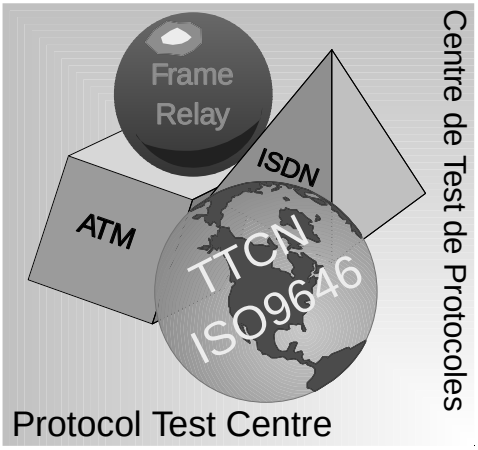
Tree and Tabular Combined Notation

TTCN: What it is and How to read it

from
HP Protocol Test Centre

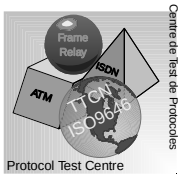


http://www-ptc.canada.hp.com

HEWLETT PACKARD

- This document presents the TTCN International Standard notation for protocol test script writing.
- The main purpose of the following pages is to help people read and understand Abstract Test Suites (ATS) written in this notation, such as ATSs found in manuals shipped with Hewlett-Packard Conformance and Interoperability Test Suites.

- Examples in this document are excerpts from publicly available ATSs, both narrowband (e.g. ISDN, Frame Relay) and broadband (e.g. ATM).

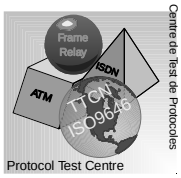


TTCN: Definition

- **Tree and Tabular Combined Notation (TTCN) is:**
  - A precisely defined notation;
  - Used for specifying test scenarios (ATSs);
  - An International Standard, part 3 of ISO 9646 Methodology. Also accepted by ITU-T.
  - Independent of:
    - test methods,
    - protocols,
    - layers,
    - test platforms.

HEWLETT PACKARD

- As part of ISO 9646, TTCN reflects the methodology and framework of this International Standard.
- As an International Standard, TTCN is recognized throughout the world by numerous standard bodies and testing committees.
- TTCN is widely used in lower layer protocols, including:
  - ATM Cell Layer (ATM Forum Conformance ATS),
  - ATM Signalling (ATM Forum ATS),
  - Frame Relay UNI & NNI (ACT-FR), both PVC and SVC,
  - ISDN LAP-D,
  - ISDN Layer 3 (NI-1 BCC, NI-1 SS, VN3),
  - X.25 DTE Layer 2 and 3,
  - MTP (SS#7).
- TTCN is also used in upper layer protocols, such as:
  - FTAM
  - MHS
  - SCCP, TCAP (SS#7)
  - Session



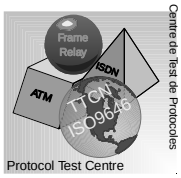
TTCN: What more does it bring?

- Provides a formal notation that **completely and unambiguously describes test scenarios.**
- TTCN **precisely** tells:
  - sequence of all possible events during execution of a test case;
  - contents of PDUs sent to the IUT;
  - contents of PDUs expected from the IUT;
  - time frame for IUT to respond;
  - what IUT must do to get a PASS;
  - how can IUT get a FAIL or INCONCLUSIVE.

HEWLETT PACKARD

- Many benefits of Conformance Testing are possible because ATSs are written in this formal notation called TTCN.

- **PDU** = Protocol Data Unit (for example, an ISDN message, a Frame Relay frame, an ATM cell).



TTCN Features: Two possible forms

- 1st form: Graphical Representation (TTCN-GR). Used for editing and printing/publishing ATSs.

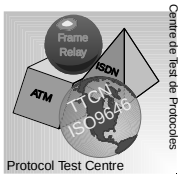
Test Case Dynamic Behaviour

Test Case Name	: Ver_VC
Group	: UNI_ATM_IS/General/
Purpose	: Verify that the IUT supports point-to-point VC connectivity.
Default	:
Comments	: Requires a VC connection Ref. 3.1

Label	Behaviour Description	Constraints Ref	Verdict	Comments
	PCO_A!CELL_NR	CELL_SQ(VPIvcca, VCIvcca,'01'O)		
	START T_Test			
LB1	PCO_B?CELL_NR	CELL_SQ(VPIvccb, VCIvccb,'01'O)	P	
	PCO_B?CELL	CELL_UNASSIGNED		
	GOTO LB1			
	?TIMEOUT T_Test		F	
	PCO_B?OTHERWISE		F	

HEWLETT PACKARD

- The examples in the following pages all use this TTCN-GR format.

