



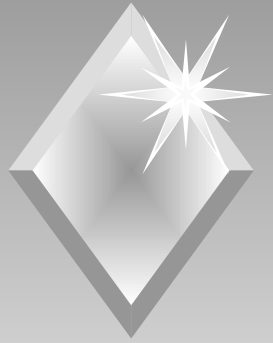
*Visualisation of TTCN test cases  
by MSCs*

Jens Grabowski

Universität Lübeck, ITM

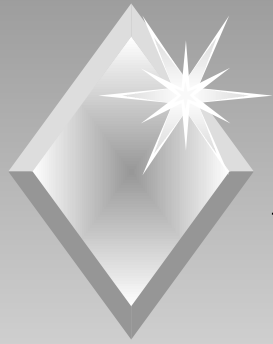
Thomas Walter

ETH Zürich



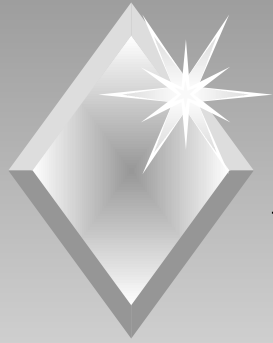
# *Contents*

- ◆ Introduction
- ◆ Specifying test purposes with MSCs
- ◆ Generating MSCs from TTCN
- ◆ Summary and Outlook



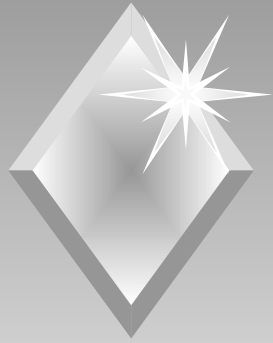
## *Introduction* (1/2)

- ◆ Concurrent TTCN is difficult to read
  - ◆ control is distributed over several test components
  - ◆ description is distributed over several tables
  
- ◆ Combined use of TTCN and MSC may improve the readability and understanding of TTCN test cases



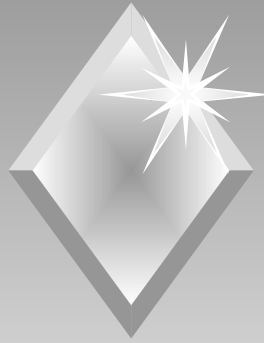
## *Introduction* (2/2)

- ◆ Basic MSCs may provide a horizontal view of selected test run examples
- ◆ HMSCs may describe the test case structure
- ◆ MSC/TTCN relationship is comparable to the MSC/SDL relationship
- ◆ MSC should not replace TTCN



## *Specifying Test purposes with MSCs*

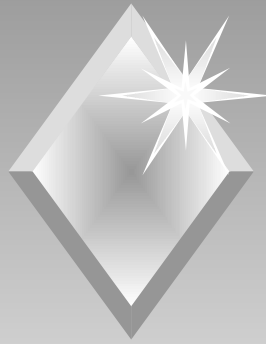
- ◆ Intelligent network example  
(test case specification for Core INAP CS-2)
- ◆ ISDN example



*Specifying test purposes with MSCs*

## *Intelligent Network example* (1/5)

- ◆ Test Suite development for Core INAP CS-2 is a three step procedure
  - ◆ informal test purpose description
  - ◆ formalisation of test purposes by MSCs  
(MSC development by using SDL simulation)
  - ◆ TTCN generation with tool support
- ◆ Informal test purposes and formal MSC test purposes are part of the protocol standard



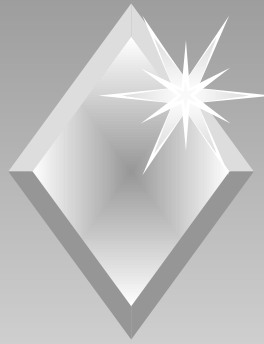
*Specifying test purposes with MSCs*

# *Intelligent Network example*

(2/5)

IN2_A_BASIC_AT_BI_01	
<b>Purpose:</b>	Test of invalid <b>Activity Test</b> (AT) invoke
<b>Requirement ref.:</b>	
<b>Preamble:</b>	O_S2P
<b>Test description:</b>	<b>Activity Test</b> invoke with argument sent by SCF to SSF during preamble
<b>Pass criteria</b>	SSF issues TC_error indicating unexpected Parameter
<b>Postamble</b>	SigConA_release_then_B

