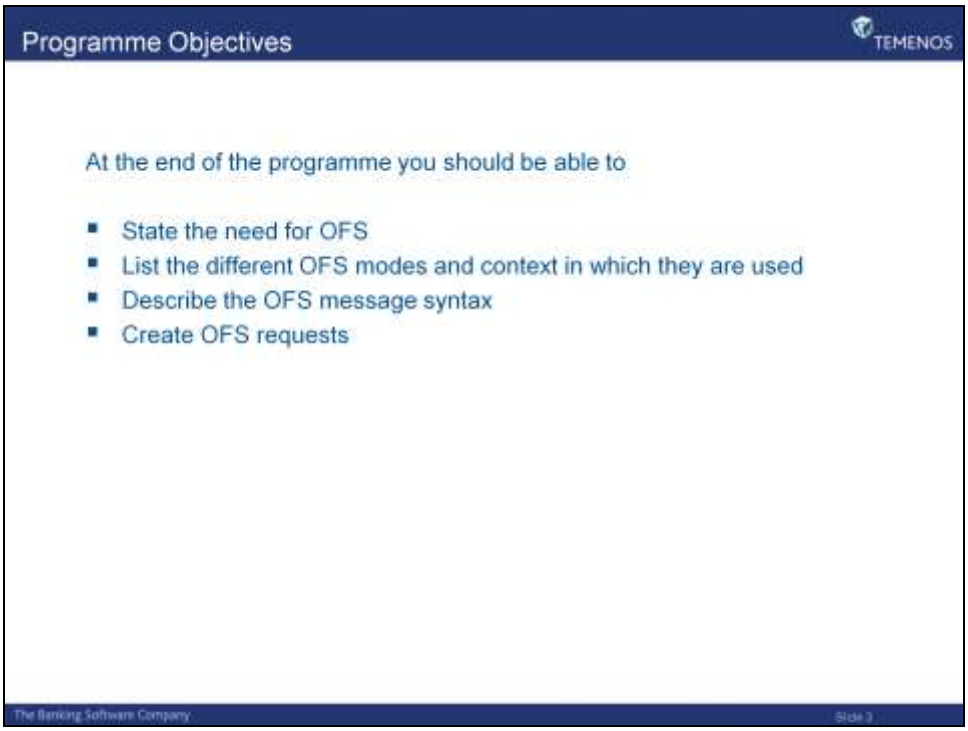
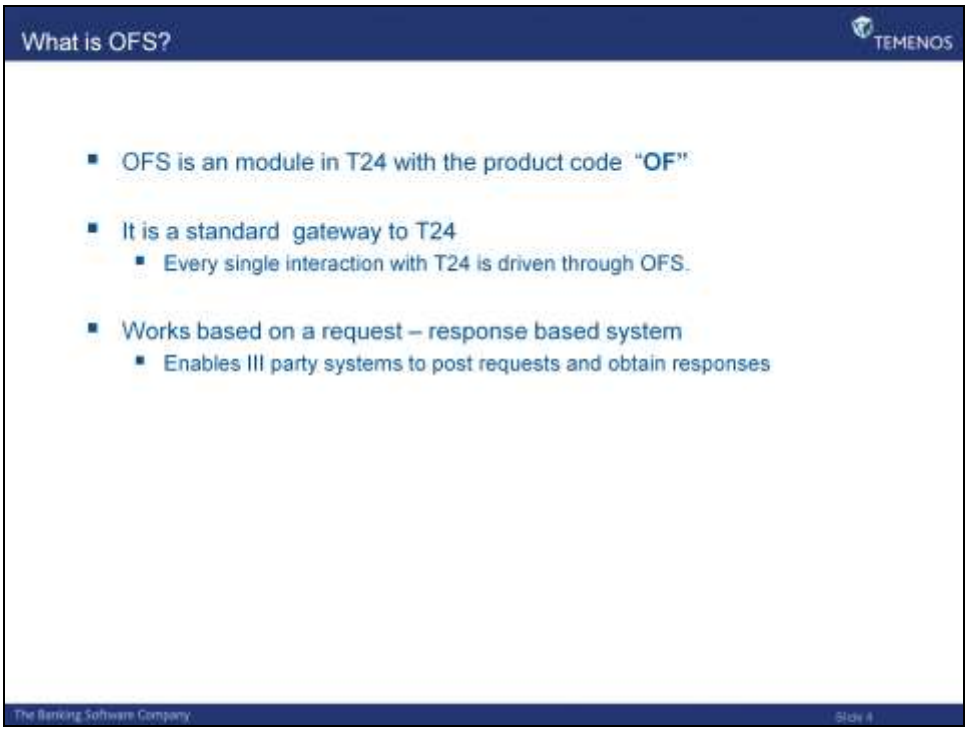
Welcome to this session on Open Financial Service.



