# Interchange Flows

This section describes the Interchange flow and its services in more detail. IPP can receive and process payment messages from different input channels (upstream systems) and from Clearing & Settlement systems (RTGS/NEFT).

If a service has an elaborated description of its functionality, a reference to the specific document will be mentioned.



Figure 3 - Incoming Interchange Flow

## Pre-Interchange Flow Service

Some services are used prior to the start of the interchange flow and will be totally independent of which flow will be initiated for the interchange. These services are intended to store an incoming interchange into the system, parse its contents according to the file type and then trigger the appropriate flow.



Figure 4 - Interchange Pre-Processing

* **Interchange Loader-**  The Interchange Loader reads an incoming payment interchange from the Interchange Queue (*6.23 Interchange Loader*)
* **Interchange Parser-** The Parser is responsible for reading the Transport Data stored in the received interchange (*6.24 Interchange Parser*)
* **Validate Interchange –** service will perform interchange specific validations (6.22 Validate Interchange)

## Incoming Interchange Flow

This section describes the Incoming Interchange flow. The Interchange flow is also independent of the inbound and outbound definitions and therefore, all interchanges go through the same processing per scheme.

This Incoming Interchange Flow consists of following steps:

1. Validate Interchange **service validates certain attributes of the Interchange according to ISO 20022 standards (*6.22 Validate Interchange*)**
2. **Accept Interchange** At the end of Interchange flow the system updates the status of an Interchange to “Accepted” to mark that the Interchange has been positively processed

# Instruction Flow

Once the interchange is parsed and validated successfully, the appropriate Instruction Flow will be triggered. In XYZ implementation following flows for Credit Transfers are configured, which will share some services and have some separate services depending on the flow.

The Instruction flow that is started depends, on whether the Instruction needs to be processed in an Inbound (message received from CSM) or Outbound flow (message received from channels)/ whether it is a Credit Transfer or an R-Instruction/which payment scheme does the message belong to.





Figure 5 - Outbound Credit Transfer Instruction Flow

This flow processes the payment request received from EMX. The system will process the payments received from EMX and then forward it to clearing or consume it within BPH.

This flow consists of following steps:

1. **Product Determination service** is used to determine the “product Name’ for specific incoming payment messages received from channels (*6.1 Product Determination*)
2. **Timeline Validation** service checks that the payment has been received within the allowed timelines (*6.8 Timeline Validation*)
3. **Validate Instruction service** is used to perform instruction specific validations on the payment (*6.2 Validate Instruction*)
4. **Duplicate Reference Id Check** service performs duplicate check on the instruction and transactions within a payment (*6.4 Duplicate Reference Id Check*). System does not perform duplicate reference check for Manual Entry initiated payments.
5. **Disposition Check service** validates the account format as well as check whether the account exists or not (*6.3 Disposition Check*)
6. **Determine Processing context** service determines the various values maintained at the agreement (*0*
7. **Determine Incoming Settlement day** service would set the settlement day based on the instructed day provided in the payment instruction (6.7 Determine Incoming Settlement day)
8. **Duplicate Suspicion Check** service ensures that instruction is not a business duplicate of an existing instruction received from the channel earlier (*6.6 Duplicate Suspicion Check*). System does not perform duplicate suspicion check for Manual Entry initiated payments.
9. **Calculate Instruction Release Date** service checks the settlement date of the instruction and updates the expected de-warehousing date and time for instruction ( 6.9.1 Calculate instruction Release Date)
10. The system will check whether the instruction release date is in future, if it is in future it will go to step 11 else to step 13**.**
11. **Warehousing service** will check the expected de-warehousing date and time, if required it will warehouse the instruction (*6.9.1.1 Warehouse an Instruction*)
12. **De-Warehousing service** would de-warehouse the instruction one the Expected de-warehousing date and time are past (*6.9.2 Release a Warehoused Instruction*)
13. The system checks if payment is NACH payment if yes, the goes to setp14 else goes to 13.1
    1. The **system** checks whether Batch booking is required or not. If batch booking is required and payment is not NACH payment then system goes to step ‘14’ else to step ‘18’.
14. **Check for Subsystem Warehousing** service is called from within the EARMARKING service. It has been shown in the flow diagram separately for purpose of clarity event though it won’t be called from flow but from within the earmarking service.

