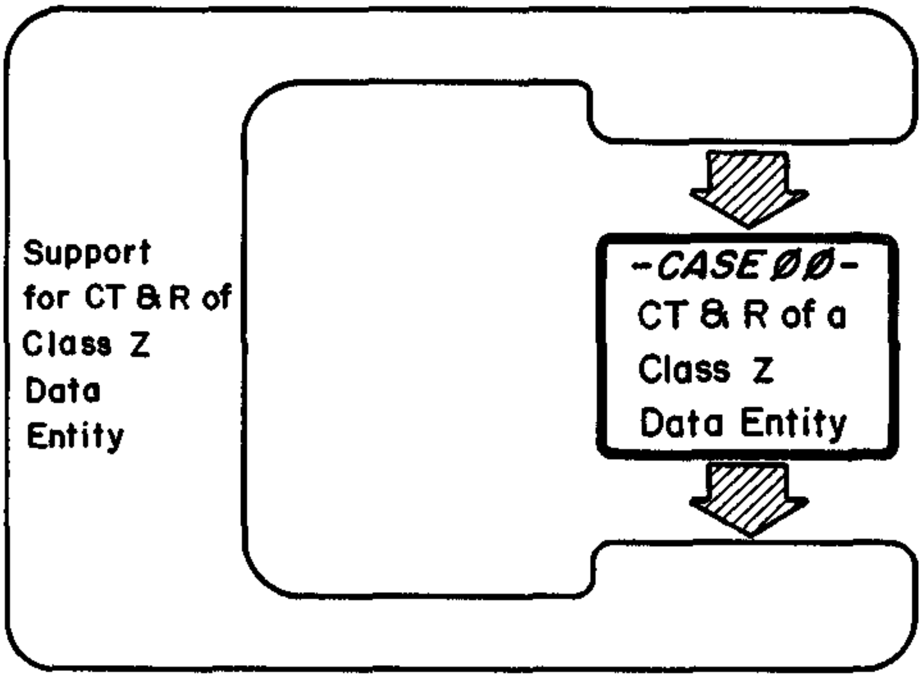

Figure 3.2. Basic Breakdown Model for Recursive CT&R of a Class Z Data Entity.

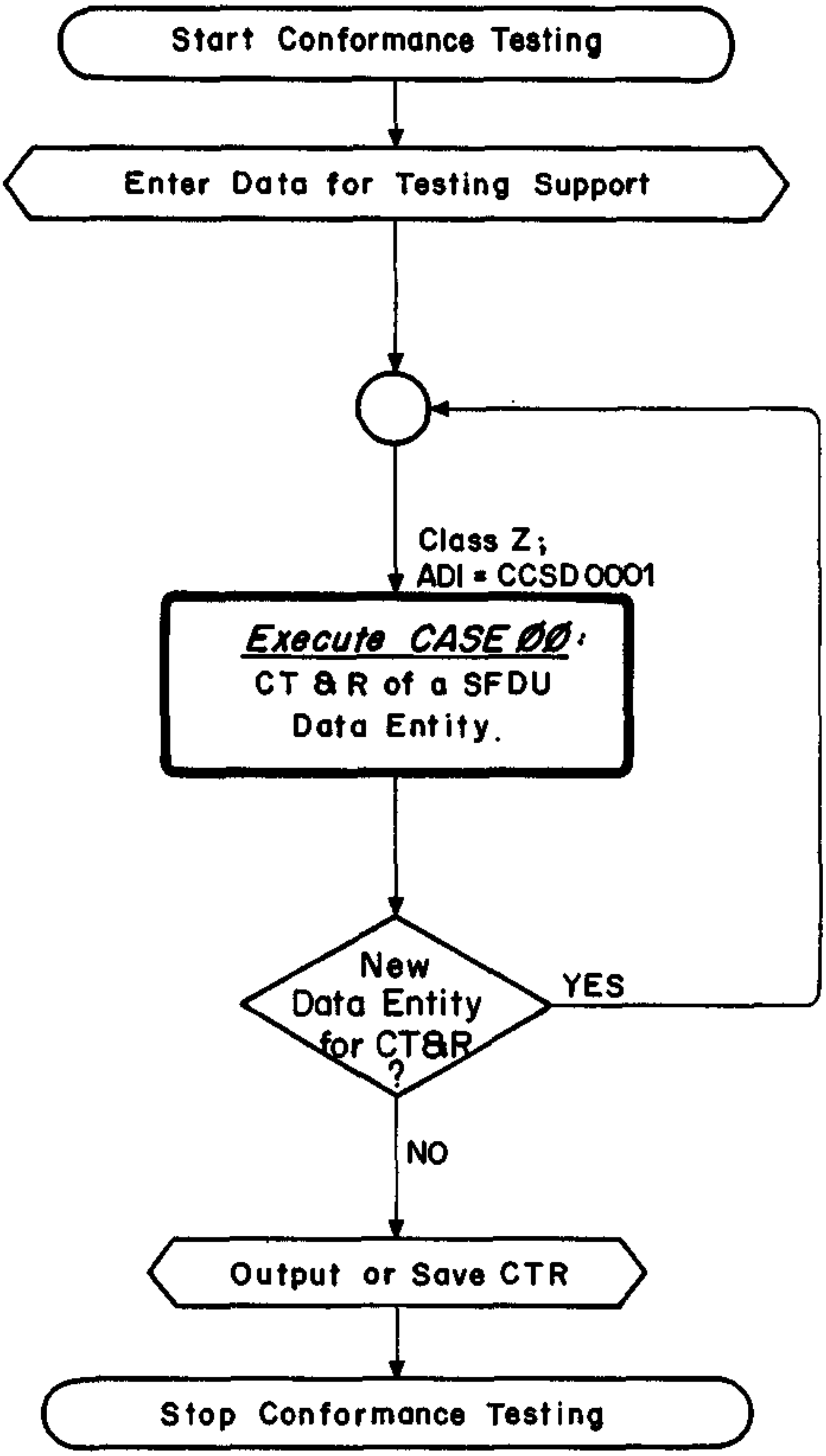

Figure 3.3. Basic Recursive Model for CT&R of SFDU Data Entities.

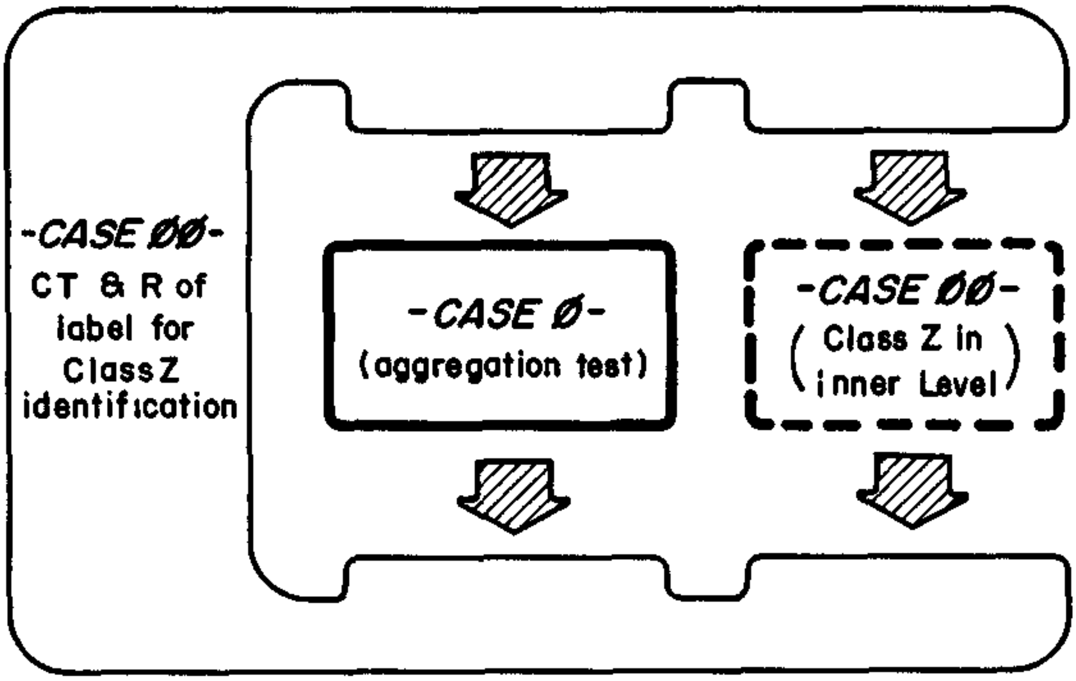

Figure 3.4. Basic Breakdown Model in the CT&R of a Class Z Data Entity.

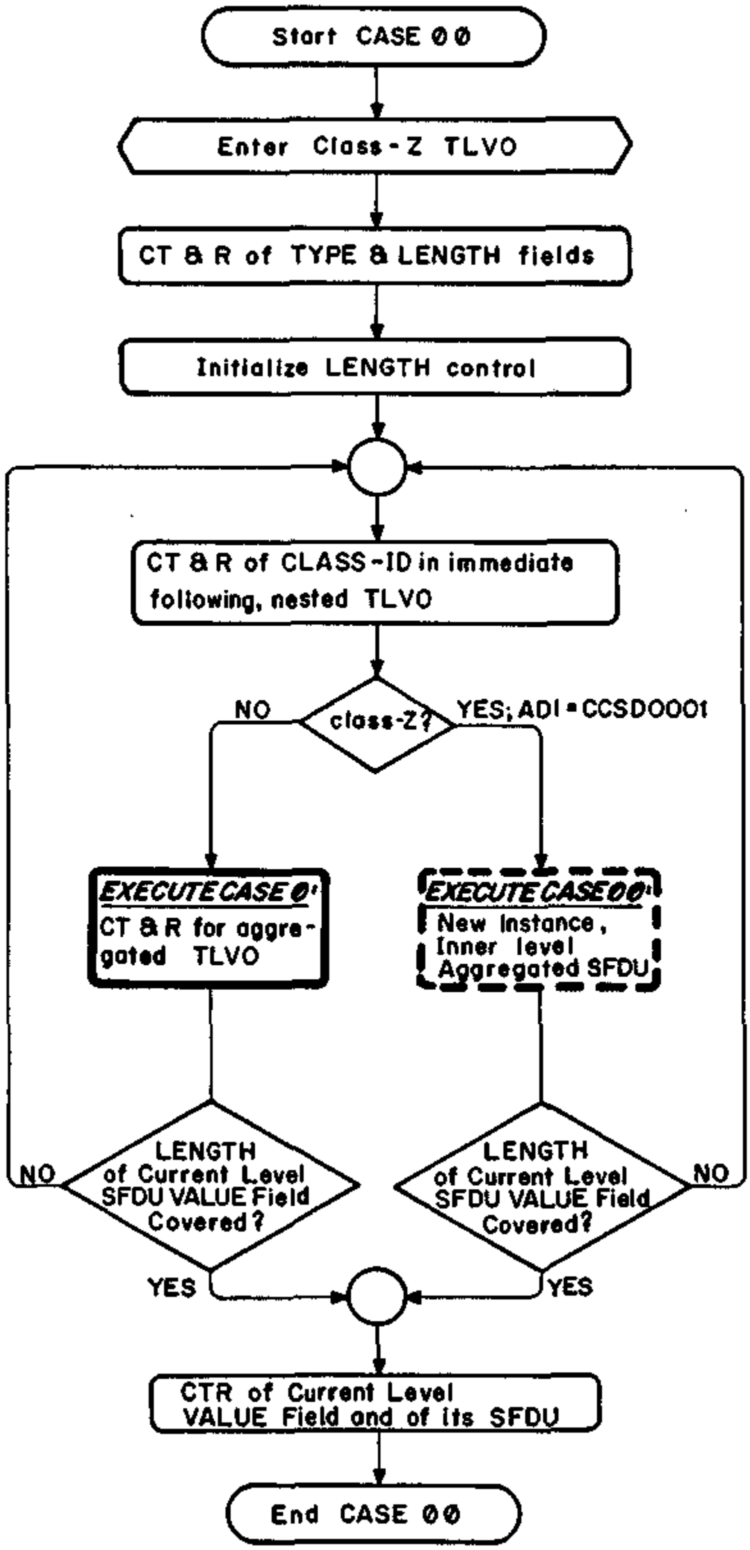

Figure 3.5. Basic Recursive Model for CT&R of a SFDU Data Entity.