- ◆ Informal test purpose description

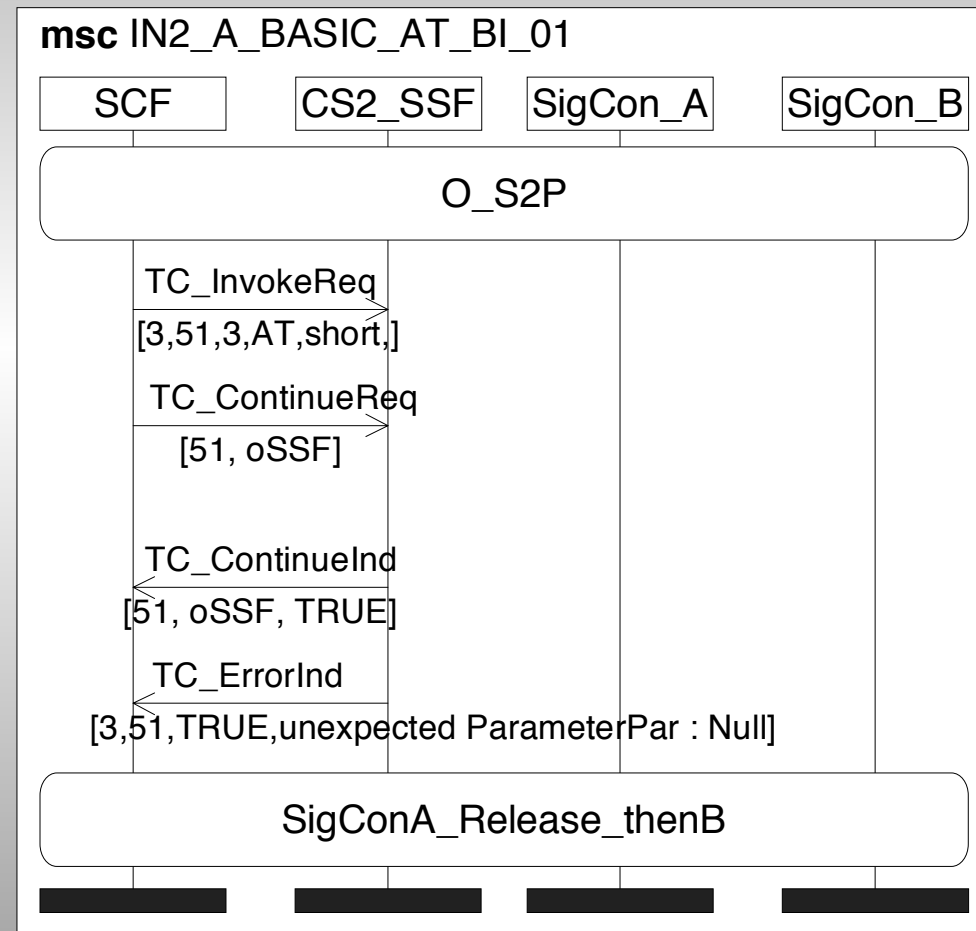


*Specifying test purposes with MSCs*

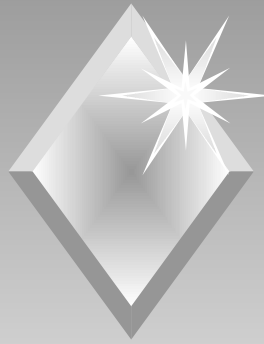
# *Intelligent Network example*

(3/5)

- ◆ Formal MSC test purpose description



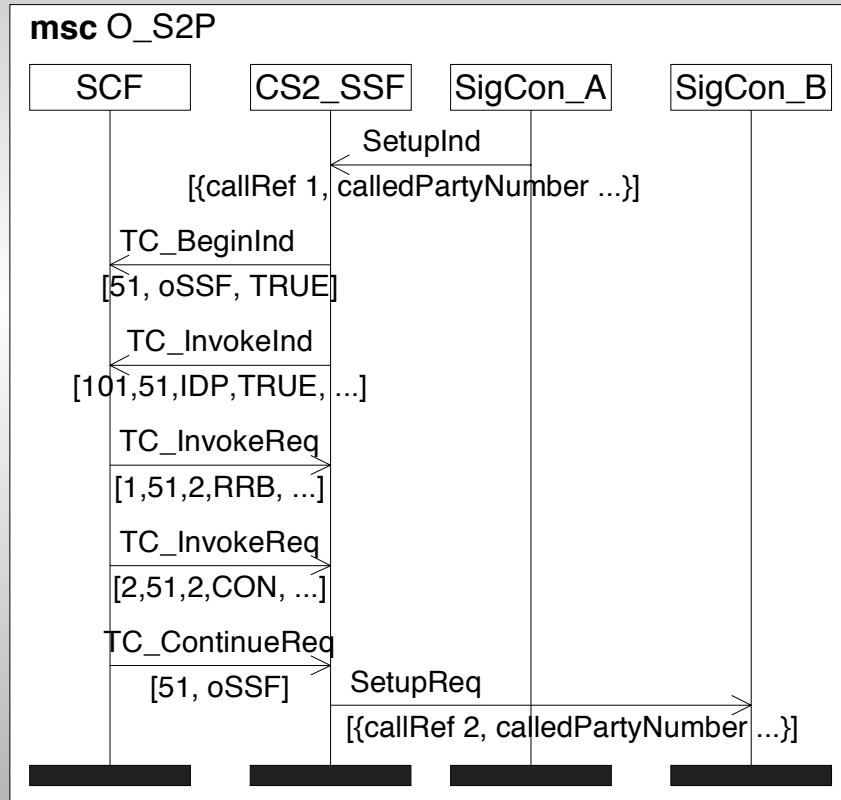




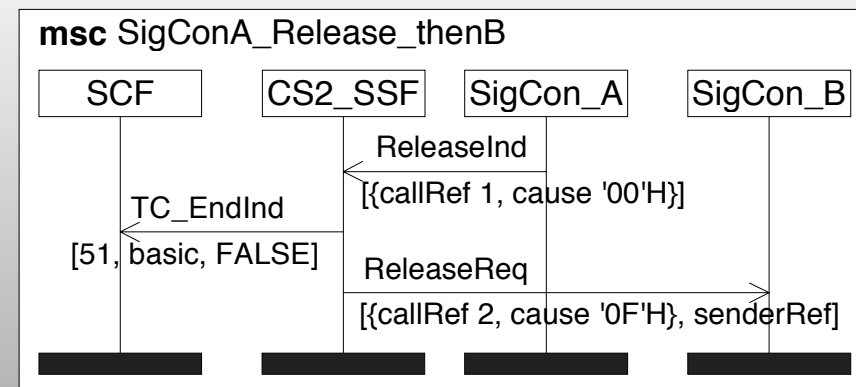
*Specifying test purposes with MSCs*

# *Intelligent Network example*

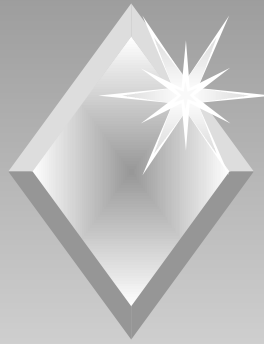
(4/5)



◆ MSC Preamble



◆ MSC Postamble



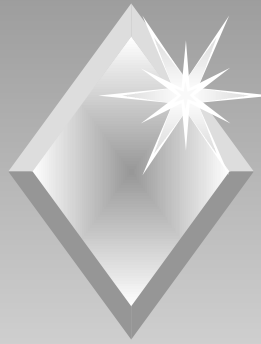
*Specifying test purposes with MSCs*

# *Intelligent Network example*

(5/5)

Test Case Dynamic Behaviour					
<b>Test Step Name:</b> IN2_A_BASIC_AT_BI_01					
<b>Purpose:</b>					
<b>Default:</b> OtherwiseFail					
Nr	L.	Behaviour Description	Constraints Ref.	Verdict	C.
1		+O_S2P			
2		SCF!TC_InvokeReq	CIR_ActivityTest_001 (PIX_Invokel3,Tsv_DialogId1)		
3		SCF!TC_ContinueReq	C_TC_ContinueReq (Tsv_DialogId1)		
4		SCF?TC_ContinueInd	C_TC_ContinueInd_001 (Tsv_DialogId1)		
5		SCF?TC_ErrorInd	C_TCErrIndunexpectedParameterPar (PIX_Invokeld3,TsvDialogId1)	(PASS)	
6		+SigConA_Release_thenB			
<b>Detailed Comments:</b>					

