



IFTT

INTERFACE TESTING TOOL

Automatic Interface Regression Tests
for SAP Middleware and SAP Backend systems

- ◆ Int4 Interface Testing Tool (int4 IFTT) is an automatic SAP interface testing software, provided as an ABAP backend system add-on.
- ◆ Officially certified by SAP:
 - ◇ for integration with SAP NetWeaver 7.40 via the SAP integration scenario ABAP Add-On Deployment for SAP NetWeaver.
 - ◇ for integration with SAP S/4HANA on-premise edition 1511 via the SAP integration scenario ABAP Add-On Deployment for SAP S/4HANA.
- ◆ It fully tests **SAP interface development after any source code / configuration changes**

Interface testing – challenges

- ◆ Time-consuming testing with manual replication of documents. Small number of test cases
- ◆ Tests of specific components (mostly PI mappings)
- ◆ Test results are limited to laborious and error-prone checks



Complete testing scope:
middleware
+
backend system

Quick and simple assertion rules setting.
do it once –
no need to change

Reduced downtime
for Integration architects,
functional experts
and developers

Support for all project phases
(waterfall / agile)

Reduced cost
of external resources.
No need to generate
new data

100% document accuracy.
Validation based
on the final data

**More funds, time
and resources available**
for improvements



**Creation of test cases
based on existing
documents**



**Full SAP landscape
test scope**



**Single test case
allows to verify
all possible checks**



**Document comparison
performed on the DB
level (per configuration)**

Process-oriented approach

categorizing test cases by business processes

Single test case and full scenario processing modes

SAP PO adapter independent
(MQ, SOAP, FTP, etc)

Testing both inbound & outbound scenarios.
Integration with ECATT

User-friendly comparison
OK / FAILED report.
Comparison based on the final business docs

Open architecture
to support other integration platforms
(HCI, AIF)

Middleware layer – SAP PO (mappings, data structures, BPM, adapter modules, etc.)

Interface Program layer in SAP backend (AIF, IDOC, BAPI, PROXY, custom ABAP)

Configuration layer (all customizing settings from transaction SPRO that may impact the interface)