Itdetermines whether the payment is received on a business day and is being processed under subsystem’s allowed submission start and end time. This check is done for a specific subsystem (like ATLAS2, SSW, etc.), for which the service has been called.

* 1. The **system** would check whether the cut-off for the subsystem is met or not. If passed, the system would proceed with the flow; else, the instruction would be marked as eligible for warehousing (*6.25.1 Settlement day re-calculation is required* )
  2. **Prepare for Subsystem Warehousing** service would re-calculate and set the settlement day and Instruction release day (*6.25.2 Prepare for Sub-System Warehousing*)

1. **Earmarking service** is used to create an earmark on the debit account for the amount of the instruction. Besides this it will also call step ‘14’ and if the cut-off is passed the ‘Warehouse an instruction’ service would be called and the instruction would be warehoused (*6.12 Earmarking*)
2. **Accounting service** creates debit accounting event for the amount of the instruction (*6.14 Accounting*)
3. **Debit Credit Notification** servicecreates a debit/credit notification whenever there is a transaction happening to the account (*6.15 Credit/Debit Reporting Service*)
4. **Accept Instruction** service marks the status of the incoming instruction as accepted (*6.26 Accept Instruction*)
5. **Route Transaction** service is used to route the transaction to its destination (*6.10 Routing*)
6. **Start Appropriate Transaction Flows** service invokes the relevant transaction flows for the transactions within an instruction.
7. **Validate Transaction** service in instruction flow is used to perform certain validations on transactions required within instruction flow (*6.17 Validate Transaction*)
8. **Re-value Date an Instruction** will update the settlement date of the instruction if it is in past (*6.36 Re-Value Date an Instruction*)

# Transaction Flow

Once the instruction is validated successfully, the appropriate Transaction Flow will be triggered. In IPP RI following transaction flows are configured, which will share some services and also have some separate services depending on the flow.

#### Outbound CT Not On-Us Transaction Flow

The Outbound CT Not On-Us Transaction flow is used for processing outbound not on-us Credit Transfer transactions (pain.001/pacs.009) which are destined for RTGS/NEFT clearing.





Figure 11 - Outbound CT Not On-Us Transaction Flow



This flow consists of following steps:

1. **Validate Transaction** is used to perform transaction specific validations on the payment (*6.17 Validate Transaction*)
2. **Duplicate Reference Id** **Check** service performs duplicate check on the instruction and transactions within a payment (*6.4 Duplicate Reference Id Check*). System does not perform duplicate reference check for Manual Entry initiated payments.
3. **Determine Processing Context** service determines the various values maintained at the agreement (6.5 Determine Processing Context)
4. **Duplicate Suspicion Check** service ensures that instruction is not a business duplicate of an existing instruction received from the channel earlier (*6.6 Duplicate Suspicion Check*). System does not perform duplicate suspicion check for Manual Entry initiated payments.
5. The **system** checks whether the payment is FICT or not. If it is FICT it goes to step ‘6’ else to step ‘9’.
6. The **system** checks whether the incoming channel is OMR. If it is OMR, system goes to step ‘7’ else to step ‘9 6.1’.
   1. **Disposition Check** validates the account format as well as check whether the account exists or not (6.3 Disposition Check). System goes to step 9.
7. **Enrich NSTP** service marks the payment as NSTP (*6.20 Enrich NSTP*)
8. **Threshold Check** service checks if the payment transaction breaches the upper limit set for RTGS ( *6.13 Threshold Check*)
9. **Check for Subsystem Warehousing** service is called from within the SSW check service. It has been shown in the flow diagram separately for purpose of clarity event though it won’t be called from flow but from within the SSW service.

Itdetermines whether the payment is received on a business day and is being processed under subsystem’s allowed submission start and end time. This check is done for a specific subsystem (like ATLAS2, SSW, etc.), for which the service has been called.