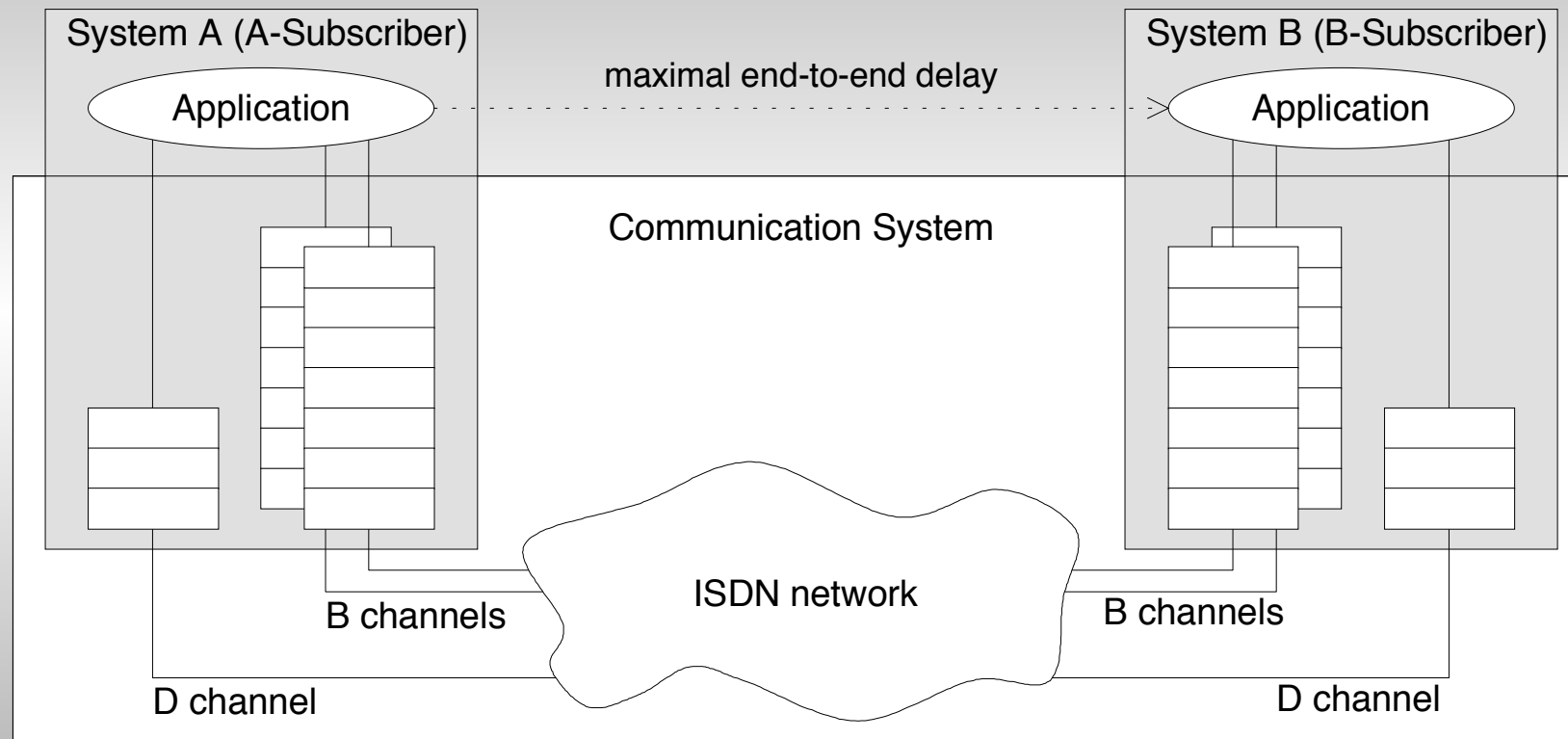
◆ Generated Core INAP CS-2 TTCN test case



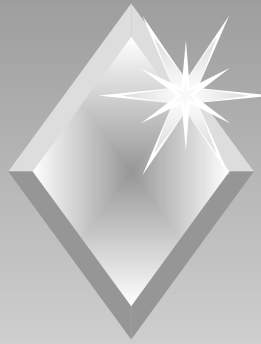
*Specifying test purposes with MSCs*

# *ISDN example*

(1/4)



## ◆ ISDN environment

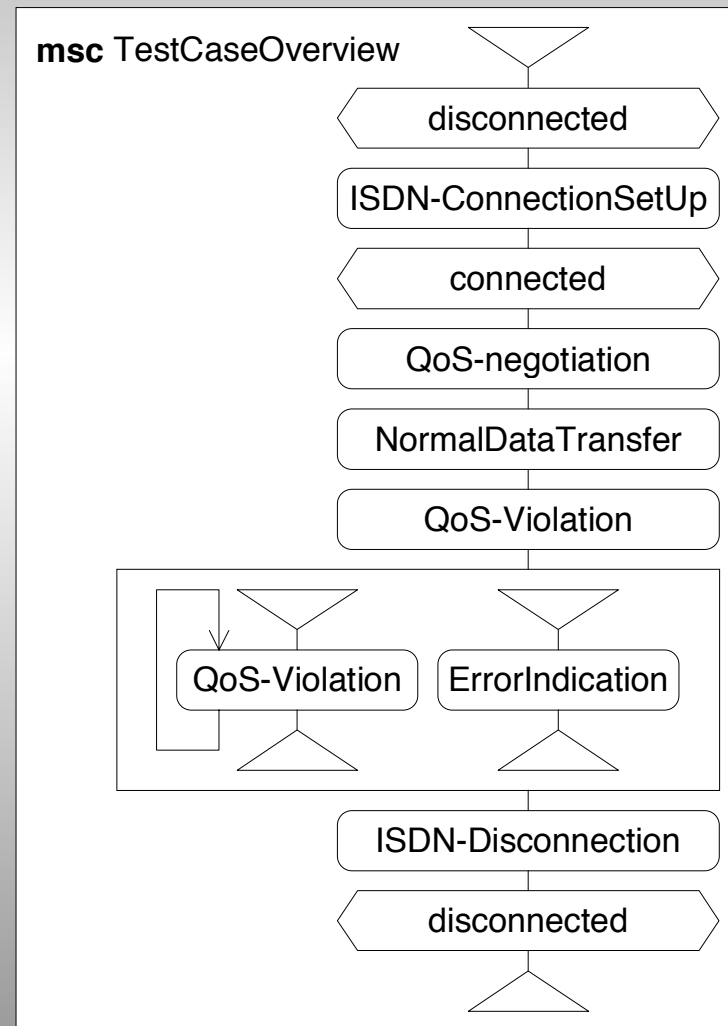


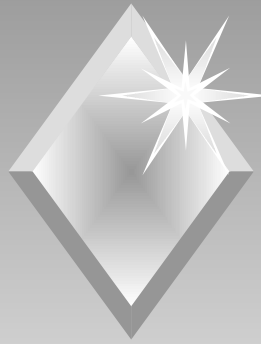
*Specifying test purposes with MSCs*

## *ISDN example*

(2/4)