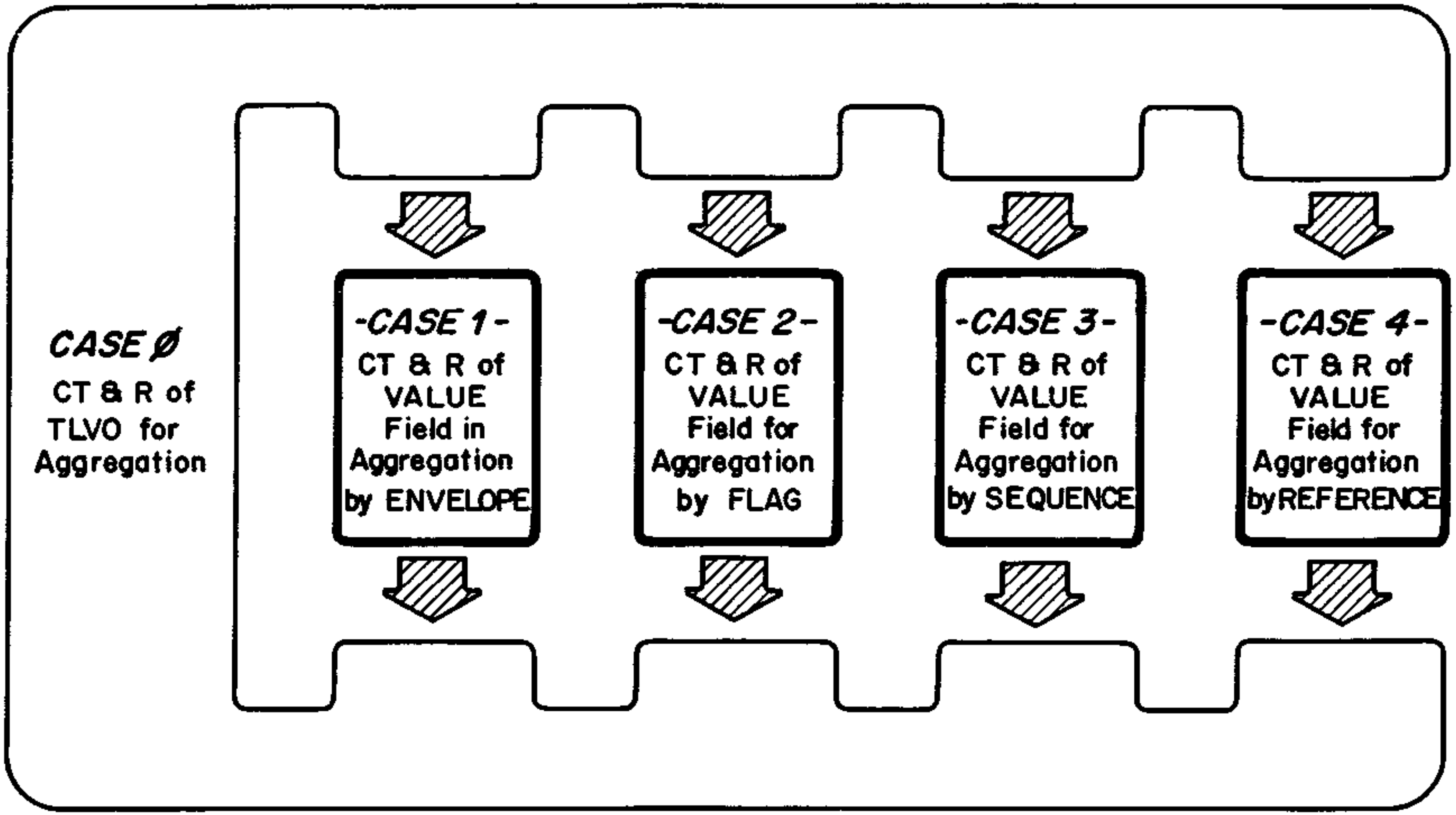

Figure 3.6. Basic Breakdown Model for CT&R of Aggregating TLVO.

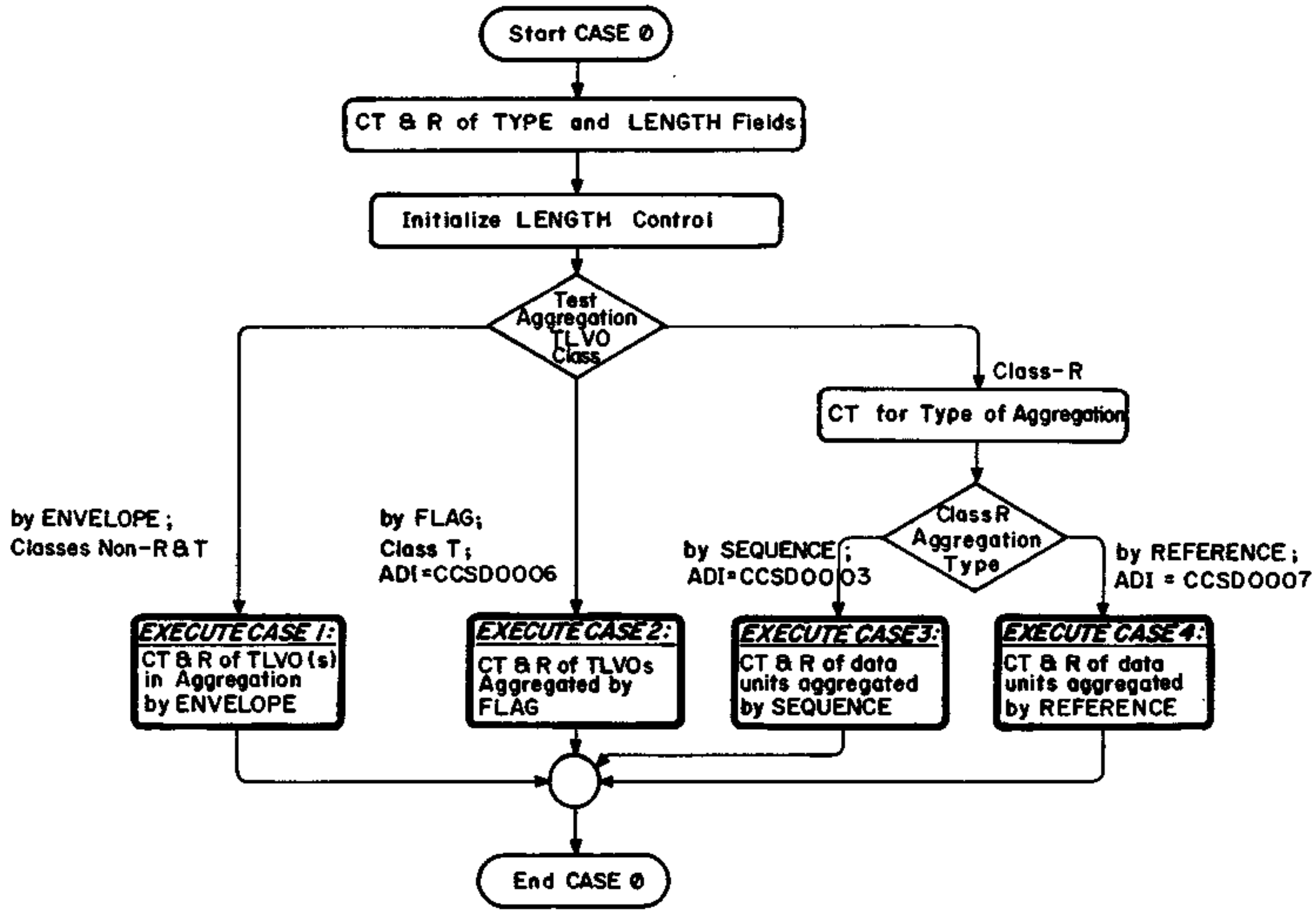

Figure 3.7. Basic Model for CT&R of Aggregating TLVO.

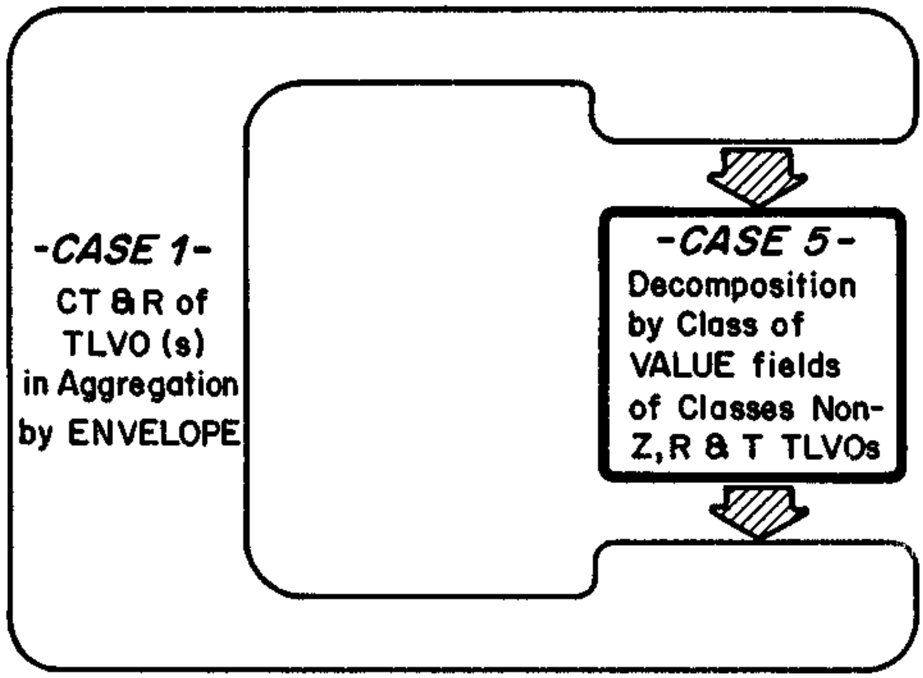

Figure 3.8. Basic Breakdown Model for CT&R of TLVO(s) Aggregated by ENVELOPE.