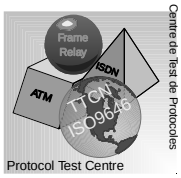


TTCN Features: Two possible forms

- 2nd form: Machine Processable form (TTCN-MP).
  - ASCII representation, thus providing an easy interchange format.
  - Completely equivalent to TTCN-GR.
  - Used to exchange ATSs between developers.
  - Follows a strict syntax. Thus, can be used as input to software tools (e.g. TTCN Translator).

HEWLETT PACKARD

- See next page for an excerpt from a TTCN-MP file.



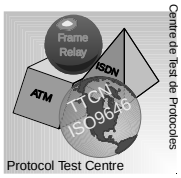
TTCN Features: Two possible forms

- Example of TTCN-MP:

```
...
$Begin_TestCase
  $TestCaseId Ver_VC
  $TestGroupRef UNI_ATM_IS/General/
  $TestPurpose
  /* Verify that the IUT supports point-to-point VC
     connectivity. */
  $DefaultsRef
  $Comment
  /* Requires a VC connection Ref. 3.1 */
  $BehaviourDescription
    $BehaviourLine
      $LabelId
      $Line [0]PCO_A!CELL_NR
      $CRef CELL_SQ(VPIvcca,VCIvcca,'01'O)
      $VerdictId
      $Comment /* */
    $End_BehaviourLine
...
```

HEWLETT PACKARD

- When compared with the Graphical Representation example (two pages ago), we can see that both TTCN-GR and TTCN-MP contents are exactly the same. The vertical and horizontal lines and headers in TTCN-GR are simply replaced by reserved keywords in TTCN-MP (words that start with "$").



TTCN Features: Sections of an ATS

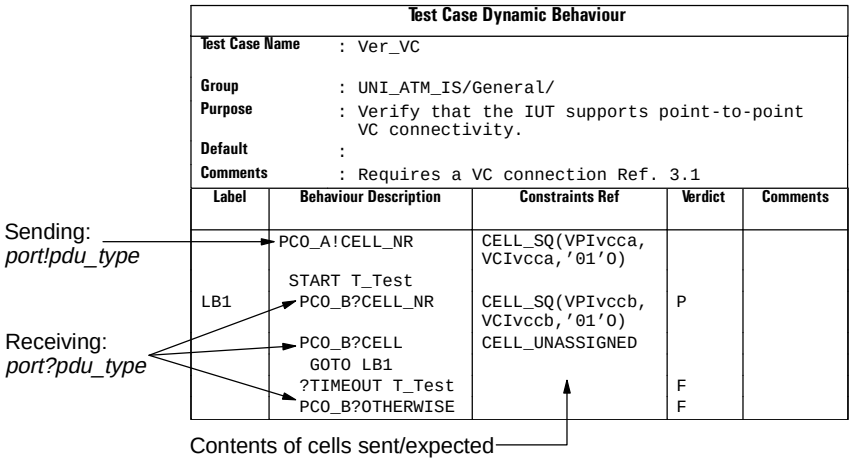
Section	What it provides
Test Suite Overview	Table of contents and Index to all test cases
Declarations	All definitions, PDU structure, etc.
Constraints	Contents of PDUs sent and expected
Dynamic Behavior	Actual test scenarios

HEWLETT PACKARD

- Every ATS in TTCN has four sections, described above. The examples in the following pages show the main tables in each section.
- Those examples are introduced in a logical order, not necessarily the order in which those tables appear in an actual ATS.

How to read an ATS in TTCN

- Sending/receiving PDUs to/from the IUT.

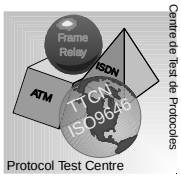


- Two key characters to remember when reading TTCN statements:

  ! means (using above example) **send** a PDU of type CELL_NR (with exact content defined by CELL_SQ) on port PCO_A **to** the IUT.