- ◆ test case overview using HMSC for testing the correct behaviour in case of QoS violation



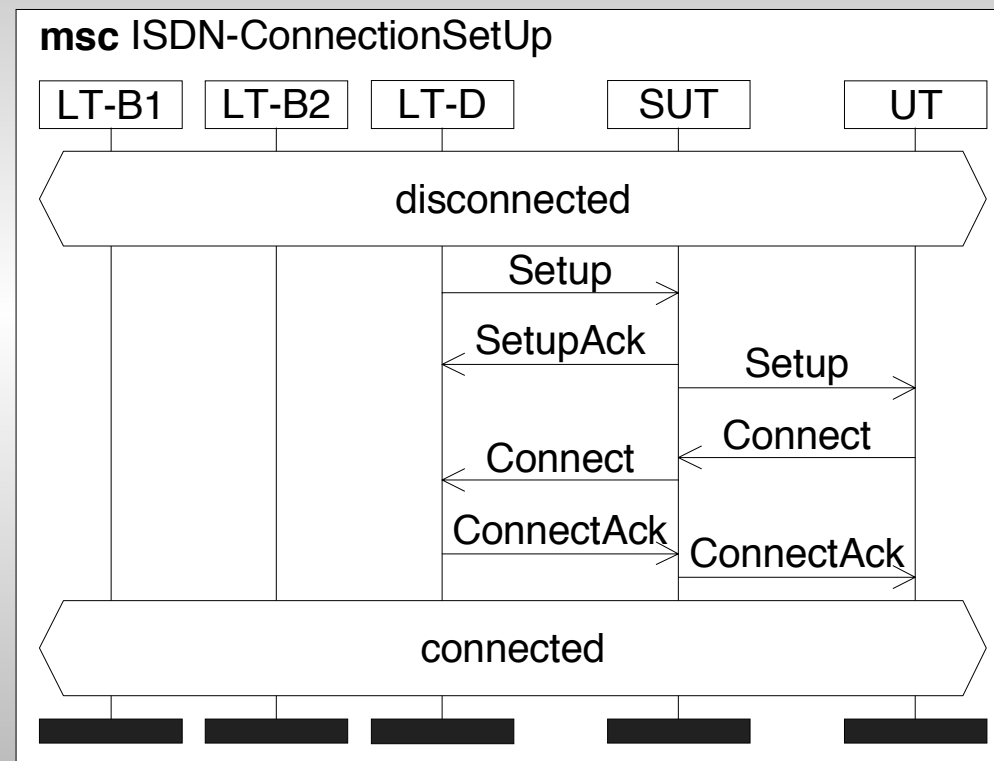


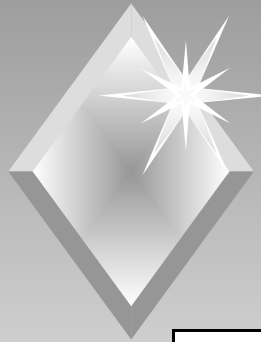
*Specifying test purposes with MSCs*

# *ISDN example*

(3/4)

- ◆ refined test behaviour described with basic MSC





*Specifying test purposes with MSCs*

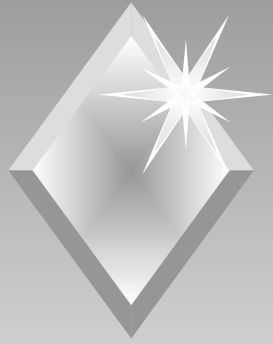
# *ISDN example*

(4/4)

Test Step Dynamic Behaviour					
<b>Test Step Name:</b> B-LT					
<b>Default:</b>					
Nr	Label	Behaviour Description	Constraints Ref.	Verdict	Comments
1	L1	CP1?CM1	C_Connected		RECEIVE
2		Start T1			Timer op.
3		? TIMEOUT			TIMEOUT
4		+QoSViolation1	ATTACH		
5		L!DATAreq	SEND		
6		GOTO L1	GOTO		
<b>Detailed Comments:</b>					

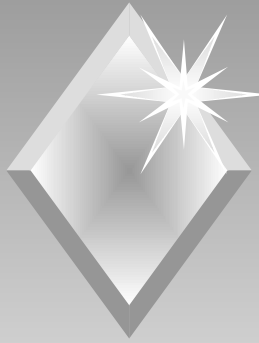
Test Step Dynamic Behaviour					
<b>Test Step Name:</b> D-LT					
<b>Default:</b>					
Nr	Label	Behaviour Description	Constraints Ref.	Verdict	Comments
1		[disconnected]	connected		Boolean
2		+ISDN_ConnectionSetUp			ATTACH
3		[connected]			
4		CP1!CM1			SEND
5		+ISDN_Disconnection			
<b>Detailed Comments:</b>					

