

Fujitsu Services End to End Release 1 - High Level Design Ref.: EA/HLD/002
Version: 1.00-3
Date: 19/01/2004

COMMERCIAL IN CONFIDENCE

Formatted

Document Title: End to End Release 1 - High Level Design

Document Type: High Level design

Release BI3 S60

Abstract: This document describes the High Level Design for End to End Release 1 Projects 1 and 3 excluding all LFS impacts. In summary, this includes the summarisation of cash and near-cash daily movements and the delivery of such data to the Horizon System boundary (for onward delivery to a new POL Financial System).

Document Status: ~~DRAFT~~APPROVED

Originator & Department: Pete Jobson (Tel: GRO) - Development Unit

Contributors:

Approval Authorities

| Name | Position | Signature | Date |
|----------------|--------------------------|-----------|------|
| Gareth Jenkins | E2E Design Authority | | |
| Dave Johns | APDU_S1 Design Authority | | |
| | | | |

Formatted

Chapter 0 - Document Control

0.1 DOCUMENT HISTORY

| Version | Date | Reason for Issue | Associated CP/ PinCL Nos. |
|---------|-------------|--------------------------------|------------------------------|
| 0.1 | 21/10/2003 | First Draft issued for comment | None |
| 0.2 | 4/11/2003 | Rework following comments | None |
| 0.3 | 422/11/2003 | Rework following comments | None |
| 1.0 | 19/01/2004 | Rework following comments. | None |
| | | | |
| | | | |

0.2 REVIEW DETAILS

| | |
|---------------------|---|
| Review Comments by: | 21st November 2003 to 29th January 2004 |
| Review Comments to: | Originator & Document Management |

Formatted

| Mandatory Review Authority | Name |
|---|--------------------|
| RASD | Gareth Jenkins(*) |
| Customer Services | Mik Peach(*) |
| Development Unit – Design Team | David Johns |
| | Rex Dixon(*) |
| | Sud Agrawal |
| | Roger Barnes |
| | Trevor Leahy |
| Development Unit – Development Team | Chris Bailey |
| | Matt Arris |
| | Martin McConnel(*) |
| Integration & Test | Walter Wright |
| | Janusz Holender |
| Post Office Ltd | |
| Optional Review/Issued for Information | |
| | Dave Wilcox(*) |
| | Pete Ambrose |
| | Tony Heaton |
| | Nick Lawman |
| | Roger York(*) |
| | James Stinchcombe |
| | Simon Fawkes |
| | Neil Gormley(*) |
| | Bob Gurney |
| | Phil Boardman |
| | Colin Mills(*) |
| | Asad Sheikh |

(*) Reviewers who returned comments

Formatted

COMMERCIAL IN CONFIDENCE

0.3 ASSOCIATED DOCUMENTS

| Reference | Doc | Vers- ion | Date | Title | Source |
|------------|---------------------|--------------|----------|--|---------------------|
| [CD001] | POL/E2E/DE S/001 | | | Automated Remittances Conceptual Design | POL |
| [CD002] | POL/E2E/DE S/002 | | | Overnight Cash Holding Conceptual Design | POL |
| [DPR02] | EA/DPR/002 | | | E2E Re-Architecting Release 1 Design Proposal | Fujitsu Services |
| [LFSHLD] | EA/HLD/001 | | | LFS Release 1 Delta HLD | Fujitsu Services |
| [AGHLD] | AD/DES/041 | 3.0 | 04/07/03 | TPS Agents for BI3 – HLD | Fujitsu Services |
| [EOD] | EP/DES/025 | 1.0 | 11/07/00 | EPOSS End of Day Service HLD | Fujitsu Services |
| [FSAIS] | EA/IFS/001 | | | POL FS AIS | Prism/ Xansa |
| [TPSHLD] | TI/DES/002 | | | TPS High Level Design | Fujitsu Services |
| [LFSAIS] | BP/DES/023 | | | LFS to SAPADS and SAPADS to LFS Application Interface Specification | Prism |
| [SFTLNCH] | NB/LLD/056 | | | SoftLaunch Low Level Design | Fujitsu Services |
| [TPSMAP] | AD/DES/047 | 4.1 | 02/06/03 | TPS Tables and Mapping | Fujitsu Services |
| [DIALOGUE] | LF/IFS/001 | | | E2E Release 1 – LFS Counter Dialogue Delta – Activity & Screen Flows | Fujitsu Services |