* 1. The **system** would check whether the cut-off for the SSW subsystem is respected or not. If passed, the system would proceed with the flow; else, the transaction would be marked as eligible for warehousing.
  2. **Prepare for Subsystem Warehousing** service would re-calculate and set the settlement day and transaction release day.

1. **De-warehouse Transaction** service would de-warehouse a transaction for which the EDWART is in past (6.27.2 Release a Warehoused Transaction)
2. **The system** checks if the warehousing type is ‘Same day’, if it is same day the **system** would go to step 14/18 else to step 1 in the flow
3. **Disposition Check** validates the account format as well as check whether the account exists or not (*6.3 Disposition Check*)
4. **SSW/Filtering Check** service will send the payment for filtering check and read the response (6.11 SSW/Filtering Check)
5. In case of **batch booking**, the system goes to step ‘8’ to perform Threshold check followed by Step 17, else to step ‘15’.
6. **Check for Subsystem Warehousing** service is called from within the EARMARKING service. It has been shown in the flow diagram separately for purpose of clarity event though it won’t be called from flow but from within the earmarking service.

Itdetermines whether the payment is received on a business day and is being processed under subsystem’s allowed submission start and end time. This check is done for a specific subsystem (like ATLAS2, SSW, etc.), for which the service has been called.

* 1. The **system** would check whether the cut-off for the subsystem is met or not. If passed, the system would proceed with the flow; else, the instruction would be marked as eligible for warehousing (*6.25.1 Settlement day re-calculation is required* )
  2. **Prepare for Subsystem Warehousing** service would re-calculate and set the settlement day and Instruction release day (*6.25.2 Prepare for Sub-System Warehousing*)

1. **Earmarking Service** creates an earmark on the debit account for the amount of the transaction(*6.12 Earmarking*)*). S*ystem goes to step ‘8’ to perform Threshold check followed by Step 17,
2. **Accounting Service** is used for creation of relevant accounting events (*6.14 Accounting*)
3. **Credit/Debit Reporting Service** creates a debit/credit notification whenever there is a transaction happening to the account (*6.15 Credit/Debit Reporting Service*)
4. The **system** marks the payment as ready for submission.
5. **Calculate Outgoing Transaction Dates** service is used to update the settlement date of the transaction in case it is in past. This service is also called after approval/repair/de-warehousing to update the settlement date ( **6.10.2 Calculate Outgoing Transaction Dates**)
   1. **Warehousing Service** will check the expected de-warehousing date and time, if required it will warehouse the Transaction**De-Warehousing Service** would de-warehouse the transaction once the Expected de-warehousing date and time are past
6. **Ready** **for Submission**, the system marks the payment as ready for submission

#### Outbound CT On-Us Transaction Flow





Figure 15 - Outbound CT On-us Transaction flow



This flow processes outbound On-us transactions i.e. book Transfers.

1. **Validate Transaction** is used to perform transaction specific validations on the payment (*6.17 Validate Transaction*)
2. **Duplicate Reference Id** **Check** service performs duplicate check on the instruction and transactions within a payment (*6.4 Duplicate Reference Id Check*). This service is skipped for payments initiated from via Manual Entry.
3. **Determine Processing Context** service determines the various values maintained at the agreement (6.5 Determine Processing Context)
4. System checks whether the processing scheme is NOT equal to NACH. If not equal to NACH then system proceeds with step 5 else Step 6.
5. **VAN Check service** will validate if the received account number is a VAN number, and extract corresponding 19 digit AT2 account number along with other VAN related fields like (Virtual Client Id, Remitter Code, remitter Name etc.) (6.37 Virtual Account Number Check Service).
6. **Disposition Check** (credit account) validates the account format as well as check whether the account exists or not (*6.3 Disposition Check*)
7. **Duplicate Suspicion Check** service ensures that instruction is not a business duplicate of an existing instruction received from the channel earlier (*6.6 Duplicate Suspicion Check).* This service is skipped for payments initiated from via Manual Entry.
8. **Check for Subsystem Warehousing** service is called from within the SSW check service. It has been shown in the flow diagram separately for purpose of clarity event though it won’t be called from flow but from within the SSW service.