◆ TTCN description



## *Generating MSCs from TTCN*

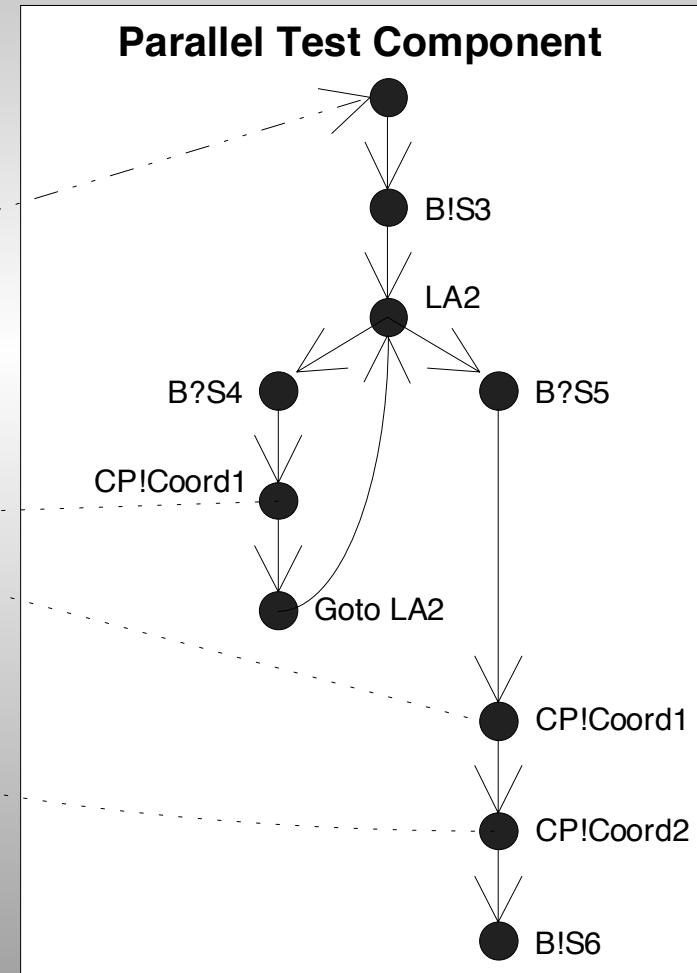
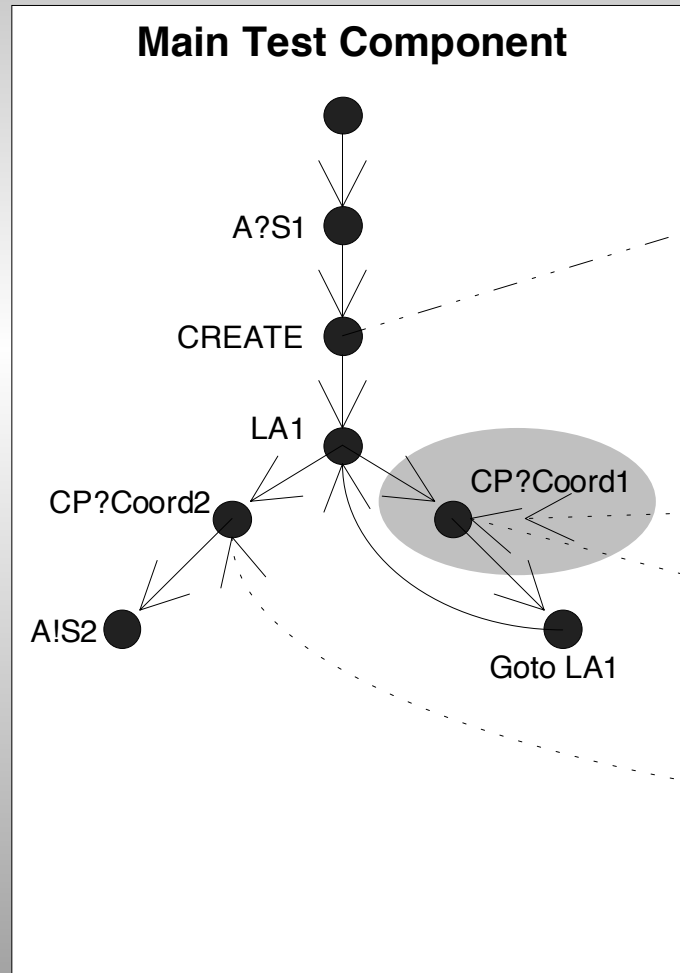
- ◆ Simulation versus direct mapping
- ◆ Problems of the simulation approach
- ◆ MSC representation



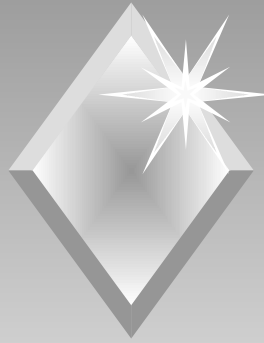
*Generating MSCs from TTCN*

# *Simulation versus direct mapping*

control  
structure  
of a  
concurrent  
test case



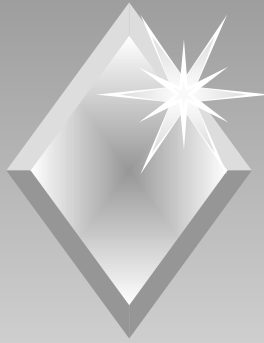




*Generating MSCs from TTCN*

## *Problems*

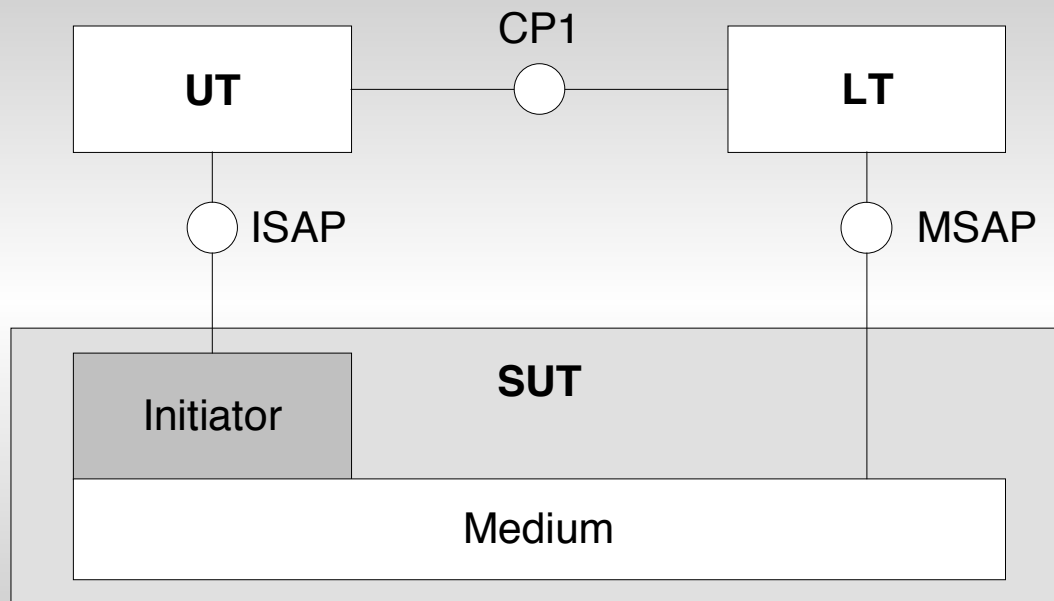
- ◆ Complexity due to simulation based on interleaving semantics
- ◆ Implicit complexity due to hidden default behaviours, test steps ...
- ◆ Data influence