Unless a specific version is referred to above, reference should be made to the current approved versions of the documents. In particular later versions of some of these documents do exist; however, it is the versions indicated that have been used for the development of this interface.

0.4 ABBREVIATIONS & DEFINITIONS

0.4.1 Abbreviations

| Abbreviation | Definition |
|--------------|--|
| ACC | Authorised Collectors Card |
| ADC | Advanced Distribution Centre: Used as an abbreviation on the Horizon desktop for Remittances to and from SAP ADS |
| AIS | Application Interface Specification |
| CBDB | Counters Business DataBase. Post Office Limited's current Accounting Systems |
| CD | Conceptual Design |
| DIT | Direct Interface Test |
| DP | Design Proposal |
| DWh | Data Warehouse |
| E2E | End to End: Used in two contexts: <ul style="list-style-type: none"> The End to End Programme (of which this is the first release) End to End testing where testing is carried out by POL of the full business processes |
| EDS | The company which Post Office Ltd's cheque processing is outsourced to |
| EOD | End of Day |
| FAD | Financial Accounts Division (FAD Code) |
| FTMS | File Transfer Management Service |

| Abbreviation | Definition |
|--------------|--------------------------------------|
| HLD | High Level Design |
| LFS | Logistics Feeder Service |
| MIS | Management Information System |
| OLA | Operational Level Agreement |
| OMDB | Operational Management Database |
| ONCH | Overnight Cash Holding |
| POA | Post Office Account |
| POL | Post Office Ltd |
| POL FS | Post Office Ltd's Financial System |
| RDMC | Reference Data Management Centre |
| RDS | Reference Data System |
| SAP ADS | SAP Advanced Distribution System |
| SLA | Service Level Agreement |
| TIP | Transactional Information Processing |
| TMS | Transaction Management System |
| TPS | Transaction Processing System |

0.4.2 Definitions

The following terms, when capitalised as here, have specific meanings as indicated.

| Term | Definition |
|-----------------------------|--|
| Branch | Post Office location with one or more Counter PCs installed as part of the Horizon programme |
| Counter | Counter PC installed in a Post Office Branch |
| Counter Application | An application resident within the Counter that contains the business logic controlling the dialogue with the Clerk, or other business specific functions on the Counter (such as End of Day processing) |
| Counter Clerk | Person working in a Branch and operating a Counter |
| Horizon | Name that encompasses the totality of the systems provided by Fujitsu Services Post Office Account to support the automation requirements of Post Office Branches |
| Near-Cash | Transactions performed at branches are settled with a promise to pay. This promise comes in the form of Sterling cash, cheques, debit-card transactions and foreign currency. Sterling currency payment deemed to be 'cash'. Cheques, debit-card and foreign-currency can easily be converted to Sterling Cash and is called 'near-cash' |
| Operational Level Agreement | A non-contractual agreement between Fujitsu Services and Post Office Ltd on the nature and quality of specific elements of a service (e.g., Interface Agreement for Problem Management (CS/IFS/009)) |
| Receipt | A printed record of the Transaction at the Branch |
| Reconciliation | Ensuring the financial integrity of Transactions across service boundaries |
| Reference Data | Data that controls the system functionality and the business operations, that is present at System Start-up and does not change during normal operation (but may be changed off-line). On rare occasions the data may represent an initial start-up state that does change during operation but is not preserved between sessions. Configuration data and parameters for use by the rest of the system, within Post Office (including the Horizon Programme) |
| Settlement | Settling a Session where the balance of the session is reduced to zero and the appropriate cash (and other items such as cheques, debit cards, tokens, stamps etc) is exchanged between the Customer and the Clerk or between differing Products (for example; during stock re-valuation) |
| Transaction | A recorded and auditable instance of business activity, involving service provision or Stock movement across organisational or service boundaries |