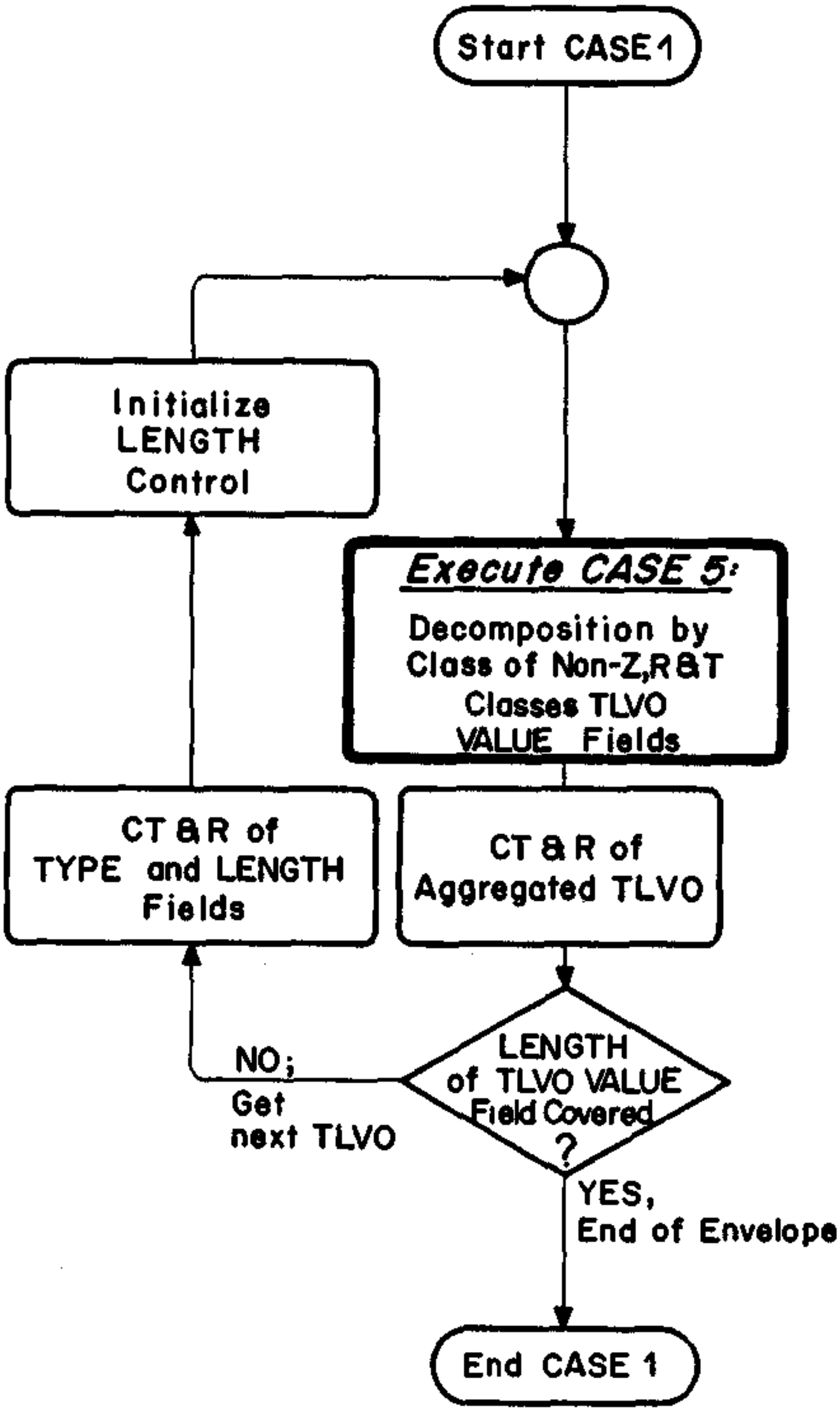

Figure 3.9. Basic Model for CT&R of TLVO(s) Aggregated by ENVELOPE.

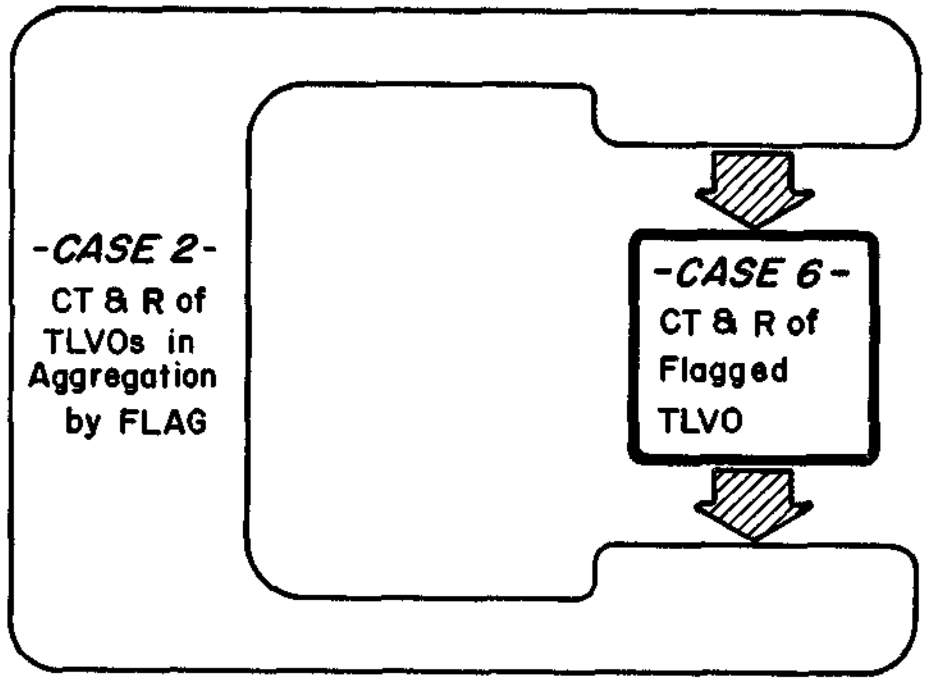

Figure 3.10. Basic Breakdow Model for CT&R of TLVOs in Aggregation by FLAG.

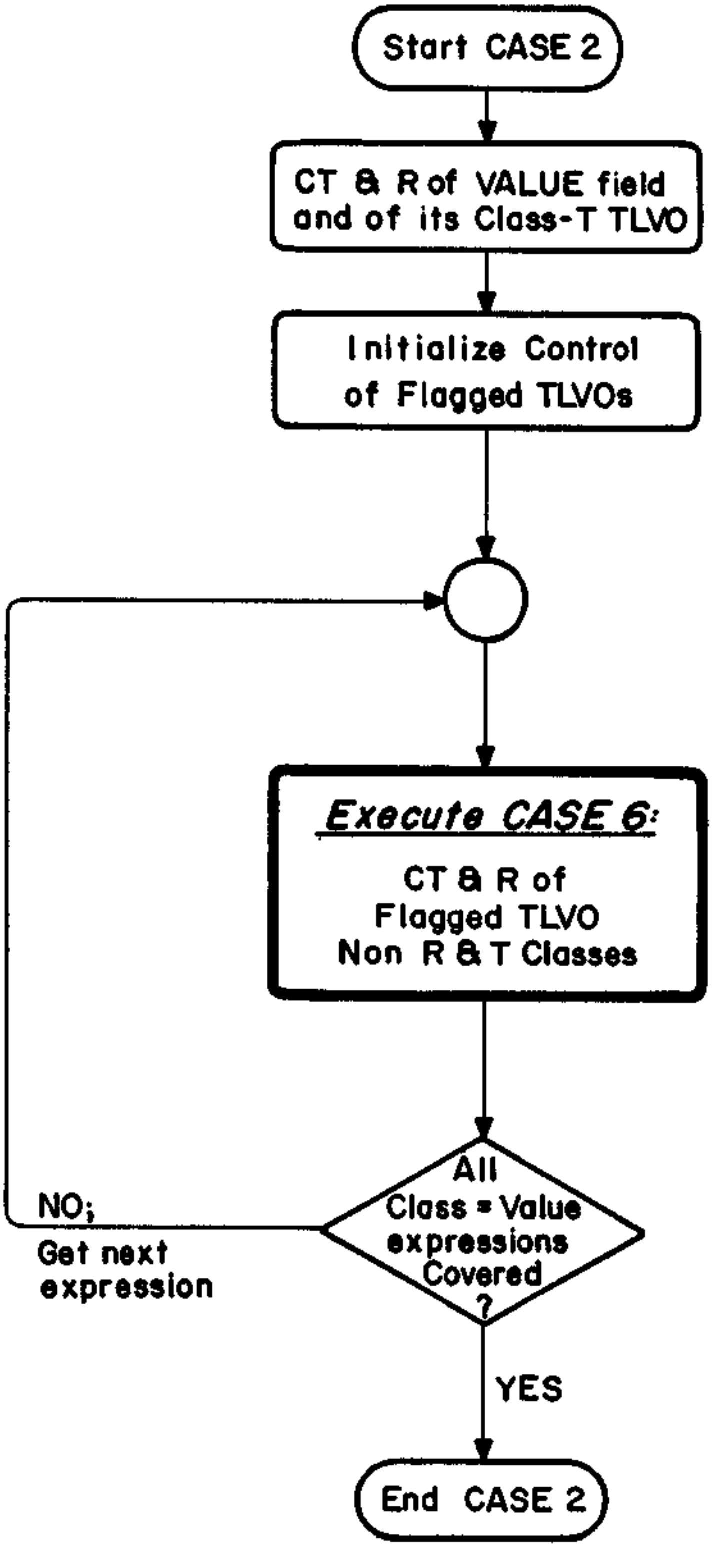

Figure 3.11. Basic Model for CT&R of TLVOs in Aggregation by FLAG.