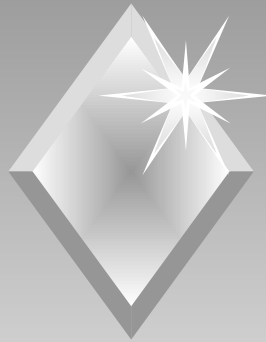
*Generating MSCs from TTCN*

# *Problems: Complexity*

(1/4)



- ◆ test architecture for testing the INRES initiator



Generating MSCs from TTCN

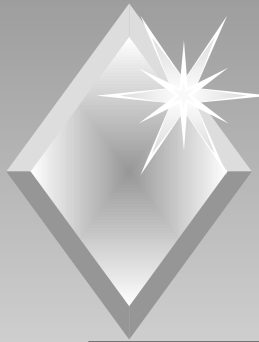
# Problems: Complexity

(2/4)

- ◆ TTCN example  
(only PASS  
behaviour,  
two test  
components)

Test Case Dynamic Behaviour					
<b>Test Step Name:</b> InresTestCaseExample					
<b>Configuration:</b> ConfigOne					
<b>Default:</b> OtherwiseFail					
Nr	Label	Behaviour Description	Constraints Ref.	Verdict	Comments
1		CREATE(LowerTester,PTCDescription)			
2		ISAP!CONreq			
3		ISAP?CONconf			
4		ISAP!DATreq (DATreq.S:=1)	DATreqdef		
5		CP1?RecDAT			
6		ISAP!DATreq (DATreq.S:=2)	DATreqdef		
7		CP1!SendDAT			
8		CP1?CorDAT		(PASS)	
9		ISAP?DISind			
<b>Detailed Comments:</b>					

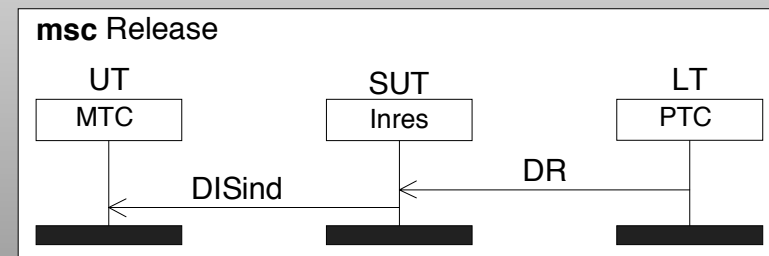
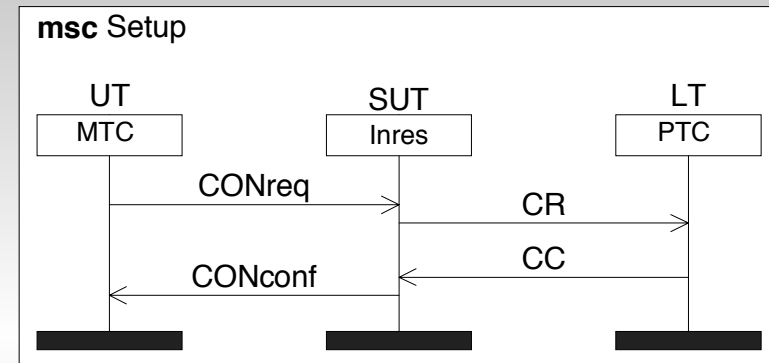
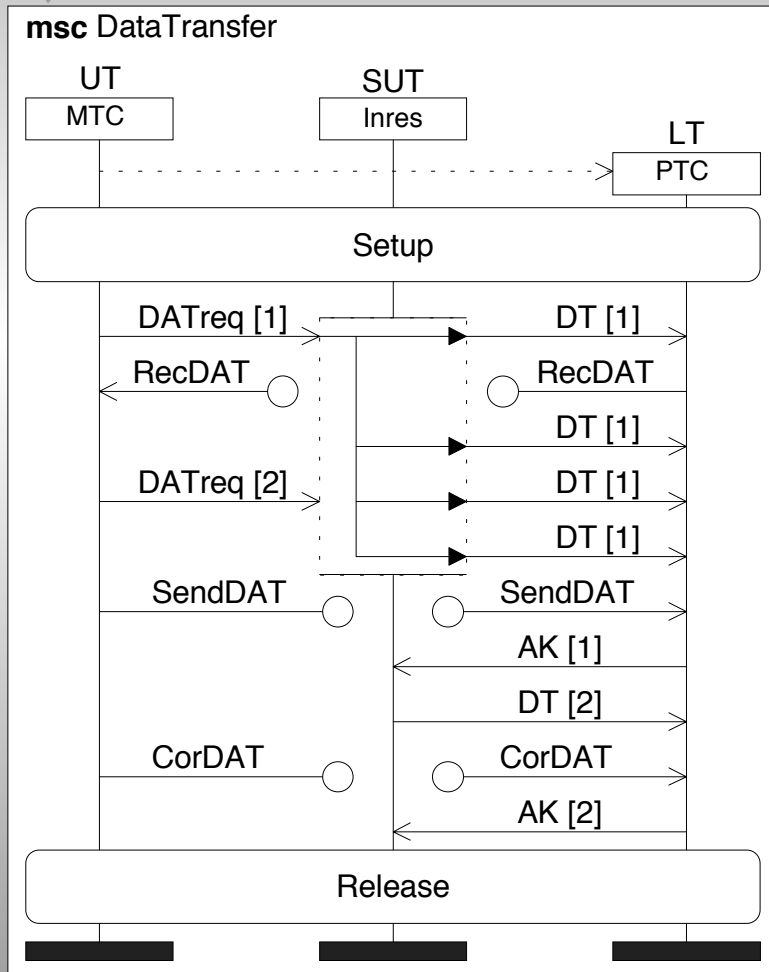
Test Step Dynamic Behaviour					
<b>Test Step Name:</b> PTCDescription					
<b>Default:</b> OtherwiseFail					
Nr	Label	Behaviour Description	Constraints Ref.	Verdict	Comments
1		MSAP?CR			
2		MSAP!CC			
3		MSAP?DT (S:=DT.S)	DTdef		
4		CP1!RecDAT			
5		MSAP?DT [S=DT.S]	DTdef		
6		MSAP?DT [S=DT.S]	DTdef		
7		MSAP?DT [S=DT.S]	DTdef		
8		CP1?SendDAT			
9		MSAP!AK (AK.Nr:=S)	Akdef		
10		MSAP?DT [DT.S = S + 1]	DTdef		
11		CP1!CorDAT			
12		MSAP!AK (AK.Nr:=S+1)	Akdef		
13		MSAP!DR			
<b>Detailed Comments:</b>					