At the end of the programme you should be able to

1. State the need for OFS
2. List the different OFS modes and context in which they are used
3. Describe the OFS message syntax
4. Create OFS requests

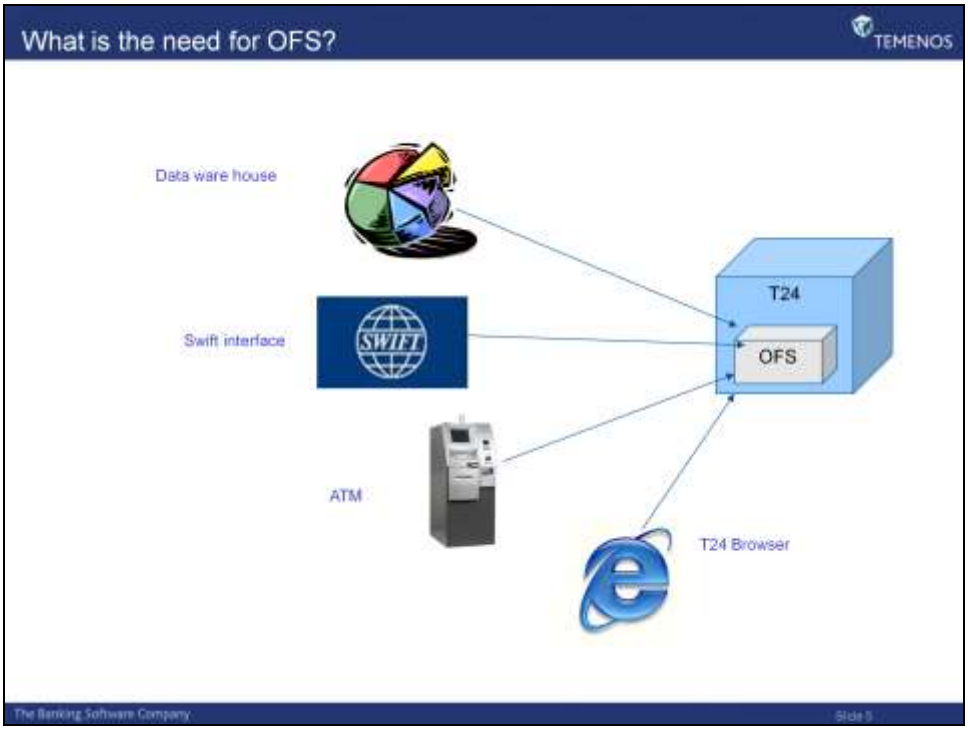


What is OFS ? Is it something outside of T24? What does it comprise of?

Well, OFS is a standard module within T24, with a module code OF. It is the ONLY standard gateway to TEMENOS T24. Now, you may be thinking what does that mean? Simply put, it means that every single interaction with T24 is driven through OFS.

The second piece of OFS are messages. OFS is message driven , i.e. it works on a request-response based system. The OFS syntax, or message structure is proprietary to TEMENOS T24. It is the native way to represent requests to execute T24 transactions, enquiries or routines.

The OFS module provides the infrastructure necessary to process the OFS messages. It has many components which we will see as we go through this course.



What is the need for OFS? TEMENOS T24 technology platform products like TEMENOS Connector, TEMENOS T24 Browser and TEMENOS Internet Bank use OFS to interact with TEMENOS T24 Applications. This implies that when we input a transaction , say a Funds Transfer in Browser, T24 actually receives an OFS message for Funds Transfer. This also implies that the OFS processing is not merely an update to the database, but goes through the same level of validations, including SMS, as in a manual effort.

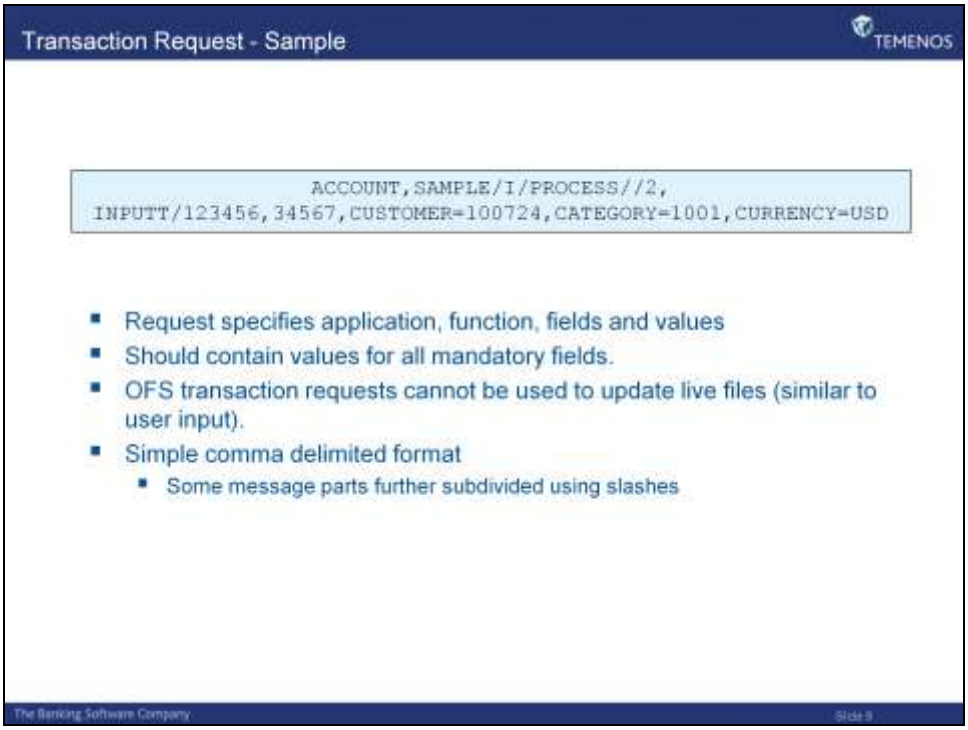
Secondly , with OFS, it becomes possible for T24 to transact with and reply to queries from various external channels (or systems). This is especially handy when we have huge volumes of transaction to be updated, or when there are systems like ATM switches and other banking systems requiring a 24/7 connection to T24.