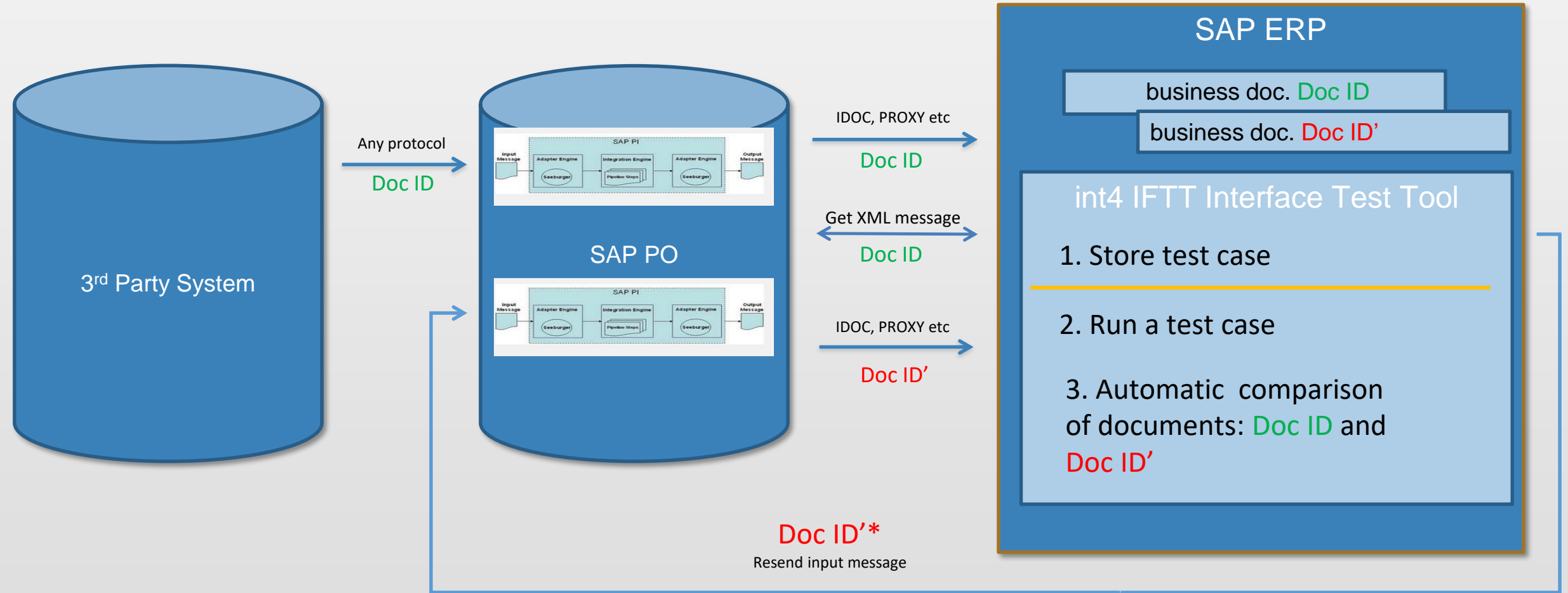
1

Create test cases automatically by selecting business documents already created in the receiving application.

2

Run the test case and automatically validate the new created business documents by comparison with the originals.





*External message ID is randomized each time

In order to **create a test case**, start with triggering the message by:

1

posting a business document using an eCATT recording



using another inbound interface*

2

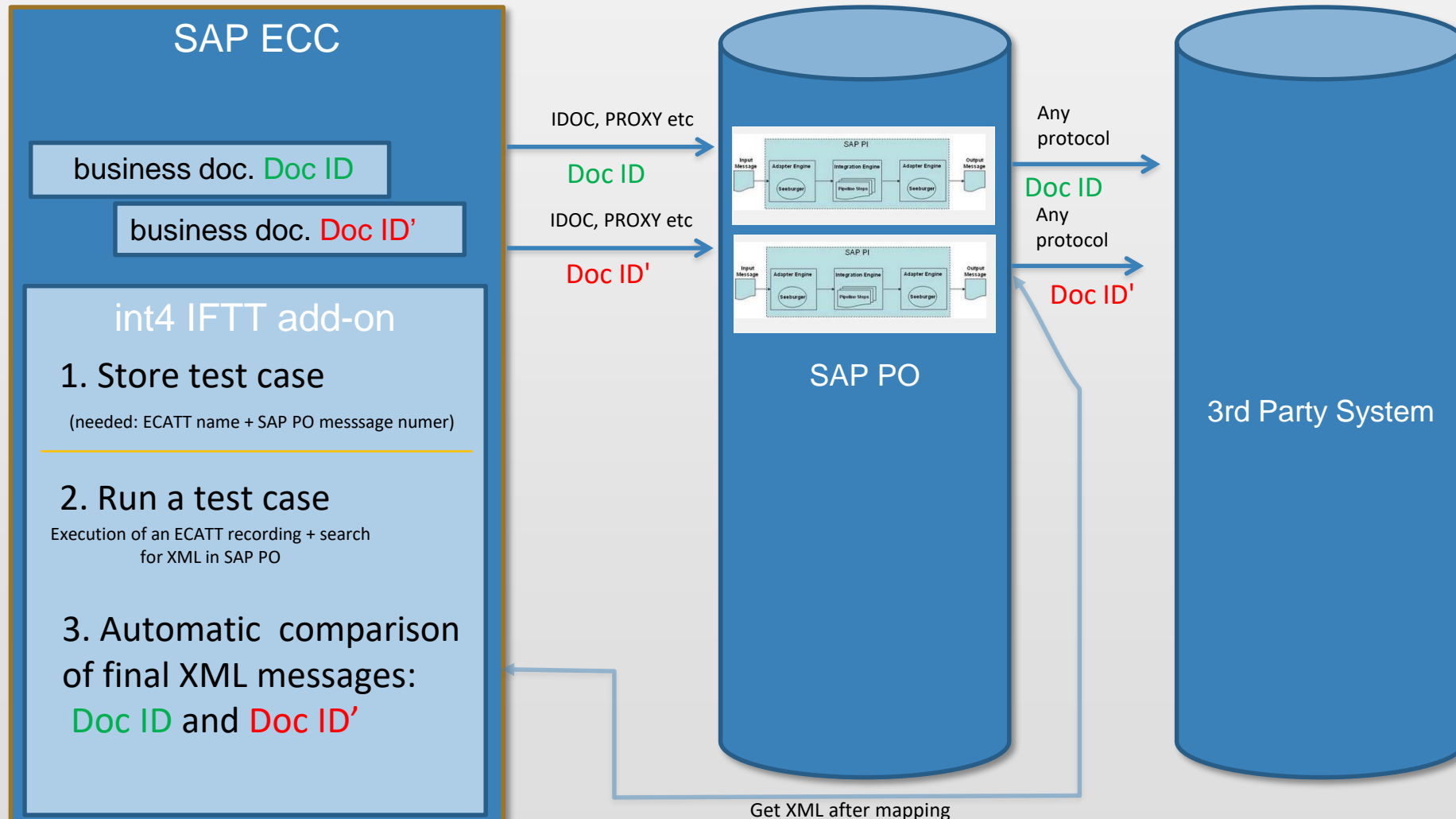
Then, use an outbound message to store the original **XML** and **associated business object name** for future assertions.