Itdetermines whether the payment is received on a business day and is being processed under subsystem’s allowed submission start and end time. This check is done for a specific subsystem (like ATLAS2, SSW, etc.), for which the service has been called.

8.1 The **system** would check whether the cut-off for the SSW subsystem is respected or not. If passed, the system would proceed with the flow; else, the transaction would be marked as eligible for warehousing (*6.25.1 Settlement day re-calculation is required* ).

**8.2 Prepare for Subsystem Warehousing** service would re-calculate and set the settlement day and transaction release day (*6.25.2 Prepare for Sub-System Warehousing*).

1. **De-warehouse Transaction** service would de-warehouse a transaction for which the EDWART is in past (*6.27.2 Release a Warehoused Transaction)*
2. **Calculate Outgoing Transaction Dates** service is used to update the settlement date of the transaction in case it is in past ( **6.10.2 Calculate Outgoing Transaction Dates**)
3. **The system** checks if the warehousing type is ‘Same day’, if it is same day the **system** would go to step 13 else to step 1 in the flow
4. **Disposition Check** (debit account) validates the account format as well as check whether the account exists or not (*6.3 Disposition Check*)
5. **SSW/Filtering Check** service will send the payment for filtering check and read the response (*6.11 SSW/Filtering Check)*
6. The system checks whether the processing scheme equals NACH. If yes then go to step 18 Check for Sub system warehouse else go to step 15. This is required because there is no debit earmarking required for NACH.
7. The system checks the batch booking flag. If batch booking indicator is ‘true’, system goes to step 18 else to step 16
8. **Check for Subsystem Warehousing** service is called from within the Earmarking service. It has been shown in the flow diagram separately for purpose of clarity event though it won’t be called from flow but from within the earmarking service.

Itdetermines whether the payment is received on a business day and is being processed under subsystem’s allowed submission start and end time. This check is done for a specific subsystem (like ATLAS2, SSW, etc.), for which the service has been called.

* 1. The **system** would check whether the cut-off for the subsystem is met or not. If passed, the system would proceed with the flow; else, the instruction would be marked as eligible for warehousing (*6.25.1 Settlement day re-calculation is required* ).
  2. **Prepare for Subsystem Warehousing** service would re-calculate and set the settlement day and Instruction release day (*6.25.2 Prepare for Sub-System Warehousing*).

1. **Earmarking Service** (Debit) creates a debit earmark on the account for the amount of the transaction (*6.12 Earmarking*)*.*
2. **Check for Subsystem Warehousing** service is called from within the Earmarking service. It has been shown in the flow diagram separately for purpose of clarity event though it won’t be called from flow but from within the earmarking service.

Itdetermines whether the payment is received on a business day and is being processed under subsystem’s allowed submission start and end time. This check is done for a specific subsystem (like ATLAS2, SSW, etc.), for which the service has been called.

* 1. The **system** would check whether the cut-off for the subsystem is met or not. If passed, the system would proceed with the flow; else, the instruction would be marked as eligible for warehousing (*6.25.1 Settlement day re-calculation is required* )
  2. **Prepare for Subsystem Warehousing** service would re-calculate and set the settlement day and Instruction release day (*6.25.2 Prepare for Sub-System Warehousing*).

1. **Earmarking Service** (Credit) creates a credit earmark on the account for the amount of the transaction (*6.12 Earmarking*)*.*
2. **Accounting Service** is used for creation of relevant accounting events (*6.14 Accounting*)
3. **Credit/Debit Reporting Service** creates a debit/credit notification whenever there is a transaction happening to the account (*6.15 Credit/Debit Reporting Service*)
4. **Create Communication** A positive PSR is created for Book transfer payments in the form of Pain.002 (6.16 Create PSR Event)
5. The **system** marks the payment as ready for submission and updates the status as “Settled”.