We have seen that OFS messages are requests to T24 . There are 2 types of common requests in OFS. Each request follows a particular syntax which you will learn as you go through this session. One is a transaction request that will create, modify or delete a record in any application in T24. The other is an enquiry request, which queries data from T24.

OFS messages follow two formats

a. Native format – this is a simple string format, following a comma delimited pattern
b. XML format – OFS also understands an XML format which is described as the Browser XML format since it is used only by Temenos Browser.

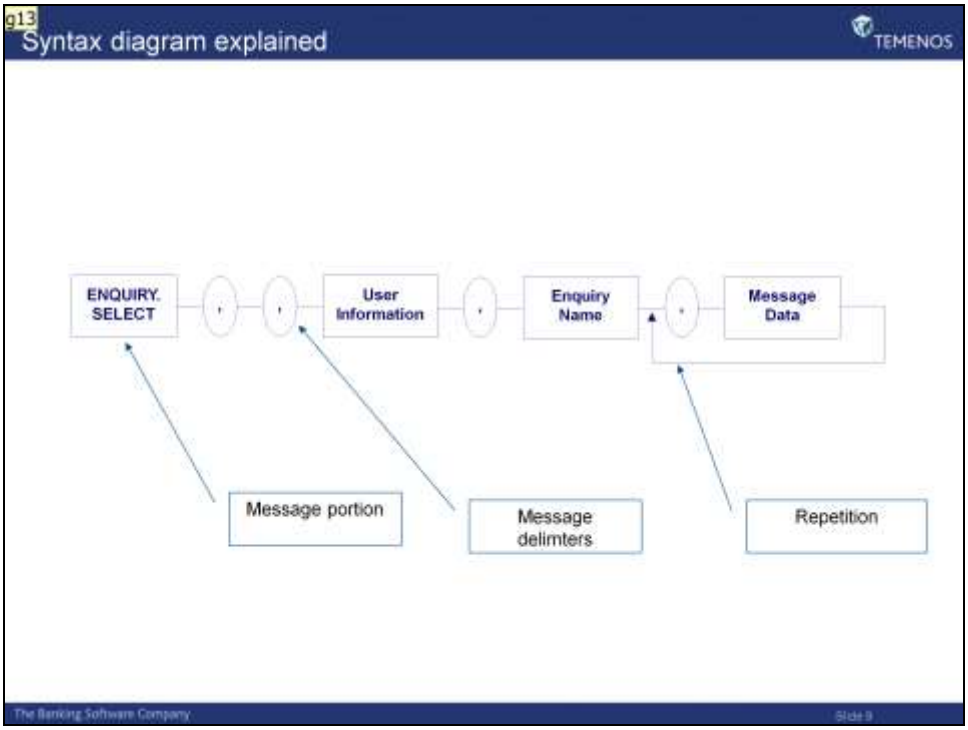
You will study the native OFS format alone in this session.



Look at the sample OFS request shown. This is a request to update an account (which is not a live application). It contains all the mandatory fields for the account such as customer number, category code and currency. It also contains other details essential for a transaction such as the T24 user id, password and the id of the account to be updated.

You can also see that the OFS messages follow a simple comma delimited format. Some parts of the message are divided further using a forward slash character.

We will look at the syntax in detail as we move on..



We have used syntax diagrams in this course to describe OFS messages. You will now learn the purpose behind the symbols used within a syntax diagram.

The sample shown is a syntax diagram for an ENQUIRY.

1. The rectangular boxes indicate message portions. Eg ENQUIRY.SELECT is one part, user information is another.

2. The ovals indicate message delimiters i.e. the comma is used as delimiter in this example

3. The arrow indicates repetition or multiple instances i.e. In the above OFS message, message data may occur many times

This diagram describes the syntax of a Transaction request.

Operation

We need to tell T24 the name of the application. Operation part of the message contains the name of the Application. Eg: ACCOUNT.

Options

Options is optional since many of the parameters here may be defaulted. This contains many sub-parts with each sub-part separated from the other using a forward slash. The sub-parts are Version name/Function/Process type/ GTS.Control value/No.of.authorisers. E.g. TRG/I/VALIDATE//2

Version name – this must a valid version name without the usual preceding application name and comma. For eg to use a version name ACCOUNT,CLIENT you will only specify CLIENT. Therefore comma versions are not valid.

Function – a T24 function such as I, R, A,H,V. Note that C and P is not supported.

Process Type – this may be VALIDATE or PROCESS and controls whether the transaction is to be validated or processed.

Gts control will be discussed later. No of authorisers may be used to specify the no of authorisers . I.e. zero, one or two. GTS control and no of authorisers if supplied will override the corresponding settings in the Version.