  ? means **wait** for a PDU of type CELL_NR on port PCO_B **from** the IUT.

- ?OTHERWISE means "any other type of PDU with any content".



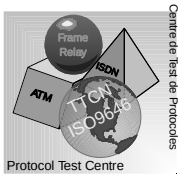
How to read an ATS in TTCN

- PDU Type: **CELL_NR**

PDU Type Definition		
PDU Name	: CELL_NR	
PCO Type	: PHYSAP	
Comments	:	
Field Name	**Field Type**	**Comments**
GFC	BITSTRING[4]	Generic Flow Control
VPI	BITSTRING[8]	Virtual Path ID
VCI	BITSTRING[16]	Virtual Channel ID
PTI	BITSTRING[3]	Payload Type ID
CLP	BITSTRING[1]	Cell Loss Priority
HEC	OCTETSTRING[1]	Header Error Check
Payload1	OCTETSTRING[1]	First payload, for the counter value
Payload2	OCTETSTRING[47]	Second payload

HEWLETT PACKARD

- The type CELL_NR defines a particular type of PDU (an ATM cell in this case).
- Add the length of every field above and you will get a total of 53 bytes.



How to read an ATS in TTCN

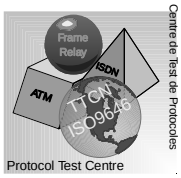
- PDU Content: **CELL_SQ**

PDU Constraint Declaration	
Constraint Name	: Cell_SQ(VPI_val,VCI_val:BITSTRING;N:OCTETSTRING)
PDU Type	: CELL_NR
Derivation Path	:
Comments	:

Field Name	Field Value	Comments
GFC	'0000'B	
VPI	VPI_val	
VCI	VCI_val	
PTI	'000'B	
CLP	'0'B	
HEC	Valid_HEC	
Payload1	N	
Payload2	PadOctet('00'O,47)	

HEWLETT PACKARD

- A constraint describes the exact content of a PDU (of a particular PDU Type; in the example above, CELL_NR).
- This CELL_SQ can then be used in the Dynamic Behavior section to either send it to the IUT or wait for it from the IUT.
- Notice that the 'Field Name's (1st column) must be exactly the same (same names, same order) as the 'Field Name's in the CELL_NR table.



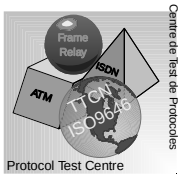
How to read an ATS in TTCN

- Basic semantics: TTCN Statements in Sequence.
  - TTCN statements in Sequence are indented once from each other.
  - When a statement is successful, control goes to the next statement in sequence.

```
PCO_A!CELL_NR
  START T_Test
    PCO_B?CELL_NR
```

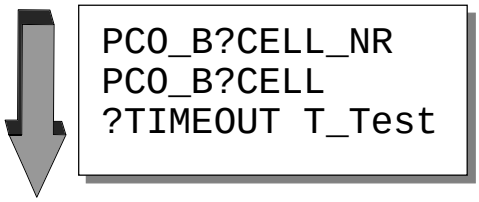
HEWLETT PACKARD

- Indentation is critical in the "Behaviour Description" column.
- A line indented from its previous line means that it will be executed **after if** and only if the previous line completed successfully.



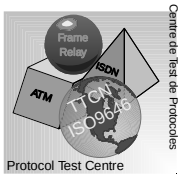
How to read an ATS in TTCN

- Basic semantics: Alternative TTCN Statements.
  - Statements at the same indentation level are possible alternative events.
  - Control loops from one alternative to the other until one of them is successful.
  - Then control goes to next statement in sequence following the successful event.



HEWLETT PACKARD

- Alternative statements imply a loop. If statements are badly coded, this may lead to an infinite loop! To avoid this, whenever you wait for a reply from the IUT:
  - start a timer before the loop (see example of timers later on this part),
  - make sure to put a "?TIMEOUT" as one of the last alternative,
  - also put a "?OTHERWISE" as the **last** alternative.