3

Run the test case and automatically validate the XML.



*(i.e. inbound sales order triggers outbound sales order acknowledgement.)



Test management

IFTT: Display Mode - Test Cockpit

Scenario Selection

Scenario Selection	Attr1	Attr2	Attr3	Attr4	Attr5	Date	Created at	User
<input type="checkbox"/> FI PROJECT 2 (ID:000000001)	CRM	AFTER REFRESH				24.06.2016 17:02:20	INT4	
<input type="checkbox"/> EDI DELIVERY (ID:000000002)	SAP EY1 POST CLOSE	MU TEST2				16.06.2016 03:32:17	INT4	
<input type="checkbox"/> VENDOR MASTER (ID:000000003)	SAP PS4 AFTER REFRESH TT					24.06.2016 16:48:07	INT4	
<input type="checkbox"/> BANK MASTER (ID:000000004)	SAP PS4 POST CLOSE					24.06.2016 17:40:31	INT4	
<input type="checkbox"/> WEBSHOP XYZ (ID:000000006)						24.06.2016 13:24:22	INT4	
<input checked="" type="checkbox"/> ECatt integration Demo (ID:000000015)						02.09.2016 16:01:29	MKOWALCZEWSK	
<input checked="" type="checkbox"/> outbound testing (ID:000000153)						13.09.2016 17:50:21	MKOWALCZEWSK	
<input checked="" type="checkbox"/> TECHED (ID:000000156)	VEGAS	2016				15.09.2016 14:44:12	MKOWALCZEWSK	

Node Description | Case ID | Description | Interface Type | DocumentNumber | ObjectDef.ID | pre caseID | Date

Node Description	Case ID	Description	Interface Type	DocumentNumber	ObjectDef.ID	pre caseID	Date
BANK MASTER (ID:000000004)	000000664	TEST3	IDoc msg number (P)	3246931	DELIVERY	0022	08.09.2016
WEBSHOP XYZ (ID:000000026)	000000351	BOOK M-13	IDoc msg number (P)	000000003228258	SALES_ORD		21.06.2016
WEBSHOP XYZ (ID:000000026)	000000361	Mat Testcase 2	IDoc msg number (P)	000000003228258	INVOICE		06.07.2016
WEBSHOP XYZ (ID:000000026)	000000441	Test After Cleanup	IDoc msg number (P)	000000003228258	INVOICE		10.08.2016
WEBSHOP XYZ (ID:000000026)	000000650	nowy	IDoc msg number (P)	000000003228258	INVOICE		06.09.2016
ECatt integration Demo (ID:000000152)	000000656	Manual creation of Purchase order (me21n)	ECATT recording	ZMKO_JFTT_PO_001	EPO_ME21N		08.09.2016
ECatt integration Demo (ID:000000152)	000000675	Testing of inbound deliver from EDI system	IDoc msg number (P)	000000003246972	DELIVERY	000000656	08.09.2016
outbound testing (ID:000000153)	000000771	inbound sales order#1	IDoc msg number (P)	000000003247823	SALES_ORD		19.09.2016
outbound testing (ID:000000153)	000000772	outbound #2	PI Outbound	00029a75f071ee69cfc3fbc6c9e9a	OUTB_SO	000000771	19.09.2016
TECHED (ID:000000156)	000000707	new test for Bill	IDoc msg number (P)	000000003247791	SALES_ORD		15.09.2016
TECHED (ID:000000156)	000000728	new test for Micheal	IDoc msg number (P)	000000003247791	SALES_ORD		16.09.2016
TECHED (ID:000000156)	000000767	new demo MKR	IDoc msg number (P)	3248842	SALES_ORD		19.09.2016