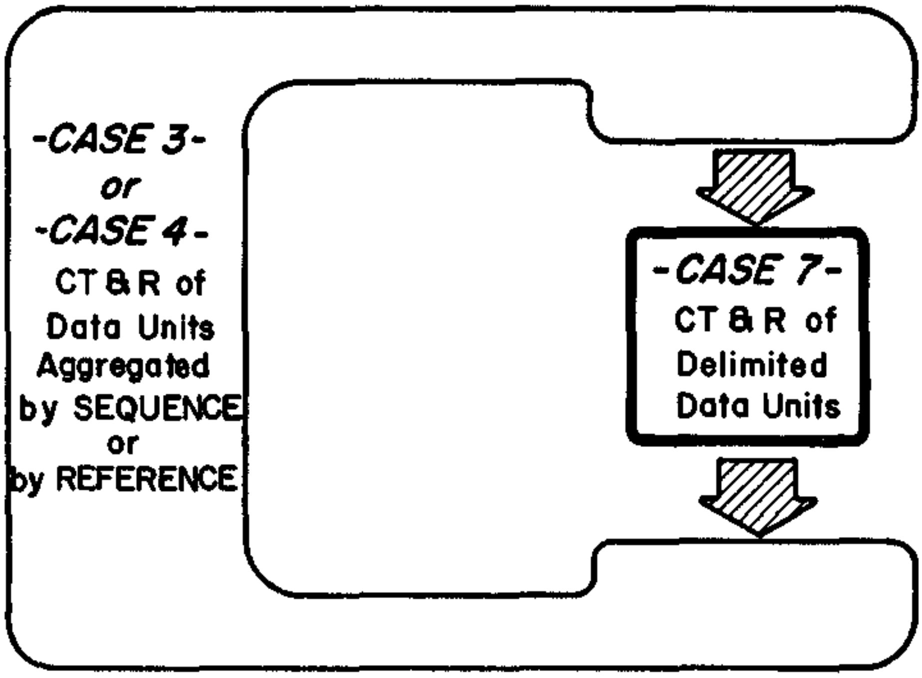

Figure 3.12. Basic Breakdown Model for CT&R of Data Units Aggregated by SEQUENCE or by REFERENCE.

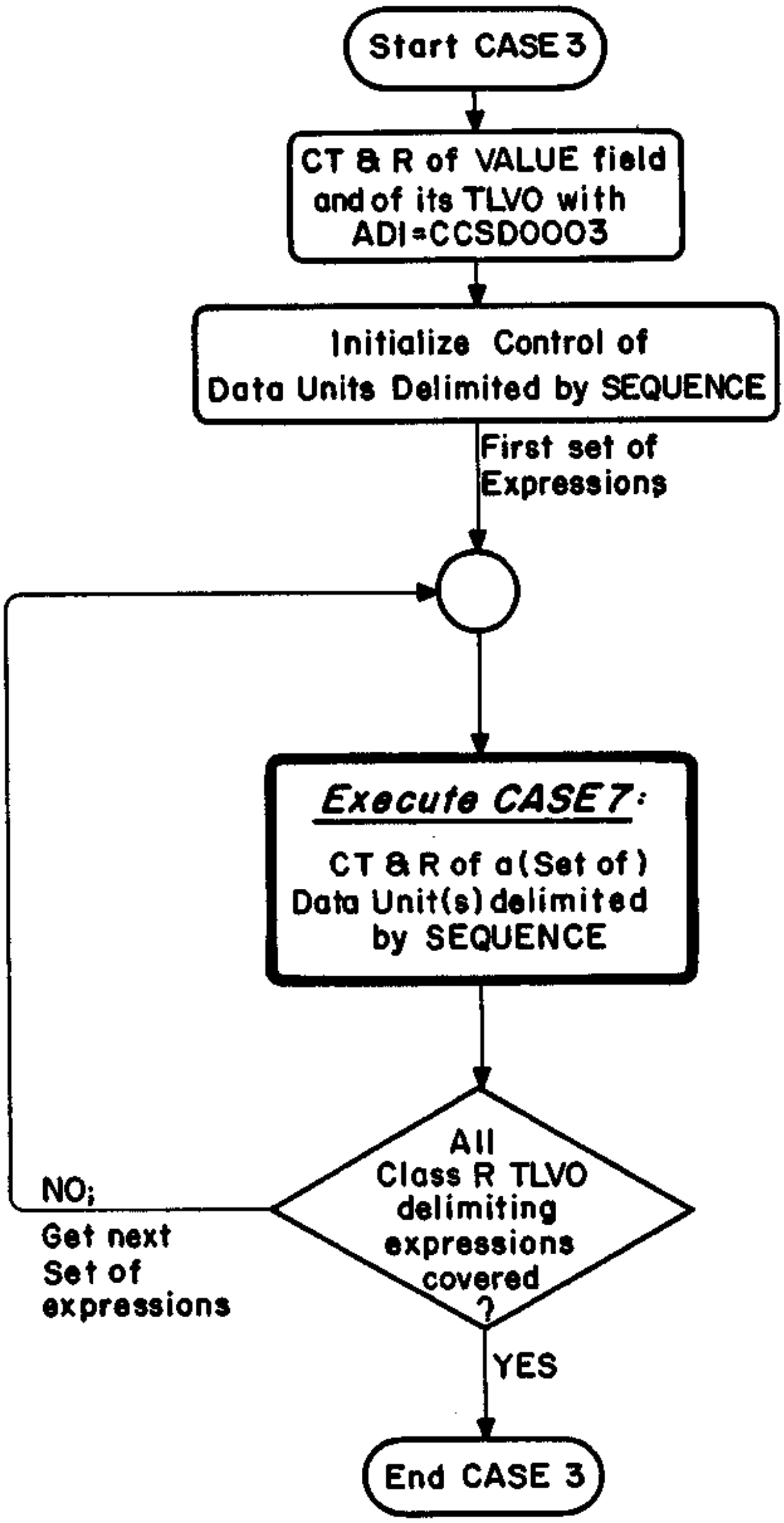

Figure 3.13. Basic Model for CT&R of Data Units(s) Aggregated by SEQUENCE.

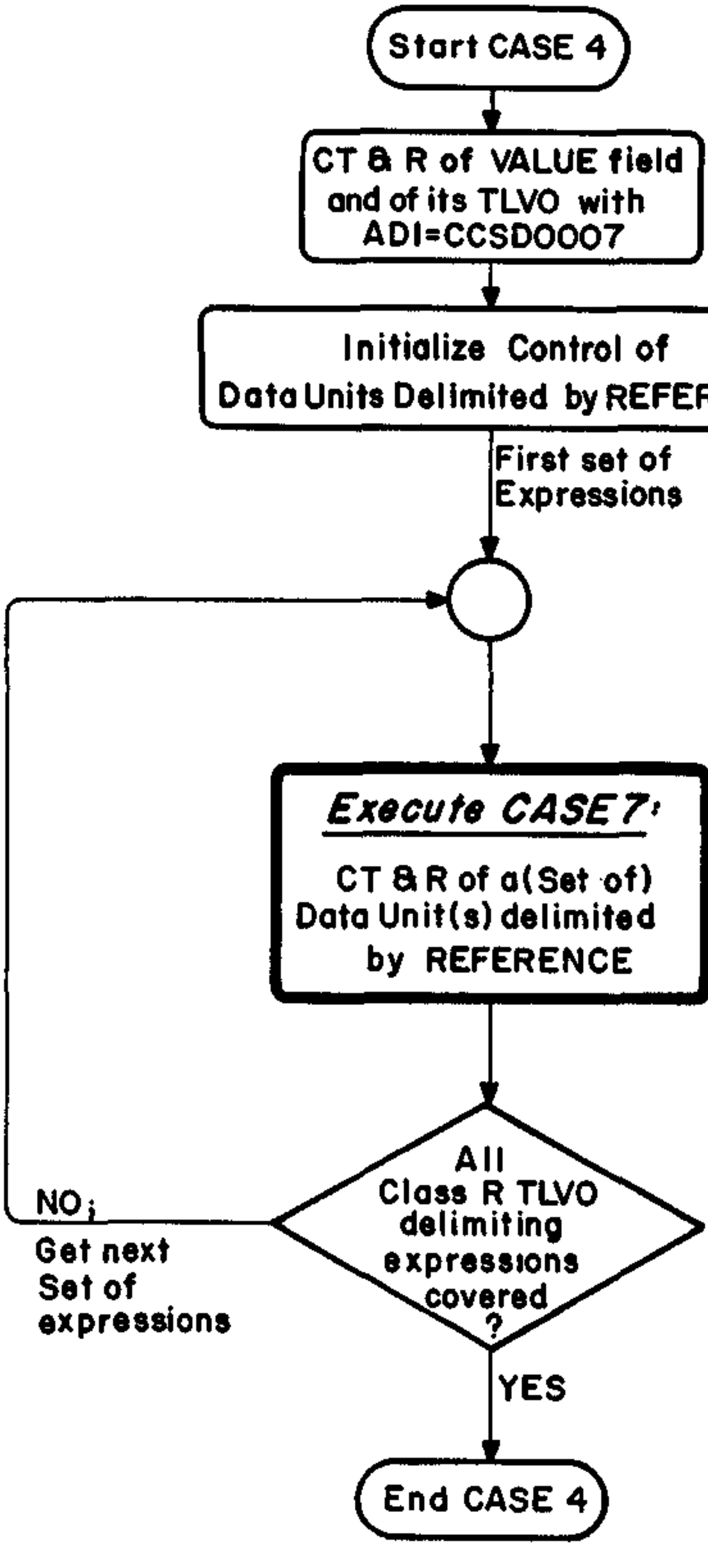

Figure 3.14. Basic Model for CT&R of Data Unit(s) Aggregated by REFERENCE.

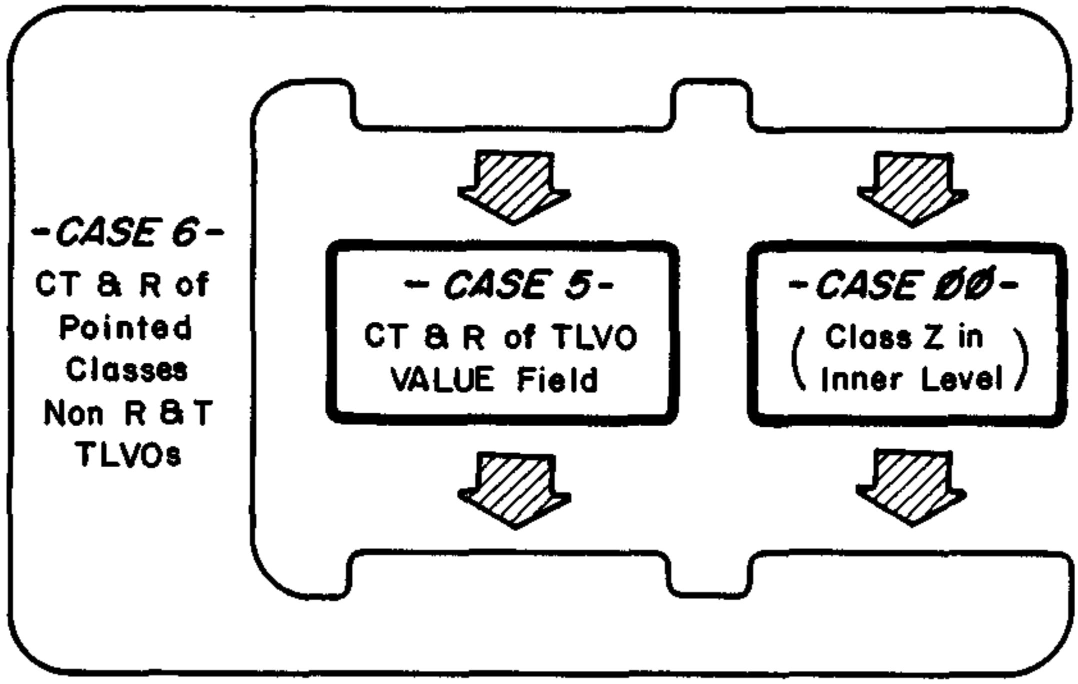

Figure 3.15. Basic Breakdown Model for CT&R of Classes Non-R&T TLVOs.