User Information

This consists of the Sign On Name/Password/Company code

E.g INPUTT/123456/GB0010001

Transaction ID

The Transaction Id part of the message contains the transaction id of the record used in the transaction. This is optional for new transactions if the underlying application is designed to allow this. In a case where it is allowed, the ID may be automatically generated by the application.

The transaction ID is mandatory for See , Authorize , Delete functions. The transaction ID can also contain an optional Message ID for the message.

For eg 20548/20081010001 .

In this case the 20548 is the record id and 20081010001 is the message id. This message id could be used by external systems to uniquely identify each OFS message.

Message Data

The message data portion of the message structure contains the data required to create or update the transaction. Message portion follows the format

Fieldname=data

Eg CUSTOMER=100297

When you need to assign values to multi values or sub values , you may follow the format

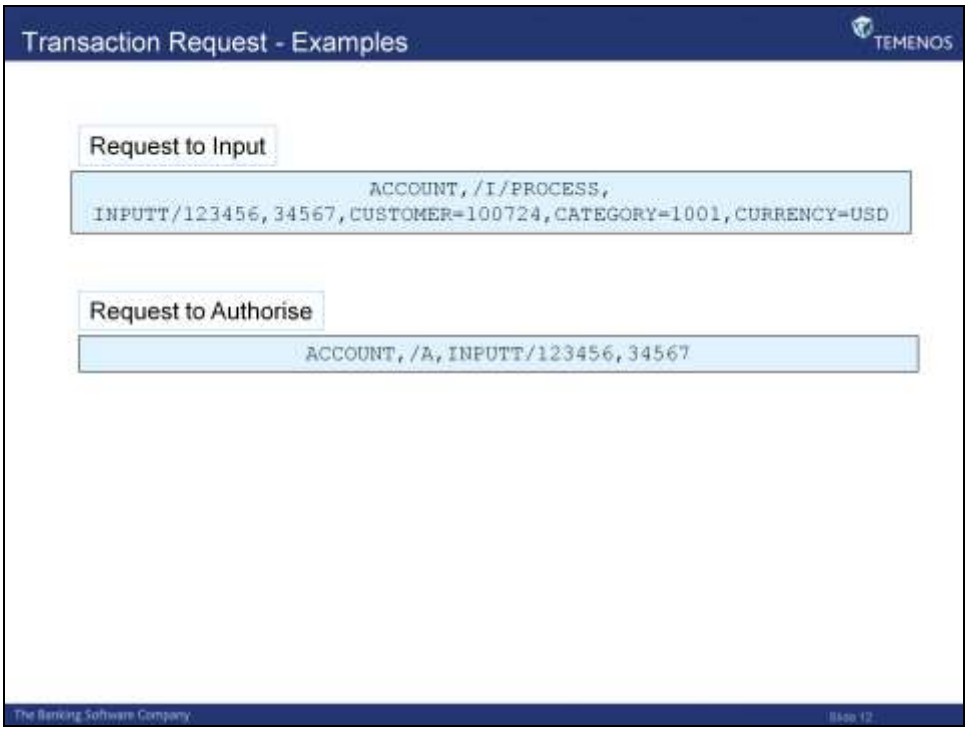
Fieldname :multi value number : sub value number = data

Eg CUSTOMER:1:1=100297

This implies the first sub value belonging to the first multi value of the field CUSTOMER is assigned a value of 100297. The first multi value or sub value is taken as the default in the absence of positions If 'NULL' is specified as field data, OFS will blank the field of all data.

The message data portion of the message can be repeated for each field separated by a

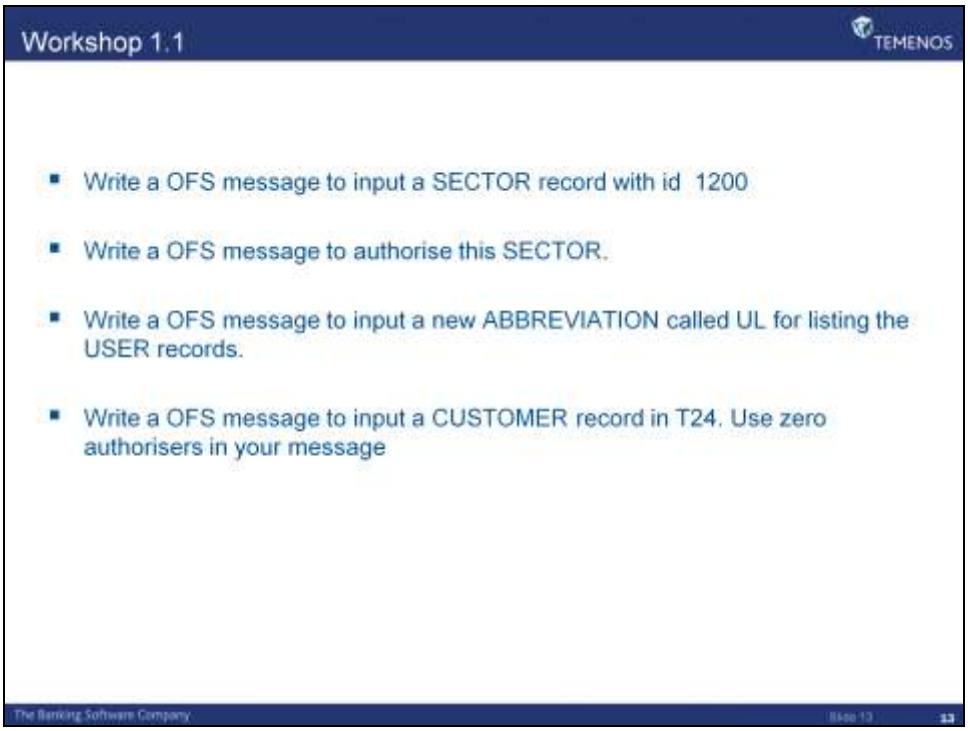
comma (,).