Summarized execution report

IFTT Test Execution Report

Scenario name: ECatt integration Demo (ID:000000152) executed on: 20.09.2016 at: 16:23:47

Test case: 0000000656 created: 08.09.2016 17:06:26 by: MKOWALCZEWSK Description: Manual creation of Purchase order (me21n) **Passed**

Test case: 0000000675 created: 08.09.2016 17:45:25 by: MKOWALCZEWSK Description: Testing of inbound deliver from EDI system **Passed**

Scenario name: outbound testing (ID:000000153) executed on: 20.09.2016 at: 16:23:47

Test case: 0000000771 created: 19.09.2016 22:48:28 by: MKOWALCZEWSK Description: inbound sales order#1 **Passed**

Test case: 0000000772 created: 19.09.2016 22:48:56 by: MKOWALCZEWSK Description: outbound #2 **Failed**

Go to detailed results: 0000000772

Scenario name: TECHED (ID:000000156) executed on: 20.09.2016 at: 16:23:49

Test case: 0000000707 created: 15.09.2016 14:47:28 by: MKOWALCZEWSK Description: new test for Bill **Passed**

Test case: 0000000728 created: 16.09.2016 10:07:28 by: MKOWALCZEWSK Description: new test for Micheal **Passed**

Test case: 0000000767 created: 19.09.2016 15:33:40 by: MKOWALCZEWSK Description: new demo MKR **Passed**

Detailed report of assertions for inbound interface*

IFTT Test Execution Report

Scenario name: Sales escenario (ID:000000180) executed on: 11.11.2016 at: 14:49:07

Test case: 0000000965 created: 03.11.2016 11:41:13 by: MKOWALCZEWSK Description: SO #1 - four items order **Passed**

Sales Document: Header Data (VBAK)

Sales Doc.	PO date	Sale	DU	DI	Sal	Sale	Curre	Net value	Price p	CO A	CCod	PO %	On	Time
22209	03.11.2016	3000	10	00	311	3010	USD	23.537,50	RVKXUS	2000	3000	DFRUS	11.11.2016	11:38:21
22322	03.11.2016	3000	10	00	311	3010	USD	23.537,50	RVKXUS	2000	3000	DFRUS	11.11.2016	14:48:59

Sales Document: Item Data (VBAK)

Sales Doc.	Item	Material	Mat1 Group	Desc.	Item	DU	Bus.	Net value	Curre	Order qty	Sal	Gross
22209	000010	M-12	02027	MAG DX 15P/Pe	TAR	A	07	7000	USD	4.495,00	5,000	ST
22322	000010	M-12	02027	MAG DX 15P/Pe	TAR	A	07	7000	USD	4.495,00	5,000	ST
22209	000020	M-10	02027	Flatacreeen MS 1775P	TAR	A	07	7000	USD	7.245,00	5,000	ST
22322	000020	M-10	02027	Flatacreeen MS 1775P	TAR	A	07	7000	USD	7.245,00	5,000	ST
22209	000030	M-10	02027	Flatacreeen MS 1775P	TAR	A	07	7000	USD	7.245,00	5,000	ST
22322	000030	M-10	02027	Flatacreeen MS 1775P	TAR	A	07	7000	USD	7.245,00	5,000	ST
22209	000040	M-14	02027	MAG PA/DX 175	TAR	A	07	7000	USD	4.552,50	5,000	ST
22322	000040	M-14	02027	MAG PA/DX 175	TAR	A	07	7000	USD	4.552,50	5,000	ST

Test case: 0000000974 created: 04.11.2016 11:43:43 by: MKOWALCZEWSK Description: SO #2 - stand. domestic order **Passed**

Sales Document: Header Data (VBAK)

Sales Doc.	PO date	Sale	DU	DI	Sal	Sale	Curre	Net value	Price p	CO A	CCod	PO %	On	Time
22221	03.11.2016	3000	10	00	311	3010	USD	266.676,00	RVKXUS	2000	3000	DFRUS	11.11.2016	11:42:53
22321	03.11.2016	3000	10	00	311	3010	USD	266.676,00	RVKXUS	2000	3000	DFRUS	11.11.2016	14:48:59

Sales Document: Item Data (VBAK)