Formatted

0.5 CHANGES IN THIS VERSION

0.5.1 Changes in Version 0.2

- Inclusion of DeskTopApps in Reference Data descriptions.
- Various minor amendments following comments
- Removal of POL FS Balancing transaction details from the Summary Trailer Record
- Reduction of generic ability to store multiple daily balance figures. Only a single balance figure will be held for Cash.
- Addition of new reference data items (including mapping for Cheques in ROAD Mode)
- All other changes can be located by using the Highlight Changes option in Word™

0.5.2 Changes in Version 0.3

- Removal of Transfer-In/Out transactions from Summarisation process and from Chart of Account Mapping
- Addition of Summarisation trailer record persistent object
- Clarification that the mechanisms of converting Type C to Type A reference data will be re-considered at later releases
- Various clarifications following comments

0.5.3 Changes in Version 1.0

- Minor clarifications and corrections following comment.
- Addition of various new reference data sections in Chapter 4 following the introduction of three new transaction modes.

Formatted: Bullets and Numbering

0.6 CHANGES EXPECTED

Updates following ~~initial review and following~~ baseline of Design Proposal

0.7 CONTENTS

CHAPTER 0 - DOCUMENT CONTROL.....2

0.1 DOCUMENT HISTORY.....2

0.2 REVIEW DETAILS.....2

0.3 ASSOCIATED DOCUMENTS.....3

0.4 ABBREVIATIONS & DEFINITIONS.....3

 0.4.1 Abbreviations.....3

 0.4.2 Definitions.....4

0.5 CHANGES IN THIS VERSION.....5

 0.5.1 Changes in Version 0.2.....5

 0.5.2 Changes in Version 0.3.....5

 0.5.3 Changes in Version 1.0.....5

0.6 CHANGES EXPECTED.....5

| | | |
|--|---|-----------|
| 0.7 | CONTENTS | 5 |
| 0.8 | TABLE OF FIGURES | 10 |
| 0.9 | TABLE OF TABLES..... | 10 |
| CHAPTER 1 - INTRODUCTION..... | | 11 |
| 1.1 | PURPOSE..... | 11 |
| 1.2 | SCOPE | 11 |
| 1.3 | READERSHIP | 12 |
| 1.4 | RELATED DOCUMENTS..... | 12 |
| CHAPTER 2 - ARCHITECTURE..... | | 13 |
| 2.1 | COMPONENTS | 13 |
| 2.2 | SAP ADS | 13 |
| 2.3 | LFS HOST..... | 14 |
| 2.4 | LFS LOADER | 14 |
| 2.5 | LFS HARVESTER | 14 |
| 2.6 | TPS HARVESTER | 14 |
| 2.7 | TPS HOST..... | 14 |
| 2.8 | POL FS..... | 15 |
| 2.9 | RDMC..... | 16 |
| 2.10 | CASH CENTRE..... | 16 |
| 2.11 | BRANCH..... | 16 |
| CHAPTER 3 - CHANGED COMPONENTS..... | | 17 |
| 3.1 | GENERAL | 17 |
| 3.2 | FINANCIAL DATA TO POL FS | 17 |
| 3.2.1 | <i>Transaction Summarisation Overview</i> | 17 |
| 3.2.2 | <i>EPOSS Changes</i> | 21 |
| 3.2.3 | <i>End of Day Branch Summarisation</i> | 23 |
| 3.2.4 | <i>TPS Harvester</i> | 29 |
| 3.2.5 | <i>TPS Host</i> | 30 |
| 3.3 | POUCH COLLECTION AND DELIVERY | 31 |
| 3.3.1 | <i>Cash Pouch Delivery</i> | 31 |
| 3.3.2 | <i>Cash Pouch Collection</i> | 31 |
| 3.4 | GENERATED CASH BALANCE..... | 31 |