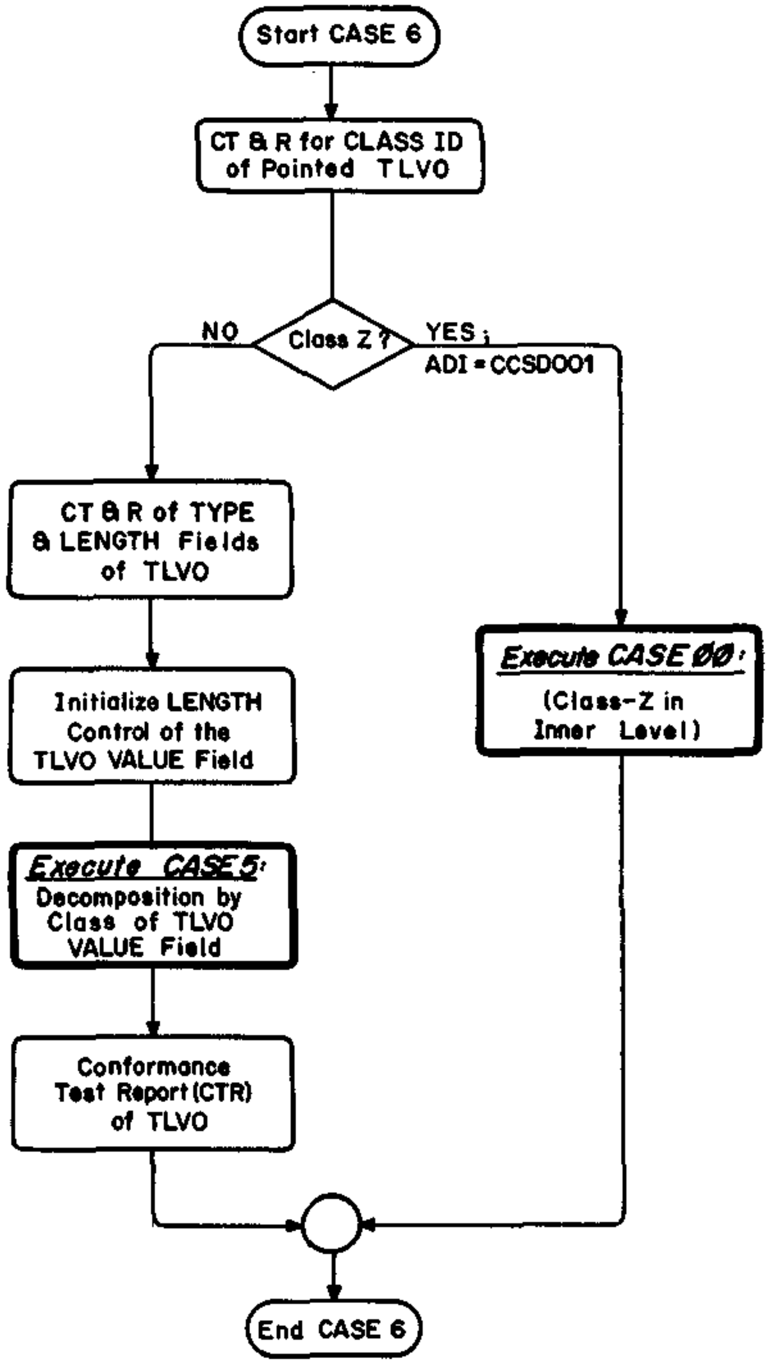

Figure 3.16. Basic Model for CT&R of Pointed Classes Non-R&T TLVOs.

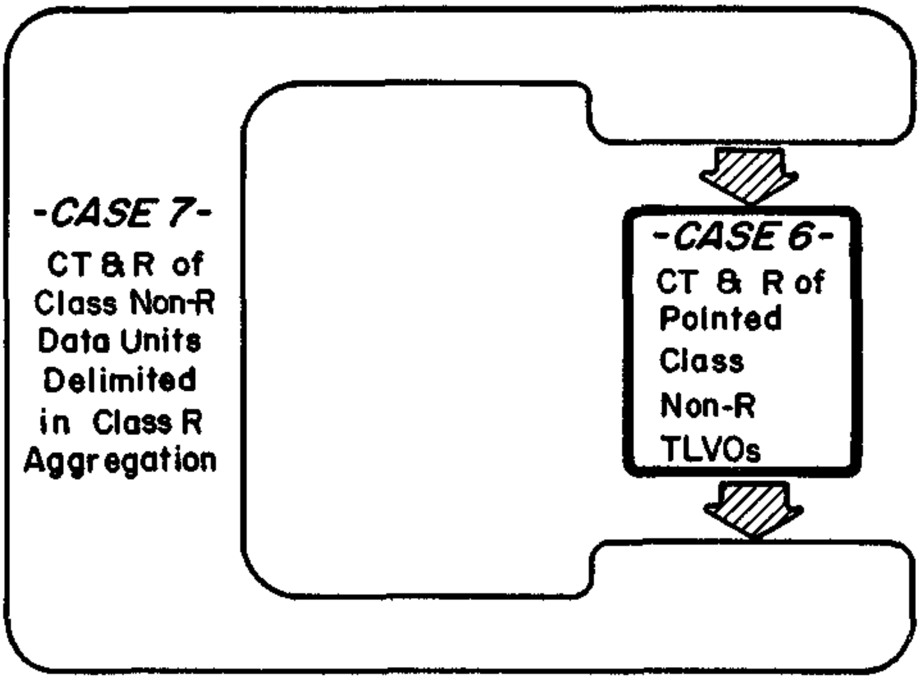

Figure 3.17. Basic Breakdown Model For CT&R of Class Non-R Data Units Delimited in Class R Aggregation.

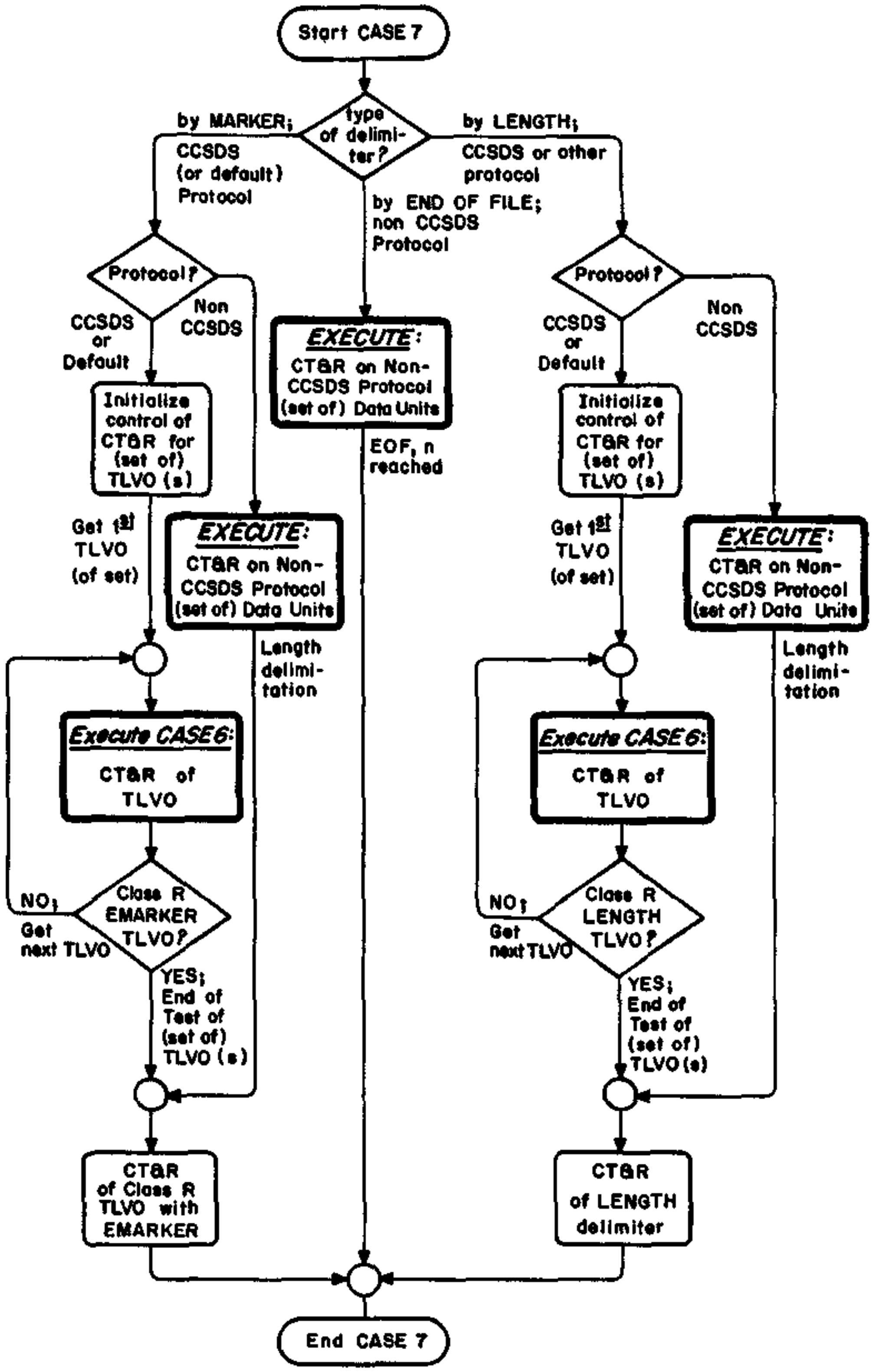

Figure 3.18. Basic Model for CT&R of Class Non-R Data Units Delimited by Class R Aggregation.