Sales Doc.	Item	Material	Mat1 Group	Desc.	Item	DU	Bus.	Net value	Curre	Order qty	Sal	Gross
22221	000010	M-15	02027	SEC Multisync XV15	TAR	A	07	7000	USD	266.676,00	213,000	ST
22321	000010	M-15	02027	SEC Multisync XV15	TAR	A	07	7000	USD	266.676,00	213,000	ST

Detailed report of assertions for outbound interface

System Help

XML comparator tool

Next difference Previous difference

ns0:MT_DocumentOrderResponse

- @ns0 http://int4.com:PurchaseOrderAcknowledgement
- DocumentOrderResponse
 - OrderResponseHeader
 - OrderResponseNumber
 - 000000003251176
 - OrderResponseDate
 - 20160414
 - OrderNumber
 - 30759079
 - OrderDate
 - 20161013
 - ResponseType
 - 20

*Presents selected fields of original document (light green) compared with the current document (dark green).



- ◆ **int4 IFTT** business objects represent assertions for business document as well as SAP PO technical details of interface (examples include: Sales invoice, inbound delivery, outbound delivery, etc.)
- ◆ For inbound interfaces the assertions are defined as a model of database tables and fields to be checked each time during the test case execution. Usually, they cover all aspects of business document, so there is no need for extensive planning during the test case creation. This means the results are always comprehensive and not limited to single checks (for example instead of checking VAT only, the whole invoice is verified)
- ◆ For outbound interfaces, the assertion is made based on comparing the original XML with the one produced during the test case execution with configured exception list managed by XPath expressions (for example the invoice number will always be different and will not fail the test case)
- ◆ Every test case must be associated with a single business object

- ◆ **eCATT** is a standard built-in SAP tool. It offers a graphical user interface with ABAP script editor and its own command syntax. It allows for recording creation of business documents.
- ◆ **int4 IFTT** integrates with **eCATT** in order to:
 - ◇ Implement scenarios where the business process starts in SAP. A good example is testing a inbound delivery interface, where each time a fresh purchase order is required as SAP blocks goods receipt over the limit. The **eCATT** recording contains creation of a particular purchase order and it is executed by **int4 IFTT** each time before the inbound delivery test case execution
 - ◇ Test outbound scenarios. Outbound scenarios need a trigger to create a message (IDoc, proxy, rfc etc). These messages are generated by recorded executions of business transactions. Hence **int4 IFTT** tests also all customizing and ABAP development (user-exits) needed to trigger outbound messages from SAP
- ◆ Thanks to the **eCATT** integration, all scenarios that require manual user actions are still fully automated.

- ◆ Test case = single business document
- ◆ Some test cases require preceding document to be executed
- ◆ **int4 IFTT** allows to group test cases and replicate the whole business flows
- ◆ It is possible to mix inbound and outbound interfaces and manual operations together
- ◆ Variable concept (part of business objects) allows test cases to pass their own document numbers between each other

◆ Sample process:

1. Inbound sales order (*interface step*)
2. Outbound sales order confirmation (*interface step*)
3. execution of **eCATT** recording to create a delivery in reference to sales order
4. Outbound **ASN** (*interface step*)
5. execution **eCATT** recording to invoice the **SO** from step 1
6. Outbound invoice (*interface step*)

- ◆ Prerequisite: The **ICO** (integrated configuration object) in the **PI/PO** needs to be set to store the message content after mapping, the one that will be delivered to the receiver
- ◆ Test case consists of two steps:
 - ◇ 1) Action to trigger the message by **eCATT** posting of business document or by posting other inbound interface (ex. inbound sales order triggers outbound sales order ack.)
 - ◇ 2) Outbound message to store the original **XML** and associated business object name for future assertions.

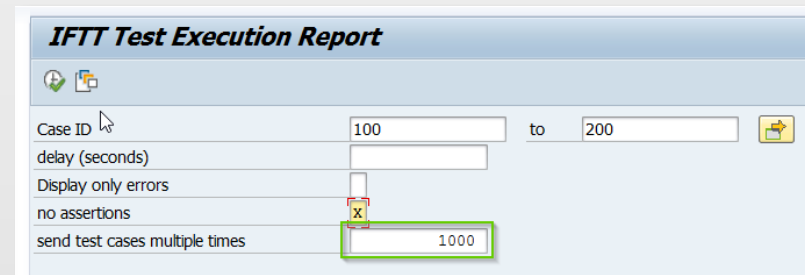
Case ID	Description	Interface Type	DocumentNumber	ObjectDef.ID	pre caseID
0000000771	inbound sales order#1	IDoc msg number (PI)	0000000003247823	SALES_ORD	
0000000772	outbound sales order ack. #1	PI Outbound	000c29a75f071ee69fccf3fbcc6c9e9a	OUTB_SO	0000000771