How to read an ATS in TTCN

- Verdicts

Test Case Dynamic Behaviour

Test Case Name	: Ver_VC
Group	: UNI_ATM_IS/General/
Purpose	: Verify that the IUT supports point-to-point VC connectivity.
Default	:
Comments	: Requires a VC connection Ref. 3.1

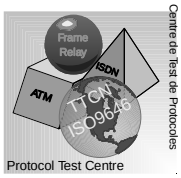
Label	Behaviour Description	Constraints Ref	Verdict	Comments
	PCO_A!CELL_NR	CELL_SQ(VPIvcca, VCIvcca,'01'O)		
	START T_Test			
LB1	PCO_B?CELL_NR	CELL_SQ(VPIvccb, VCIvccb,'01'O)	P	
	PCO_B?CELL	CELL_UNASSIGNED		
	GOTO LB1			
	?TIMEOUT T_Test		F	
	PCO_B?OTHERWISE		F	

If proper cell with proper content is received, IUT gets a **PASS**

If IUT does not respond within T_Test seconds or sends an incorrect cell, IUT gets a **FAIL**

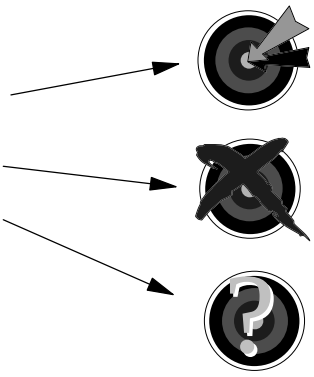
HEWLETT PACKARD

- If a test case is properly coded, every possible outcome of a test will render a verdict.
- Again, clear and unique verdicts are key features of conformance testing.



How to read an ATS in TTCN

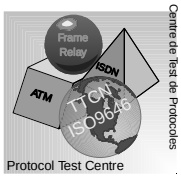
- Types of Verdicts:
  - Preliminary verdicts:
    - Pass = (P)
    - Fail = (F)
    - Inconclusive = (I)
  - Final verdicts:
    - Pass = P
    - Fail = F
    - Inconclusive = I
    - R (keep latest preliminary verdict assigned).



HEWLETT PACKARD

- If a preliminary verdict is encountered, verdict is recorded in memory and execution continues.
- Several preliminary verdicts can be encountered during execution. The final verdict reported by the test case will be the worst of those preliminary verdicts.
- Verdicts, from *good* to *bad*, are: PASS, INCONCLUSIVE, FAIL.

- If a final verdict is encountered, execution stops at this statement and the reported verdict is this final verdict, regardless of previous preliminary verdicts.



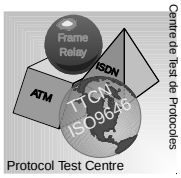
How to read an ATS in TTCN

- Basic semantics: End of Execution.
  - Execution stops when there are no more statements in sequence after a successful event.
  - Or execution stops when a final verdict is met.
  - A verdict must be assigned before execution stops.

If this event is successful, execution stops after PASS verdict is assigned.

Label	Behaviour Description	Constraints Ref	Verdict	Comments
	PCO_A!CELL_NR	CELL_SQ(VPIvcca, VCIvcca,'01'O)		
	START T_Test			
LB1	PCO_B?CELL_NR	CELL_SQ(VPIvccb, VCIvccb,'01'O)	P	
	PCO_B?CELL GOTO LB1	CELL_UNASSIGNED		
	?TIMEOUT T_Test		F	
	PCO_B?OTHERWISE		F	

HEWLETT PACKARD



How to read an ATS in TTCN

- Timers

Test Case Dynamic Behavior

Test Case Name	:	Ver_VC
Group	:	UNI_ATM_IS/General/
Purpose	:	Verify that the IUT supports point-to-point VC connectivity.
Default	:	
Comments	:	Requires a VC connection Ref. 3.1

Label	Behavior Description	Constraints Ref	Verdict	Comments
	PCO_A!CELL_NR	CELL_SQ(VPIvcca,VCIvcca,'01'O)		
	START T_Test			
LB1	PCO_B?CELL_NR	CELL_SQ(VPIvccb,VCIvccb,'01'O)	P	
	PCO_B?CELL	CELL_UNASSIGNED		
	GOTO LB1			
	?TIMEOUT T_Test		F	
	PCO_B?OTHERWISE		F	

Start timer T_Test

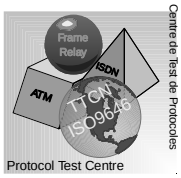
If timer T_Test expires

Timers can also be CANCELed

HEWLETT PACKARD

- Actions that can be performed on timers:
  START
  ?TIMEOUT
  CANCEL
  READTIMER

- All timers used in the Dynamic Behavior section must be defined in a separate table in the Declarations section.