Look at the sample requests shown.

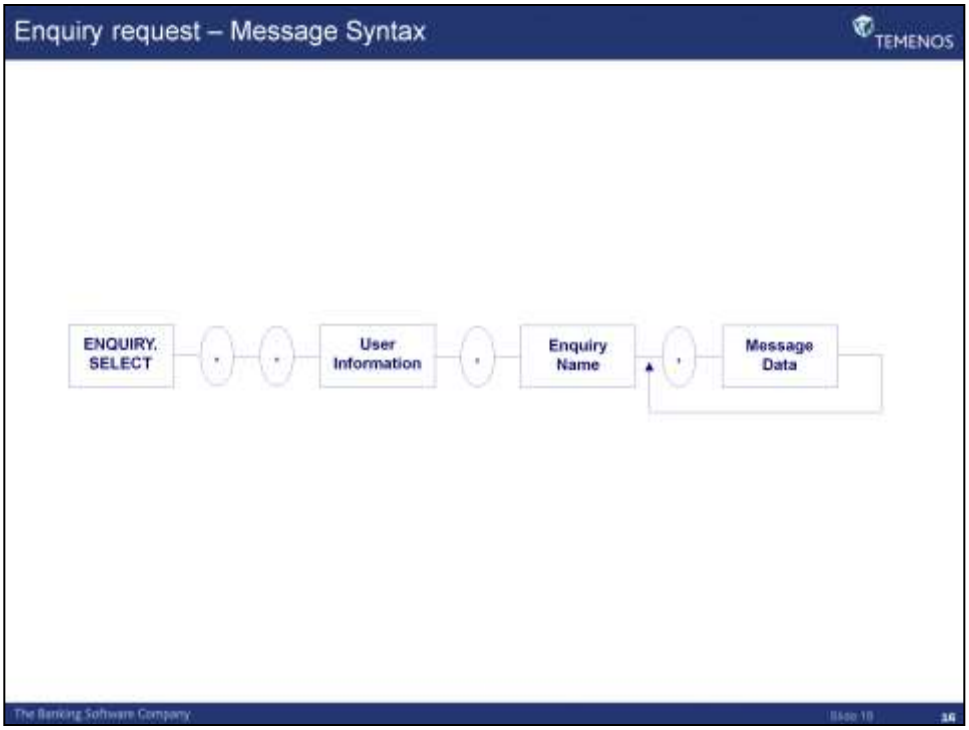
The first one is a sample request to input an account record. This has not used a version. The mandatory field values have been specified.

The second request is one to authorise the record previously input. This does not contain values in the data portion. PROCESS has not been specified either since that is the default process type.



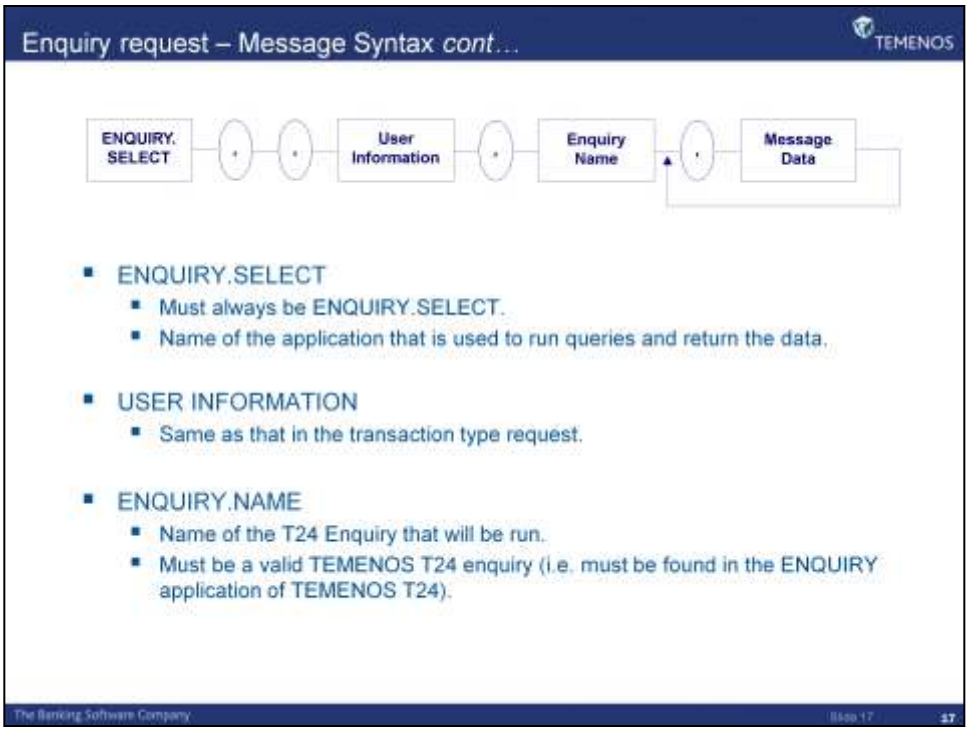
Hint : Open the applications from browser or classic. Identify the mandatory fields. Then write the messages.