- ◆ During the execution, **int4 IFTT** automatically searches SAP PI/PO database of processed messages and finds the proper XML message by interface name and namespace stored in the business obj. definition and business document references given by the previous step.
This message would be used for comparison with the original message stored during the test creation.
- ◆ The assertion allows for exceptions, which are stored as XPath expressions in the obj. def. assigned to the test case. Exceptions include: *creation date & time, business document number, etc.*
- ◆ Scope: ECC output determination, standard and custom ABAP routines to generate the message, SAP PI routing and mapping steps, RFC lookups, BPM
- ◆ Out of scope: operations in outbound communication channel

- ◆ Once created, interface test case may not be valid after implementing Change Requests to its functionality
- ◆ It can be updated with the following procedure:
 - ◇ Run the test case, the results will fail due to the current definition
 - ◇ Validate & confirm new business document created by the execution
 - ◇ Update the test case, using the current interface message as its basis

int4 IFTT can be used as an interface load generator tool:

- ◆ It supports mode where the messages from select test cases are sent to the middleware platform (SAP PI/PO) multiple times:



IFTT Test Execution Report	
Case ID	100 to 200
delay (seconds)	
Display only errors	no assertions
send test cases multiple times	1000

- ◆ It would use the standard method of configuration variables to generate unique document numbers and randomize the content (in a limited way)
- ◆ As a standard, **int4 IFTT** only tests the SAP part of the integration flow, but with a simple workaround, the **int4 IFTT** can deliver messages outside SAP, for example to MQ (by creating a separate interface in SAP PI and attaching it to the **int4 IFTT** configuration). This approach allows performing a real end-to-end performance testing.

◆ In a typical project landscape:



IFTT can be installed and used in:

- ◆ **DEV:** used by **Developers / Integration Architects / Functional Experts** mostly **during the development phase and unit testing** but also for any interface-related **defect fix /CR implementation and retesting**.
- ◆ **QA:** used by **Integration Architects and Functional Experts** during the **Integration Testing and UAT**. During the UAT all scenarios and test cases should be finalized and the regression testing should start on regular basis. It should be mandatory for all defect fixes and CRs to confirm if interfaces are working in the way that was accepted and signed off by UAT
- ◆ **TST:** This is a system closest to production environment, it should contain the most comprehensive database of interface test cases that cover all business process operated globally by all sites (not only the scope of the current project). In TST the **int4 IFTT** is mostly operated by **Run teams**. **Mandatory test runs** should precede **any deployment** to production of interface-related changes and in particular any **new site go-live**.
- ◆ **int4 IFTT** is not intended and should never be used in a production environment (even when deployment is disabled by checking the mandatory attributes)

Case study: Existing interfaces, due to incorrect implementation, cause low data quality and high maintenance cost. No business involvement expected – business logic remains intact.

1

Automatically compare business documents posted using both – the new and the old interfaces on two separate test systems in order to catch any possible inconsistencies.

2

Save time and resources with testing
2 000 business documents in 1 minute!

3

Get your recently tested interface running on the production system **without engaging the business!**



*(i.e. inbound sales order triggers outbound sales order acknowledgement.)

- ◆ **Testing both** the integration platform and the application systems **simultaneously**
- ◆ **Quick and simple** test case generation (just choose any of the existing business documents; no need to create one from scratch like in Tosca Test Suite, for example)
- ◆ **Flexible configuration** with pre-delivered content
- ◆ Testing **does not involve time & resources** of the source system



- ◆ **define what needs to be checked only once** – results based on database entries and associated business objects
- ◆ **Support for Agile Methodology**, quick tests after each development cycle
- ◆ Applicable for **both for development and support team**
- ◆ **Assurance of business system stability**
- ◆ Nearly **End-to-End testing** → covers the whole SAP Landscape





<http://int4.com>
Office@int4.com

Michal.Kowalczewski@int4.com
Expert Consultant
+48 604 996 052