How to read an ATS in TTCN

- Labels and GOTO statements.

Test Case Dynamic Behaviour

Test Case Name	: Ver_VC
Group	: UNI_ATM_IS/General/
Purpose	: Verify that the IUT supports point-to-point VC connectivity.
Default	:
Comments	: Requires a VC connection Ref. 3.1

Label	Behaviour Description	Constraints Ref	Verdict	Comments
	PCO_A!CELL_NR	CELL_SQ(VPIvcca, VCIvcca,'01'O)		
	START T_Test			
LB1	PCO_B?CELL_NR	CELL_SQ(VPIvccb, VCIvccb,'01'O)	P	
	PCO_B?CELL	CELL_UNASSIGNED		
	GOTO LB1			
	?TIMEOUT T_Test		F	
	PCO_B?OTHERWISE		F	

Label → LB1

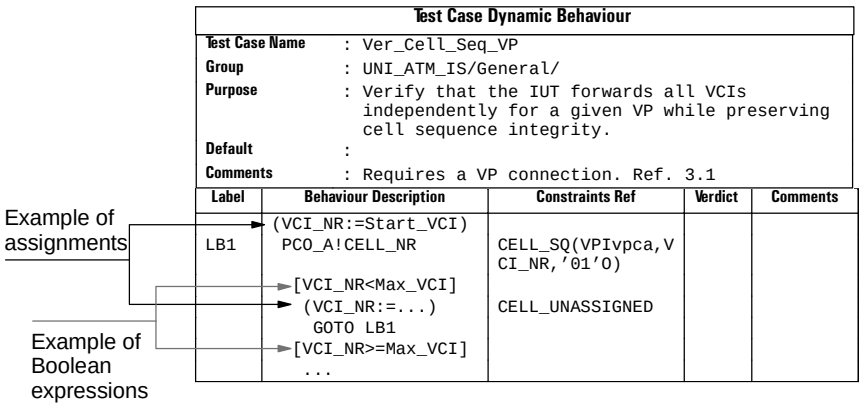
Used in GOTO → GOTO LB1

HEWLETT PACKARD

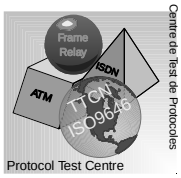
- GOTO provides a simple way to implement a loop.
- The TTCN syntax includes a REPEAT statement, but no WHILE or FOR statements (as in C language for example).

How to read an ATS in TTCN

- Assignments and Boolean Expressions.



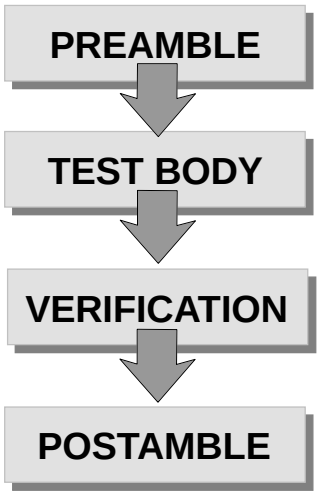
- All variables and constants must be defined in specific tables in the Declarations section.