You will now study the message format for an enquiry



The diagram shows the syntax of OFS Enquiry type request messages. All the main portions of the message for Enquiry type requests are shown in it.

The structure is very much similar to a Transaction Request. You may have noticed that the options part has been omitted, since it is not relevant to an enquiry. However commas are given to indicate the placeholder, and retain the general structure of an OFS message.



The portions of an Enquiry Request are

ENQUIRY.SELECT

The first portion of an enquiry type request *must* always be ENQUIRY.SELECT. This is the name of the routine that is used to run queries and return the data.

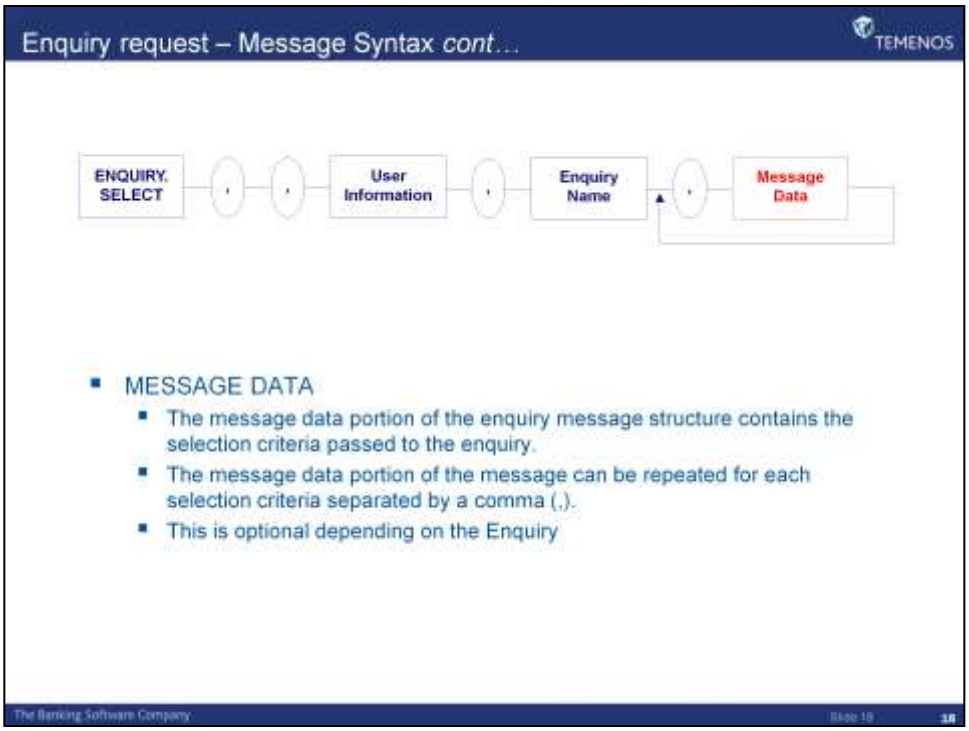
User information

The user information portion of the message structure is the same as that of the transaction type request.

Enquiry Name

You will specify the name of the T24 Enquiry to run in this part.

The Enquiry name supplied here must be a valid T24 enquiry (i.e. it must be found in the ENQUIRY application of T24).



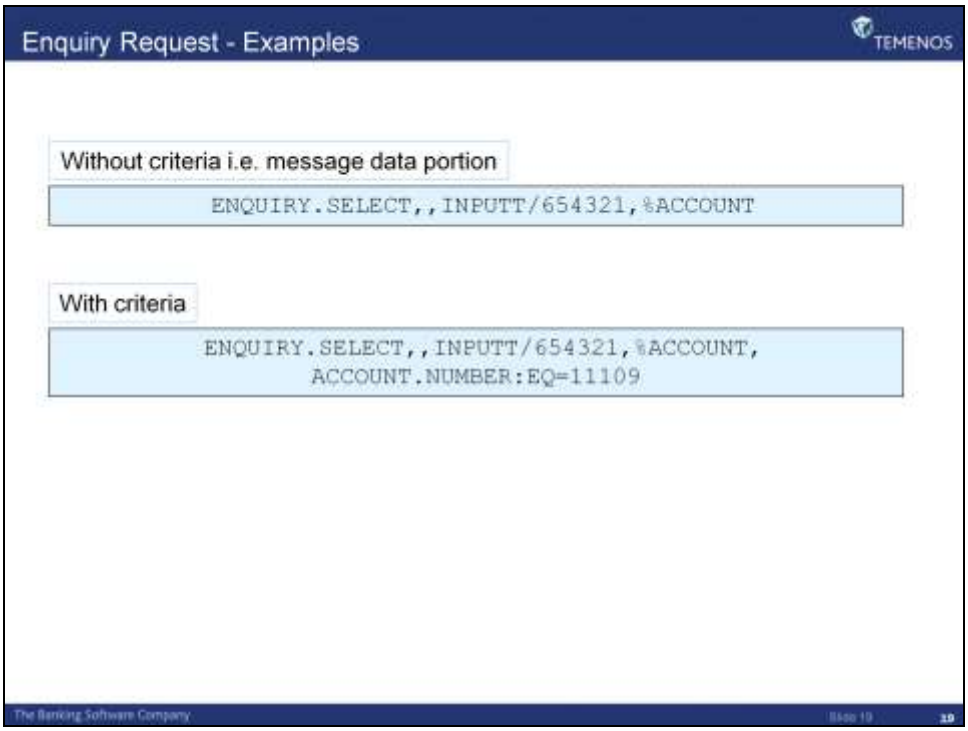
What does the message data portion of the enquiry request contain?