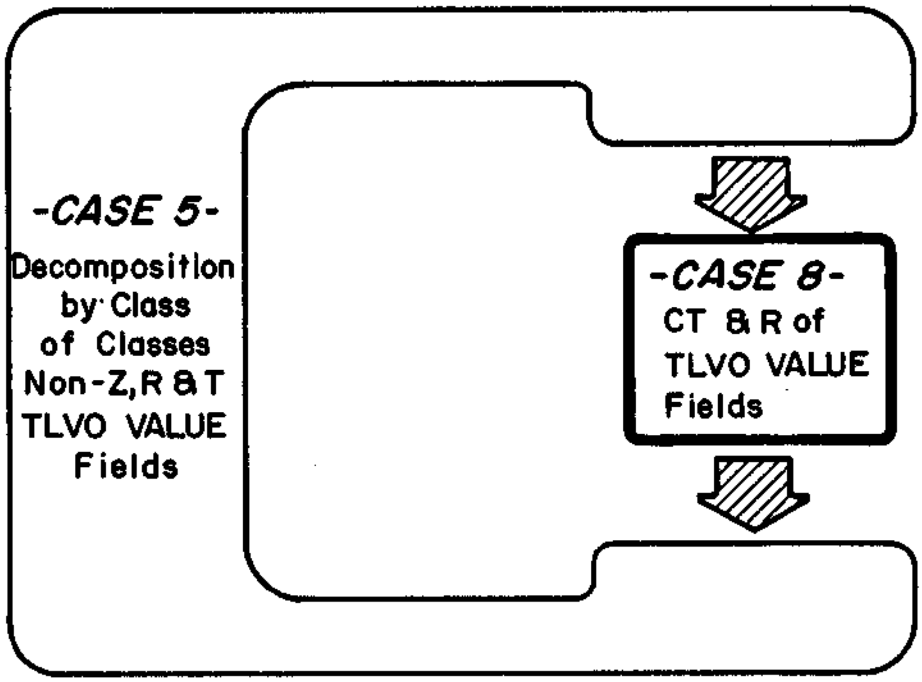

Figure 3.19. Basic Breakdown Model for Decomposition by Class of Classes Non-Z,R&T TLVO VALUE Fields.

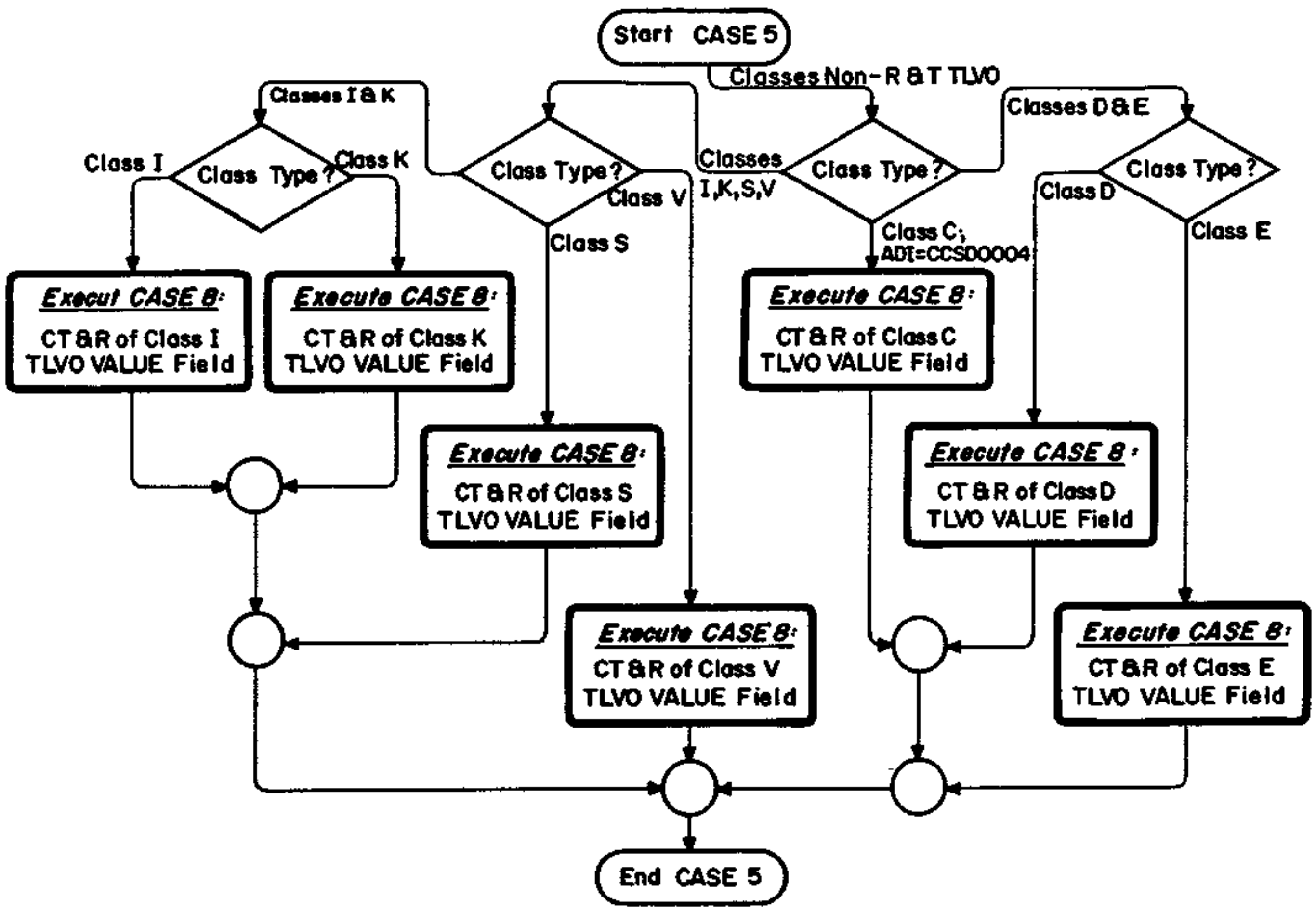

Figure 3.20. Basic Model for Decomposition by Class of Classes Non-Z,R&T TLVO VALUE Fields.

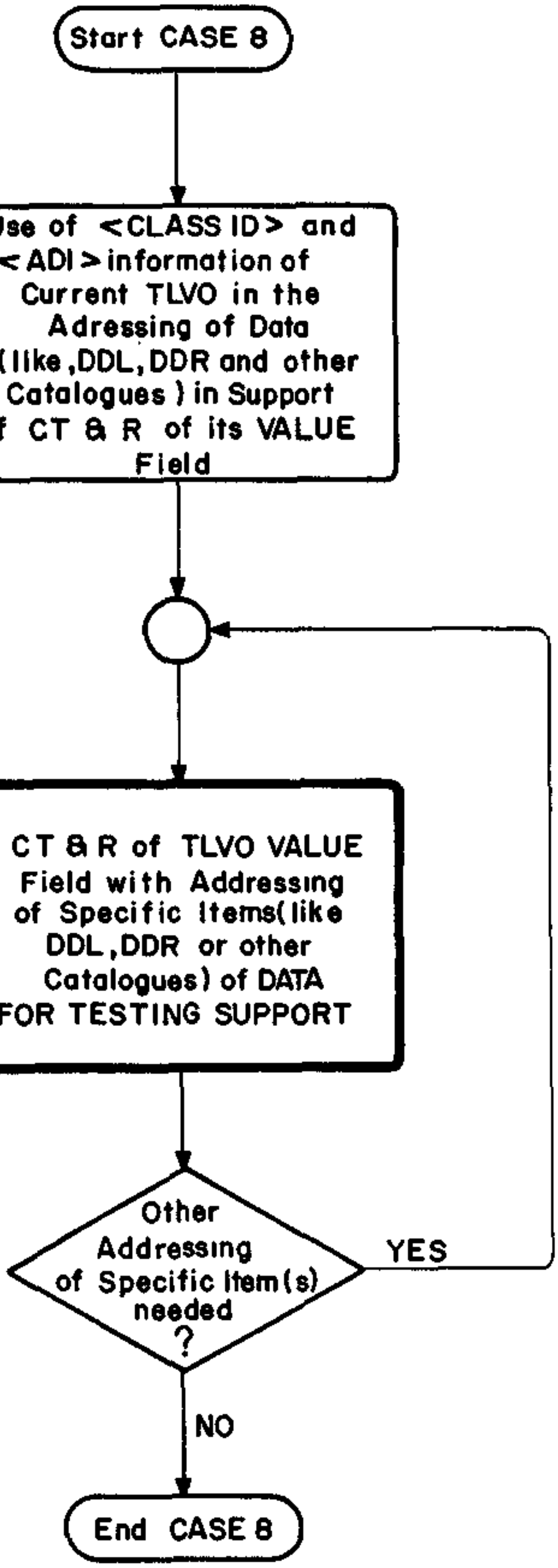

Figure 3.21. Basic Model for CT&R of TLVO VALUE Field of Classes Non-Z,R&T.

3.3 - The Characterization of Data Items

The need for Conformance Test and Report of data items at specific points has been shown in the collection of schemes of preceding Section 3.2, where the structure breakdown for data entities which observe the CCSDS protocol is represented. In the mentioned structure breakdown it is evident that, considering the TLVO as a data entity: the **TYPE (T)** field, **LENGTH (L)** field, or a combination of both, besides the **VALUE (V)** field and, of course, the **TLVO** itself have been characterized as **Data Items** and, therefore, passive of conformance test and report (CT&R).

After a data item has been characterized (e.g. a T field), a next step toward its testing consists in the structuring of its Test Event Vector (TEV). The following Test Events, for instance, can be defined to characterize **a Test Event Vector for a T-Field**, to be submitted to CT&R:

CA	Control Authority ID
VERSION	Version ID
CLASS	Class ID
SPARE	Current 00 Value expected
DDR	Data Desc. Record
ADI	CA & DDR concatenation
AGG	Designates TLVO type of aggregation
LOA	Level of Authority of current TLVO
T	Data item under test (T-Field)

Therefore, whitout pretending, at this point,**completeness of the Test EVENT list**, the Test Event Vector for CT&R of the T-Field, as a Data Item, is:

TEV (T) = (CA, VERSION, CLASS, SPARE, DDR, ADI,
AGG, LOA, T)

After defining the TEV of a Data Item, the **Test Steps** for **each** Test Event should be defined. The **requirements** that must be observed for each Test Event **play a fundamental role in the definition of the Test Step.** Different cases of testing steps can be devised for a single test event, depending on testing alternatives associated to possible **different meanings** of different instances of the test event.