How to read an ATS in TTCN

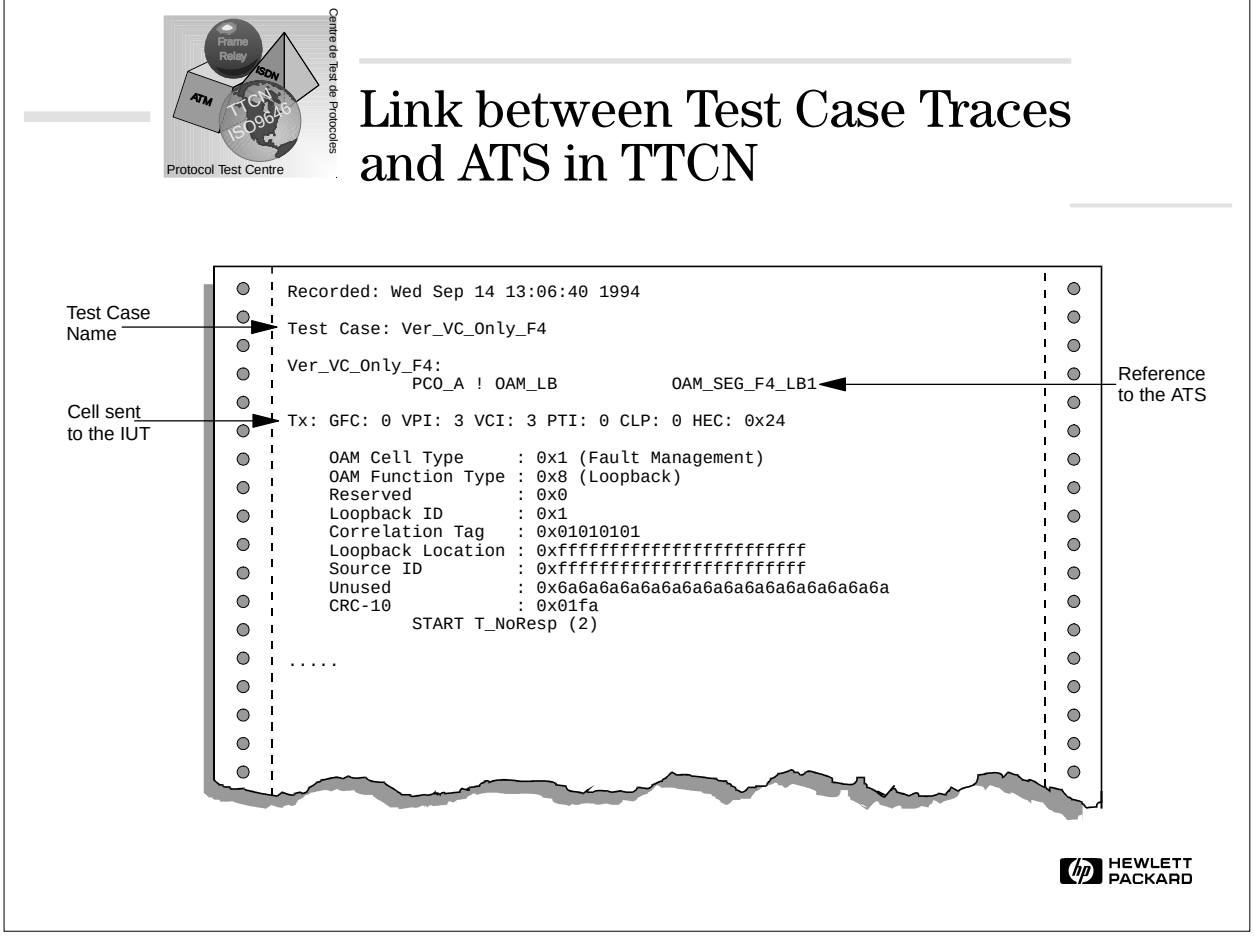
- Tree attachments (i.e. function calls) to Test Steps.

Test Case Dynamic Behavior				
Test Case Name	: PS1_03V			
Group	: T1617_Annex_D/Periodic_Polling/General/			
Purpose	: Verify that the IUT accepts a STATUS w/ full status report type containing PVC status IE identifying an unknown DLCI and the new bit set to 1 when the IUT is in state S1. The final IUT state is expected to be S2. Standard Ref.: D.4.3			
Default	:			
Comments	: Requires a VP connection. Ref. 3.1			
Label	**Behavior Description**	**Constraints Ref**	**Verdict**	**Comments**
	+PS1_PREAMBLE **+INCR_SN(SSN)** L!Status **+P_VERIFICATION** **+P_POSTAMBLE**	ST_V3(SSN,RSN)		