- The message data portion of the enquiry message structure contains the selection criteria passed to the enquiry.
- The message data portion of the message can be repeated for each selection criteria separated by a comma (,).
- This is optional depending on the Enquiry

The three different parts of a Selection Criteria part of the MESSAGE DATA are:

- Selection Field – denotes the name of the field of that must be either a field in the STANDARD.SELECTION for the file on which the enquiry is based (ie- the application that is specified in FILE.NAME field of the ENQUIRY application)or a value set in SELECTION.FLDS field of the ENQUIRY application.
- Operand – denotes the operand that must be used for selection for the value specified in the SELECTION.FLDS field of the ENQUIRY application. The operands can be EQ, NE, GE, GT, LE, LT, UL, LK and NR. The operand must be separated from the selection fields using a colon (:).
- Field Value – denotes the data value for the values specified in the SELECTION.FLDS (field of the ENQUIRY application) and the operand for selection. This must be separated from the operand using a equal sign (=).

- E.g. ACCOUNT.NUMBER:EQ=11107



This is an enquiry request to call the default account enquiry

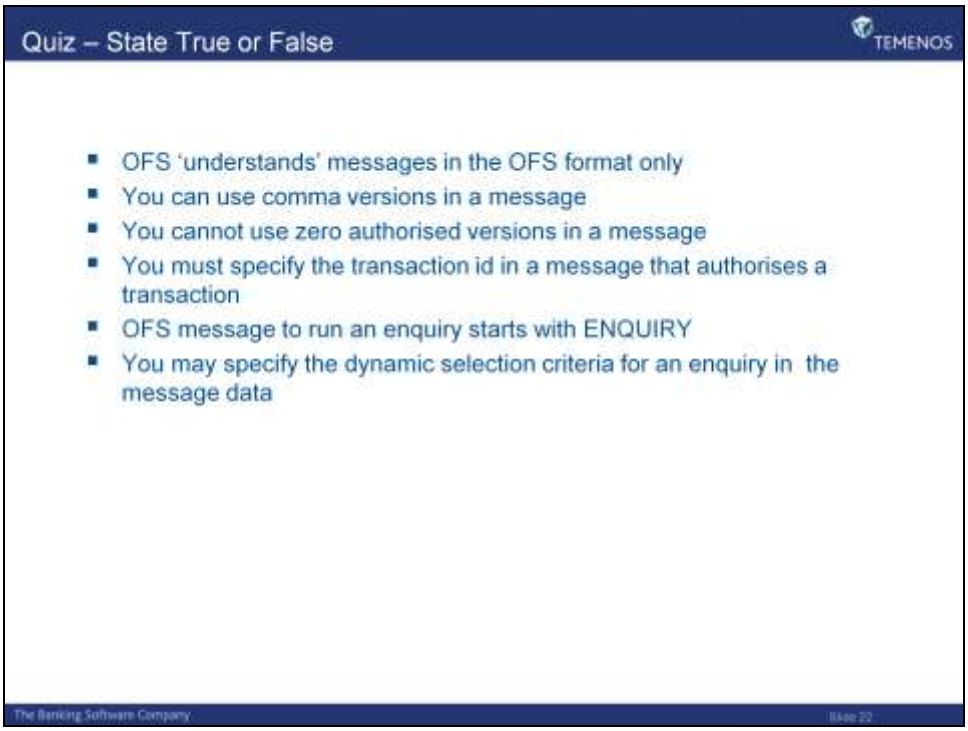
ENQUIRY.SELECT,,INPUTT/654321,%ACCOUNT