The different sequences (Test Cases) of test steps that lead, each one of them, to achieve one purpose of test on a data item, depending on the alternative requirements that must be observed, are denominated **Test Group.** If just one purpose is being observed, the **strict** (or minimized) set of test steps that can lead to the conformance test of the considered purpose is denominated **Test Body.** In general, a sequence of Test Steps achieve a test purpose as a **Test Case.**

In the current example, conformance test of the T-Field data item can lead to **different Test Steps** when conformance test is applied, for instance, to the CLASS (Id) test event. The same observation is valid concerning the ADI of the T-Field. By extension, **different Test Cases** may result from concatenating the conformance test of the ADI and that of the CLASS (Id), both, Test EVENTS of the T-Field. Another natural extension of this reasoning is the resulting **different Test Cases** that could be expected from alternative test steps that, when executed, would generate the outcomes and verdicts of the Conformance Test Report (CTR) and,possibly, of the CTRT, which would generate the **Test Event Vector of** the T-Field **Data Item,** ie, **TEV (T).**

The **repertoire of Test Cases** that may be executed in a logical order for obtaining the CTR and, of course, the CTRT, may characterize different **Test Groups** for the Data Item, which is, in this example, the T-Field. The **complete** set of Test Groups related to a Data Item characterize a **Test Suite.**

It must be clear that, whenever it is the case, **Data for Testing Support** (Testing Parameters, Library of Data Description Items, Library of Executable Conformance Tests, etc.) **must be available** for execution of Test Steps, Test Cases or Test Bodies, whenever applicable. This implication is explicitly suggested in the schemes of Figures 2.1, 2.2 and 2.3.

Other Data Items, besides the T-Field, can, naturally, be defined in the context of Conformance Test of data entities which observe the CCSDS Protocol. Other examples of Data Items are: L-field; T&L label, VALUE (V) - field; the TLV object itself; an Aggregation of data objects that compose a data entity.

Furthermore, a Data Item like, for instance, the VALUE field may be classified in **different** types of Data Items. This is the case if we consider that VALUE field structures and contents can be characterized from different, well defined Test Event requirements, for instance, as a function of the particular combination of CLASS (Id) and ADI Test Events. In this case, **different aggregation techniques** will **result in different Test Event Vectors** for the VALUE field, as a Data Item. As a result, **different Test Suites** may result in the conformance test of VALUE fields which define each one of the data aggregation techniques.

It must be also observed, in this context, that the CT&R of a data entity will naturally imply on the **concatenation of an hierarchy** of CTRs (or CTRTs), considering the **"chain" or repertoire of Data Items** which must be tested and reported for conformance. A formal verification of this principle can be derived by observing the sequence of schemes of structure breakdown for conformance test of data entities, as represented in Figures 3.2 up to 3.21. This is a reason for providing the **CTRT Table Structure** presented in Figure 2.1 with two heading fields denominated: 1) "**Pointer to Originating Report (POR)**" and; 2) "**Pointer to Following Report (PFR)**".

3.4 - **A Reporting Structure**

Based on the concept of Conformance Test Report Table (CTRT) and on the Structure Breakdown for Conformance Test presented in Sections 2.3 and 3.2, respectively, and considering the comments of the proceding Section 3.3, some **ideas for structuring**, in general, a **set of CTRTs** that would **cover** the basic Conformance Test Report of a Data Item, are presented. This proposal would permit a **commonality** in the reporting of Conformance Test among different users. A common reporting approach would also directly benefit the evaluation of Conformance test among different users by means of a **common** "language" and terminology which would also, desirably, rely on similar syntatic and semantic procedures, for this purpose.

A simple data entity, denominated TLVz, with the basic structure of a data entity, is exemplified in Figure 3.22 to illustrate the basic concept of a reporting structure.

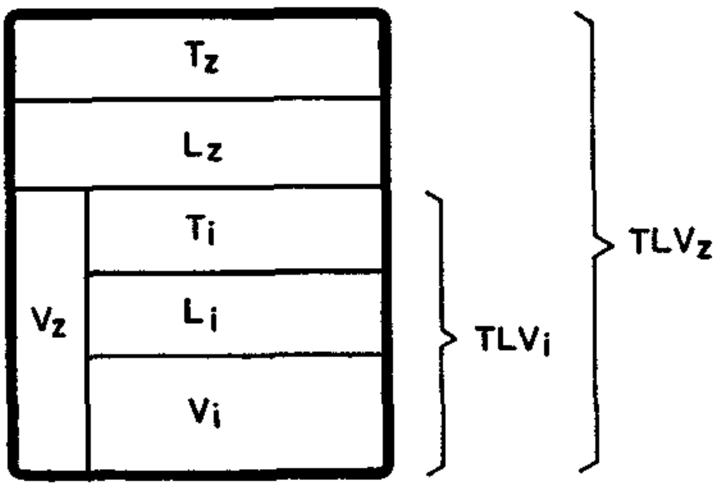

Figure 3.22. Simple TLVz Data Entity

If the data entity structure illustrated in Figure 3.22 is submitted to a Conformance Test and Report Structure as presented in Sections 2.3 and 3.2, a so called **Structure List** of CTRTs would result in a natural sequence as represented in Figure 3.23.

Start Conformance Testing
Enter DATA FOR TESTING SUPPORT

Start CASE ØØ
Enter DATA ENTITY (Class-Z TLV)

- CTRT of T_z Field Data Item
- CTRT of L_z Field Data Item
- CTRT of (CLASS ID)$_i$ Data Item

} T_z & L_z

Start CASE Ø

- CTRT of T_i Field Data Item
- CTRT of L_i Field Data Item

Start CASE 1
Start CASE 5
Start CASE 8

- CTRT of V_i Field Data Item

End CASE 8
End CASE 5

- CTRT of TLV_i Data Item

} TLV_i Object

End CASE 1
End CASE Ø

- CTRT of V_z Field Data Item
- CTRT of TLV_z Data Item

} V_z & TLV_z Obj.

End CASE ØØ
Output or Save CTR of Data Entity
Stop Conformance Testing

Figure 3.23. CTRT Reporting Structured List for the TLVz Data Entity.

It could also be derived from the structured List of CTRTs illustated in Figure 3.23 that the Conformance Test Report Vector of TLVi, as a Data Item, i.e., **CRV (TLVi)** can be derived in its CTRT table as a **function of CRV (Ti) and CRV (Li).** It should, therefore, be understood that **Ti** and **Li** would be **Test Events** of the CTRT Table for Data Item TLVi, **among other Test Events** that would be considered for deriving CRV (TLVi). Naturally, CRV (Ti), CRV (Li) would be **inherited** by the CTRT table of TLVi. Furthermore, still observing Figure 3.23, based on the same preceding reasoning, it can be inferred that the CTRT table for Data Item TLVz would derive the CRV (TLVz) vector from Test Events like Tz, Lz, (CLASS ID)i and TLVi, among others, considered pertinent in the execution of its Conformance Test. Also, in this case, CRV (Tz), CRV (Lz) and CRV ((CLASS ID)i) would be inherited by the CTRT Table that derives CRV (TLVz).

There is a concept of **Conformance Test Report Vector inheritance or (synthesis)** in the **hyerarchy** of Conformance Test Report Tables (CTRTs) which would be derived for a data entity, a SFDU, in this case.

4 - A CASE STUDY

This section is devoted to present and comment aspects of a Case Study of Conformance Test and Report (CT&R) related to a Data Entity (an SFDU), which is exemplified in Figure 4.1.

This case study example illustrates a SFDU data entity where **Aggregation by SEQUENCE** is employed. A Data Definition Package (DDP) is, together with a corresponding **TLVO data instance**, aggregated to the SFDU. Due to the nature of the aggregation, a Class-R TLVO was included in the data entity (SFDU) example, to terminate the sequence of TLVOs which are based on the CCSDS protocol. A data product (FITS file) based on a Non-CCSDS protocol is also aggregated to the data entity, characterizing a more general structure to the exemplified SFDU.

The CT&R structure and basic modelling introduced in Chapters 2 and 3 have been applied to the SFDU example.

The high level, **structured list of CT&Rs and CTRs** presented in Figure 4.2 would result from **parsing** the case study example presented in Figure 4.1, in the same basic fashion as considered in the example given in Section 3.4. Therefore, it would be implied that **Conformance Test Report procedures** would be invoked by the **Syntatic and Semantic parsing processes** which would also structure the Conformance Test and Report System.

In a careful analysis of the contents of Figure 4.2, it can be noticed that the CTR of a TLV object (which is, itself, a data item) is made **only** after its T, L and V fields have been reported as separated data items of the same object.

If a CTRT table like the one introduced in Figure 2.1 is utilized, separately, to report on conformance test of the T,

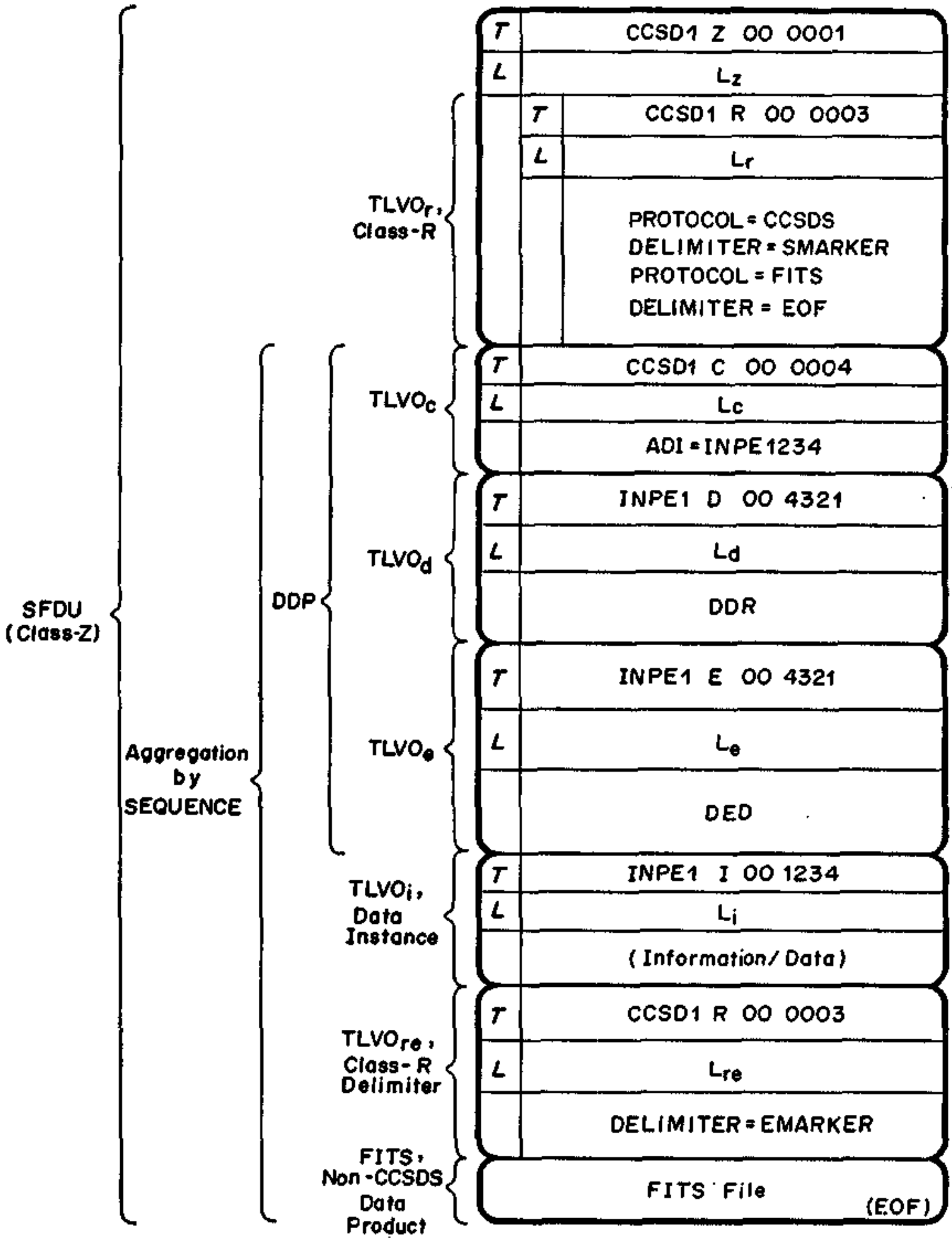

Figure 4.1. A Case Study Example of Data Entity (SFDU) for Conformance Test.