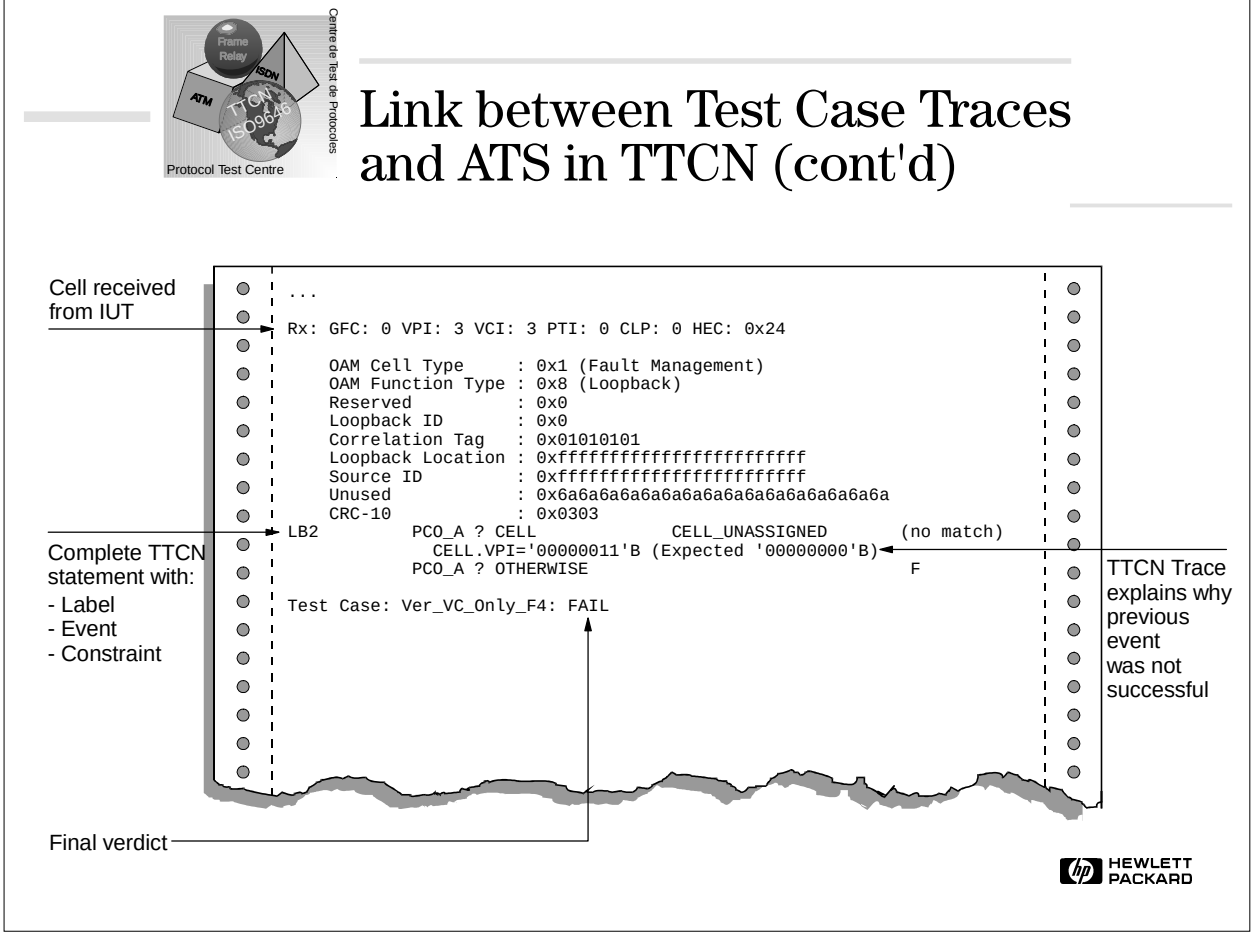


HEWLETT PACKARD

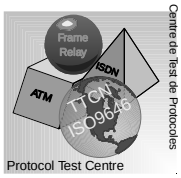
- Tree attachments (starts with the character '+') provide a clean way to modularize the code and are similar to function calls in regular programming languages.
- Test step tables are identical in syntax and semantic to test case tables.
- Execution can only start at the top of a test case.
- Tree attachment can only be made to a test step.



- A test case trace shows all PDUs sent and received, decoded in an easy-to-read format.
- Plus, it may contain (as in this HP test case trace example) statements that reproduce lines in the ATS in TTCN to help the test engineer follow the course of events and detect exactly where a problem occurred and what was expected, according to the ATS.



- HP test case traces also tell why a test case has failed and indicate what was expected instead, again referencing actual lines in the ATS.



TTCN: To Probe Further

- Standards:
  - ISO: *Information Technology - Open Systems Interconnection - Conformance Testing methodology and Framework - Part 3: The Tree and Tabular Combined Notation (TTCN)*.
  - ITU: *ITU-T Recommendation X.209*.

HEWLETT PACKARD