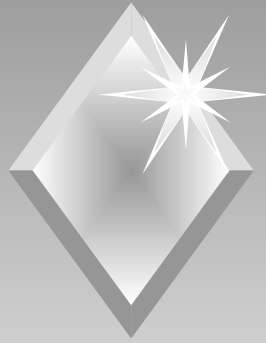


Generating MSCs from TTCN

# Problems: Complexity

(3/4)



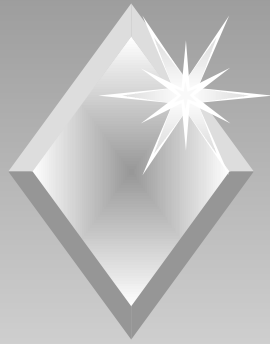


*Generating MSCs from TTCN*

## *Problems: Complexity*

(4/4)

- ◆ more than 9000 interleaved sequences of events
- ◆ one possible solution '*partial order simulation*' of TTCN test cases

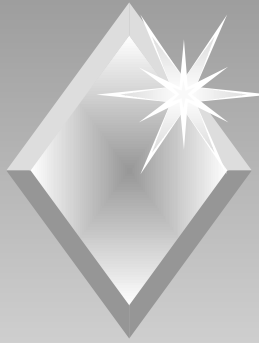


*Generating MSCs from TTCN*

## *Mapping TTCN events onto MSC events*

<b>TTCN Construct</b>		<b>MSC Construct</b>
CREATE	⇒	create
SEND	⇒	message output
RECEIVE	⇒	message input
Assignments	⇒	tasks
timer operations	⇒	timer operations (TTCN READTIMER is mapped onto MSC task)
qualifiers	⇒	conditions or comments

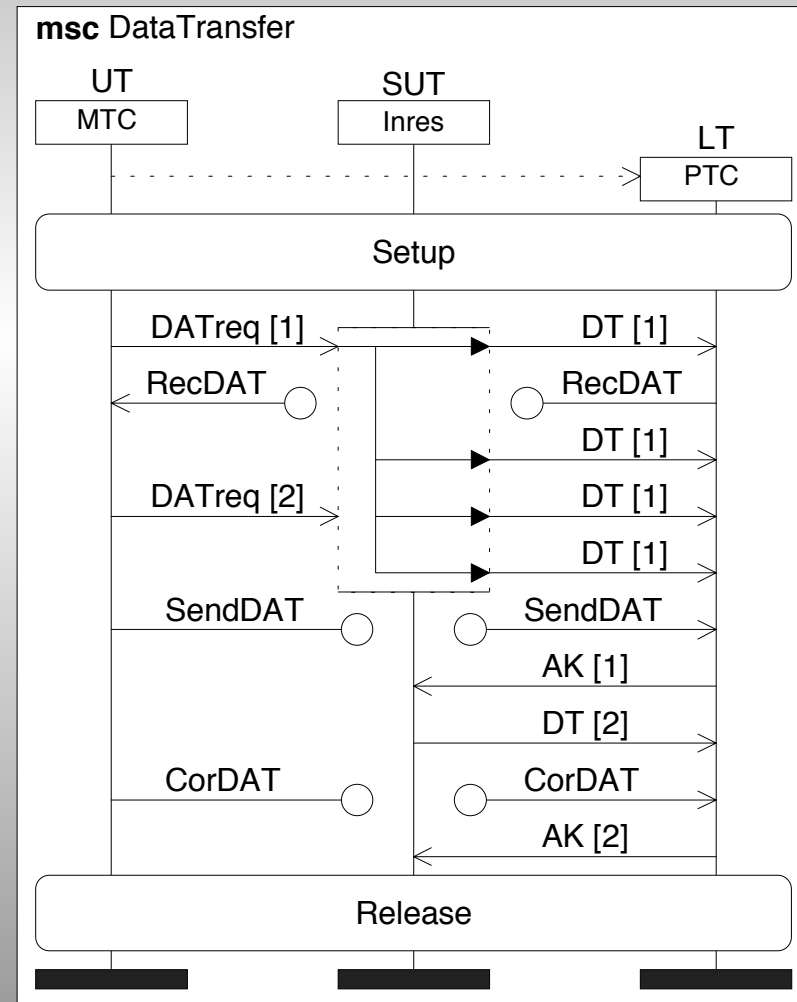
Message inputs and outputs of the environment are not described in TTCN but can be added automatically in MSC

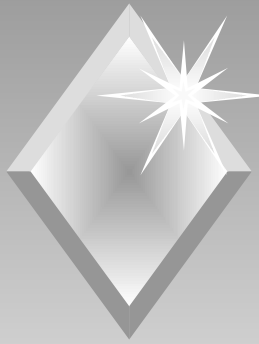


*Generating MSCs from TTCN*

# *MSC representation (1/3)*

- ◆ expected view, representation of SUT and test components (abstraction from PCOs and CPs)