This is the same enquiry request with a criteria

ENQUIRY.SELECT,,INPUTT/654321,%ACCOUNT, ACCOUNT.NUMBER:EQ=11109

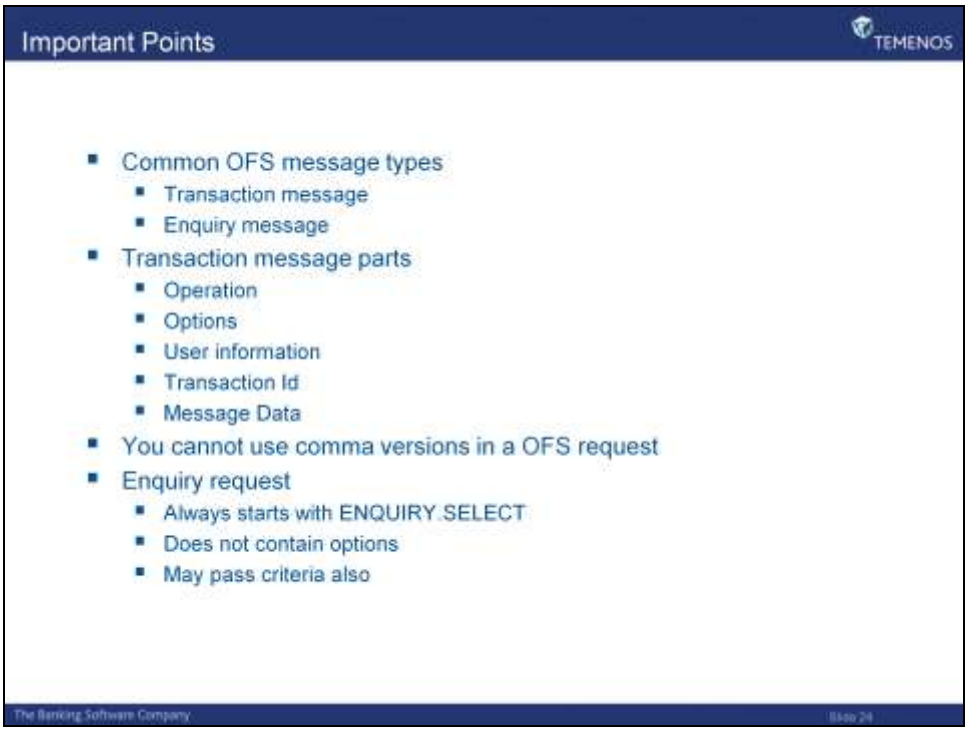
Workshop 1.2 – Enquiry Type Messages

- Write an Enquiry type request to find the list of INDUSTRY records found in yourT24 area
- Write an Enquiry type request to find the current day's balance summary of an ACCOUNT (say for example 29987). (TIP: Use the enquiry ACCT.BAL.TODAY)



State whether the following statements are true or false

- OFS 'understands' messages in the OFS format only
- You can use comma versions in a message
- You cannot use zero authorised versions in a message
- You must specify the transaction id in a message that authorises a transaction
- OFS message to run an enquiry starts with ENQUIRY
- You may specify the dynamic selection criteria for an enquiry in the message data



Common OFS message types

- Transaction message
- Enquiry message

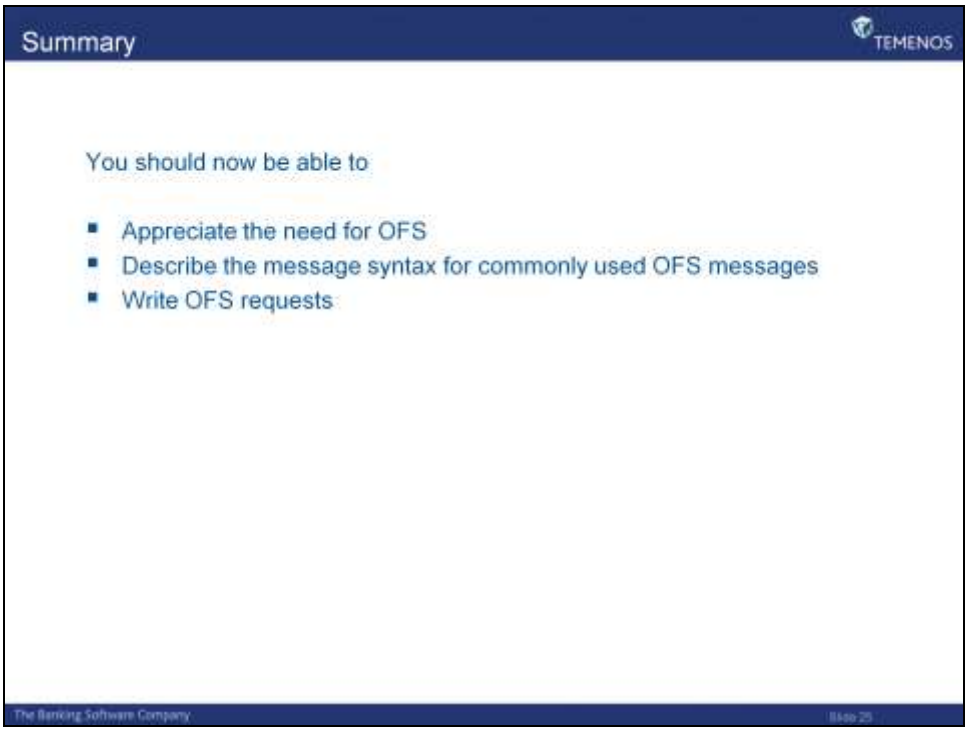
Transaction message parts

- Operation
- Options
- User information
- Transaction Id
- Message Data

You cannot use comma versions in a OFS request

Enquiry request

- Always starts with ENQUIRY.SELECT
- Does not contain options
- May pass criteria also



You should now be able to

1. Appreciate the need for OFS
2. Describe the message syntax for commonly used OFS messages
3. Write OFS requests