Start Conformance Testing
Enter DATA FOR TESTING SUPPORT

Start CASE 00
Enter DATA ENTITY (Class-Z TLVO)

- CT & R of T_z & L_z Fields
- CT & R of CLASS ID of T_r Field

} T_z & L_z ; Class ID of T_r

Start CASE Ø

- CT & R of T_r & L_r Fields
- CT for (Class R) Type of Aggregation

Start CASE 3

- CT & R of V_r (Value) Field and of TLV_r Object

} T_r, L_r, V_r and TLV_r Object

Start CASE 7
Start CASE 6

- CT & R for CLASS ID of TLV_c Object
- CT & R of T_c & L_c

Start CASE 5
Start CASE 8

- CT & R of V_c (Value) Field

End CASE 8
End CASE 5

- CTR of TLV_c Object

} TLV_c Object

End CASE 6
Start CASE 6

- CT & R for CLASS ID of TLV_d Object
- CT & R of T_d & L_d

Start CASE 5
Start CASE 8

- CT & R of V_d (Value) Field

End CASE 8
End CASE 5

- CTR of TLV_d

} TLV_d Object

End CASE 6

Figure 4.2. PART A - A Structured List of Naturally Sequenced CT&Rs and CTRs resulting from Parsing the Exemplified Case Study Data Entity.

Start CASE 6

- CT & R for CLASS ID of TLV_e Object
- CT & R of T_e & L_e

  Start CASE 5

  Start CASE 8

  - CT & R of V_e (Value) Field

  End CASE 8

  End CASE 5

- CTR of TLV_e Object

} TLV_e Object

End CASE 6

Start CASE 6

- CT & R for CLASS ID of TLV_i Object
- CT & R of T_i & L_i

  Start CASE 5

  Start CASE 8

  - CT & R of V_i (Value) Field

  End CASE 8

  End CASE 5

- CTR of TLV_i Object

} TLV_i Object

End CASE 6

- CT & R of Class R TLV_{re} with EMARKER } TLV_{re} Object
- CT & R of FITS (Non-CCSDS Protocol) File } FITS File

End CASE 7

End CASE 3

End CASE Ø

- CTR of Current Level VALUE Field and of Its SFDU } V_z and SFDU

End CASE ØØ

Output or Save CTR (of Data Entity)

Stop Conformance Testing

Figure 4.2. PART B (Conclusion) - A Structured List of Naturally Sequenced CT&Rs and CTRs resulting from Parsing the Exemplified Case Study Data Entity.

and V fields, their respective **Conformance Test Report Vectors (CRVs)** should be considered as **inputs** to the T, L and V Test Events of the CTRT table of the corresponding TLV object which is, of course, a Data Item itself.

Again, as already mentioned in the example of Section 3.4, the **hyerarchical** use of the Conformance Test Report Vector (CRV) of one or more data items as inputs to the CTRT tables of **other** data items would be desirable, if not necessary. This possibility would permit the structuring of the conformance test report procedures to be invoked by the conformance test system by means of its syntatic and semantic parsing processes. An implementation with this type of resource would concur for **compactness** and, of course, **consistency** and **efficiency** in the execution of conformance tests.

5 - CONCLUSIONS

The concepts introduced in this work are an attempt to contribute to the formalization of a **basic methodology** for Conformance Test of Data Entities which observe the SFDU structure and construction rules. Although an introductory formalism was structured to characterize some aspects of what would be the desired methodology, other related concepts were only identified and indicated. It is suggested that these and, of course, other concepts, not yet identified, should be explored, in further development, in harmony with the desired methodology. It would, therefore, be expected that the whole range of concepts considered of interest should be formalized in an integrated fashion. As a result a **complete methodology** for conformance testing would be formalized. Desirebly, the **completeness** of the methodology would, in principle, assure a **unique approach** for implementing conformance test systems in different environments (e.g., agency control authorities, disciplinary areas, etc.).

Hopefully, such an approach would make **viable, consistent** and secure the exchange and storage of data products within an user community. Futhermore, a unique, basic methodology for Conformance Test would support a desired **certification of data products** to be exchanged between control authorities, possibly, executed by a **CCSDS "certification laboratory"**. The certification concept could be applied on a sample basis, if it is the case. A common methodology for Conformance Test would also facilitate the **assessment** to verify compliance of data products with the requirements of the CCSDS protocol. In the same way, the **auditability** on data products would be facilitated to independent parties or control authorities, while executing a conformance evaluation process.

Among other topics which would have to be explored to obtain a methodology for conformance testing, it is suggested that the following (topics) could be developed:

- **Revision and detailing of the Break down Structure for Conformance Test,** for consistency and completeness;

- Complete identification, characterization and desired structure or organization of **DATA FOR TESTING SUPPORT;**

- Revision of the **Conformance Test Report** scheme using **Tables** (CTRTs), if so, also for consistency, completeness and sufficient detailing;

- Establish a basic scheme for **inheritance** of Data Items **testing results (DRVs)** as inputs to a, supposedly, complete set of concatenated CTRTs that would result from the application of conformance test to a data product;

- Characterize a **complete** set of **Data Items** (T, L, V and other items) that would be passive of testing and report;

- Define a **"complete"** set of **Test Events** for each data item which would be identified for testing and report;

- Define **output reporting structures,** besides CTRTs, for instance, that would contribute to the analysis and, possibly, to the assessment of conformance test results (e.g. structured lists to trace syntatic and semantic parsing, etc.);

- **Other** (T.B.D.).

- ACRONYMS -

ADI	CA & DDR IDs concatenation
AGG	Designates TLVO type of aggregation
ASCII	American Standard Code for Information Interchange
CA	Control Authority ID
CAID	Control Authority Identifier
CCSDS	Consultative Committee for Space Data System
CLASS	Class ID
CRV	Conformance Test Report Vector
CT	Conformance Test
CTR	Conformance Test Report
CT&R	Conformance Test and Report
CTRT	Conformance Test Report Table
DAI	Data Item
DDL	Data Descriptive Language
DDP	Data Descriptive Package
DDR	Data Descriptive Record
DED	Data Element Dictionary
EOF	End of File
ETV	External Conformance VERDICT Vector
EXO	Expected Event OUTCOME
EXV	Expected Event OUTCOME Vector
EXV	External Conformance VERDICT
FITS	Flexible Interchange Transport System
ID	Identifier
IEV	Instance of Test EVENT Vector
INE	Instance of EVENT
IP	Innoportune
L	LENGTH Field
LCV	Local Conformance VERDICT Vector
LOA	Level of Authority of Current TLVO
LOV	Local Conformance VERDICT
MA	Member Agency

NA	NOT APPLICABLE
N/O	No Observation
OBO	Observed Event OUTCOME
OBS	OBSERVATION
OBV	Observation Vector
OLV	Overall Conformance VERDICT Vector
OOV	Observed Event Outcome Vector
OTV	Overall Event OUTCOME Vector
OVO	Overall Event OUTCOME
OVV	Overall Conformance VERDICT
POR	Pointer to Originating Report
PFR	Pointer to Following Report
RA	Restricted ASCII
SFDU	Standard Formatted Data Unit
SPARE	Current 00 Value expected
T	TYPE Field
T.B.D.	To Be Determined
TEE	Test Event
TEV	Test EVENT Vector
TLV	Type - Length - Value
TLVO	TLV Object
UV	UNVALID
V	VALUE Field
VA	VALID
VERSION	Version (ID)

MINISTÉRIO DA CIÊNCIA E TECNOLOGIA
INSTITUTO DE PESQUISAS ESPACIAIS

PROPOSTA PARA PUBLICAÇÃO

- [ ] DISSERTAÇÃO
- [ ] TESE
- [x] RELATÓRIO
- [ ] OUTROS

TÍTULO

CONFORMANCE TEST CONCEPTS FOR DATA ENTITIES BASED ON SFDU STRUCTURE & CONSTRUCTION RULES

IDENTIFICAÇÃO

AUTOR(ES): Eduardo Whitaker Bergamini

ORIENTADOR:

CO-ORIENTADOR: (DISS. OU TESE)

LIMITE: __/__/__ **DEFESA:** __/__/__ **CURSO:** **ORGÃO:**

DIVULGAÇÃO: [ ] EXTERNA [ ] INTERNA [ ] RESTRITA

EVENTO/MEIO: [ ] CONGRESSO [ ] REVISTA [ ] OUTROS

REV. TÉCNICA

NOME DO REVISOR: Juan Suñe Perez

RECEBIDO: 06/06/89 **DEVOLVIDO:** 15/06/89 **ASSINATURA:** [signature]

APROVAÇÃO

NOME DO RESPONSÁVEL: Eduardo Whitaker Bergamini

APROVADO: [x] SIM [ ] NÃO **DATA:** 08/06/89 **ASSINATURA:** [signature]

REV. LINGUAGEM

Nº: **PRIOR:** **RECEBIDO:** __/__/__ **NOME DO REVISOR:** Juan Suñe Perez

PÁG.: **DEVOLVIDO:** 15/06/89 **ASSINATURA:** [signature]

DATILOGRAFIA

OS AUTORES DEVEM MENCIONAR NO VERSO INSTRUÇÕES ESPECÍFICAS, ANEXANDO NORMAS, SE HOUVER

RECEBIDO: **DEVOLVIDO:** __/__/__ **NOME DA DATILÓGRAFA:** Natalia Santos Gomes

Nº DA PUBLICAÇÃO: **PÁG.:**
CÓPIAS: **Nº DISCO:** **LOCAL:**

DIRETOR

AUTORIZO A PUBLICAÇÃO: [x] SIM [ ] NÃO 11/08/89 [signature]

OBSERVAÇÕES E NOTAS

APÓS A PUBLICAÇÃO, ESTA FICHA SERÁ ARQUIVADA JUNTAMENTE COM A VERSÃO ORIGINAL