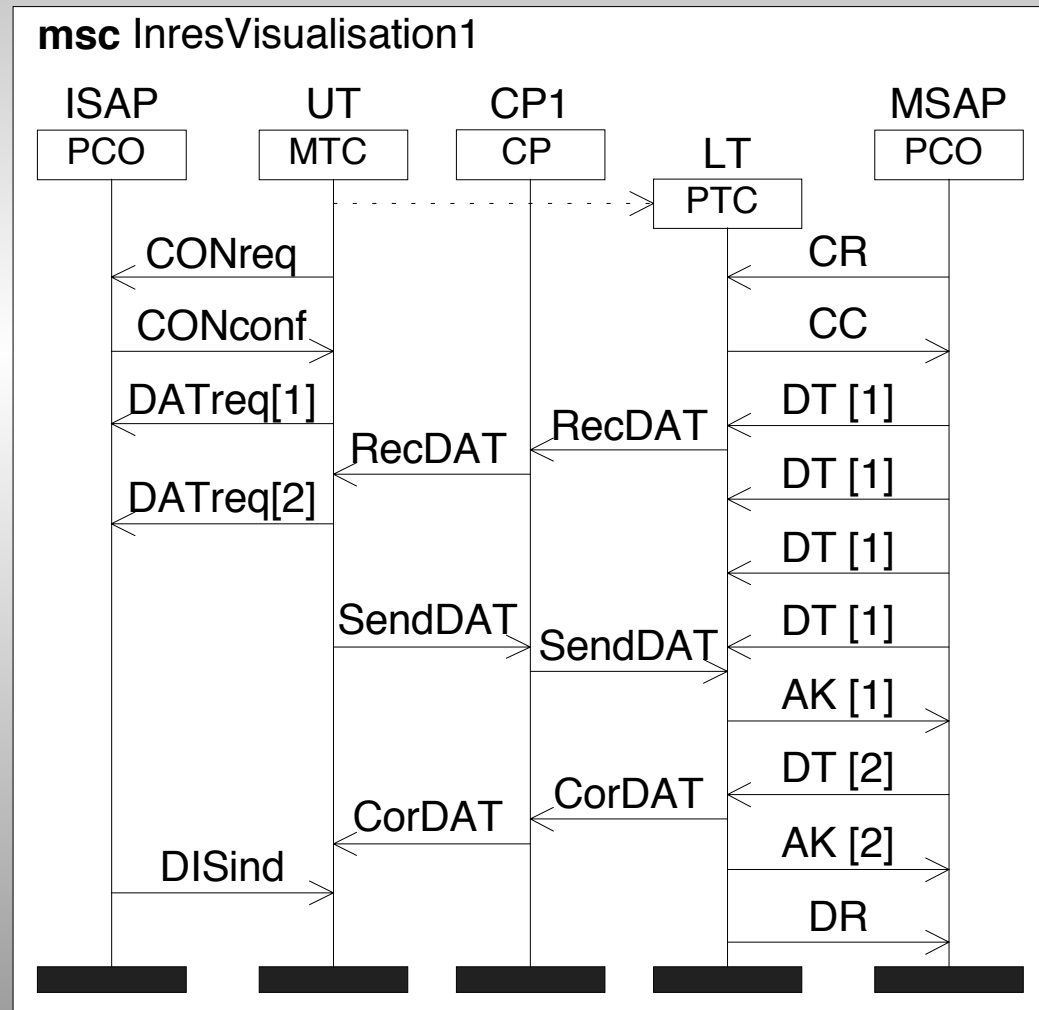




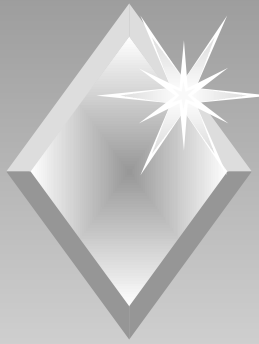
Generating MSCs from TTCN

# MSC representation (2/3)

- ◆ representation of PCOs, CPs and test components (abstraction from IUT)



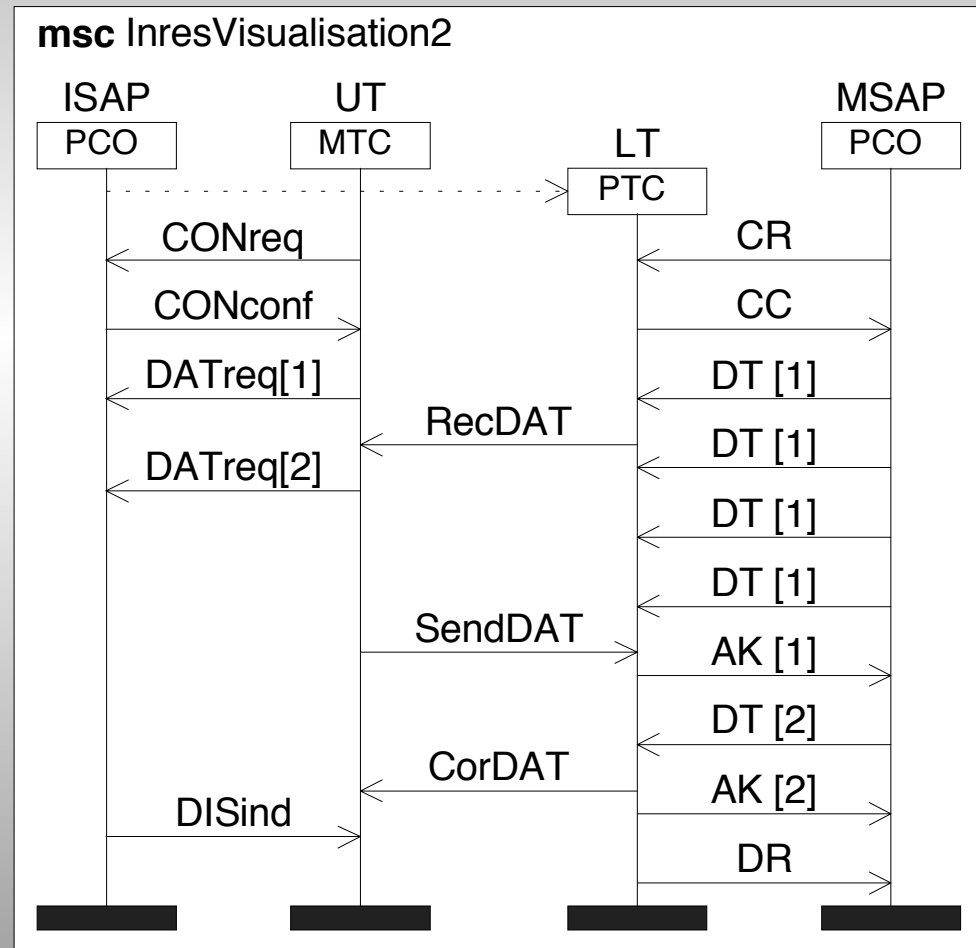


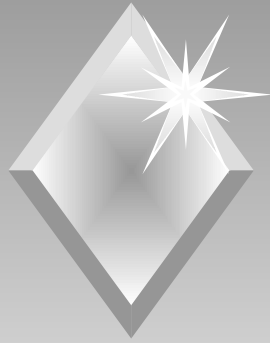


Generating MSCs from TTCN

# MSC representation (3/3)

- ◆ representation of PCOs and test components (abstraction from IUT and CPs)





## *Summary and Outlook*

- ◆ We discussed the possibilities and problems of improving the readability of TTCN test cases by using MSCs as additional notation.
- ◆ Our conclusion is that the approach is promising, especially if tools support the visualisation.
- ◆ Our future work will focus on solving the identified problems and on the use of HMSC constructs for MSCs generated by means of TTCN simulation.