| | |
|--|-----------|
| CHAPTER 4 - REFERENCE DATA | 33 |
| 4.1 TYPE A REFERENCE DATA REQUIREMENTS | 33 |
| 4.1.1 Item | 33 |
| Item Version | 33 |
| 4.1.3 Transaction Mode | 34 |
| 4.1.4 Item Transaction Mode | 34 |
| 4.1.5 Item Transaction Mode Code | 36 |
| 4.1.6 Cash Account Table Line | 36 |
| TYPE B REFERENCE DATA REQUIREMENTS | 37 |
| 4.2.1 Product Mapping | 37 |
| 4.3 TYPE C REFERENCE DATA REQUIREMENTS | 37 |
| 4.3.1 Chart of Accounts Collection | 37 |
| 4.3.2 Chart of Accounts Products Collection | 39 |
| 4.3.3 Settlement Product Modes Collection | 40 |
| 4.3.4 Mode Parameters | 41 |
| 4.3.5 EPOSS Products Collection | 43 |
| 4.3.6 Buttons | 44 |
| 4.3.7 Token Definitions | 45 |
| 4.4 OTHER REFERENCE DATA REQUIREMENTS | 46 |
| 4.4.1 Type D Reference Data | 46 |
| 4.4.2 RDDS Meta Data | 46 |
| 4.4.3 Global Objects | 47 |
| 4.5 OTHER REFERENCE DATA ISSUES | 48 |
| CHAPTER 5 - SYSTEM QUALITY ATTRIBUTES | 49 |
| 5.1.1 Security | 49 |
| 5.1.2 Scalability | 49 |
| 5.1.3 Resilience | 49 |
| CHAPTER 6 - SPECIFICALLY EXCLUDED | 50 |
| 6.1 SERVICE LEVEL MEASUREMENT | 50 |
| 6.2 BRANCH REPORTING REQUIREMENTS | 50 |
| CHAPTER 7 - MIGRATION | 51 |
| 7.1 OVERVIEW | 51 |
| 7.2 FINANCIAL DATA TO POL FS | 51 |
| 7.3 GENERATED CASH BALANCE TO SAP ADS | 52 |
| 7.4 REMITTANCE IN/OUT | 52 |
| 7.4.1 Remittance-Out | 53 |
| 7.4.2 Remittance-In | 55 |
| 7.5 EPOSS | 55 |
| CHAPTER 0 - DOCUMENT CONTROL | 2 |
| 0.1 - DOCUMENT HISTORY | 2 |

COMMERCIAL IN CONFIDENCE

| | | |
|---------------------------------------|-----------------------------|-----------|
| 0.2 | REVIEW DETAILS | 2 |
| 0.3 | ASSOCIATED DOCUMENTS | 3 |
| 0.4 | ABBREVIATIONS & DEFINITIONS | 3 |
| 0.4.1 | Abbreviations | 3 |
| 0.4.2 | Definitions | 4 |
| 0.5 | CHANGES IN THIS VERSION | 5 |
| 0.5.1 | Changes in Version 0.2 | 5 |
| 0.6 | CHANGES EXPECTED | 5 |
| 0.7 | CONTENTS | 5 |
| 0.8 | TABLE OF FIGURES | 9 |
| 0.9 | TABLE OF TABLES | 10 |
| CHAPTER 1 – INTRODUCTION | | 11 |
| 1.1 | PURPOSE | 11 |
| 1.2 | SCOPE | 11 |
| 1.3 | READERSHIP | 12 |
| 1.4 | RELATED DOCUMENTS | 12 |
| CHAPTER 2 – ARCHITECTURE | | 13 |
| 2.1 | COMPONENTS | 13 |
| 2.2 | SAP ADS | 13 |
| 2.3 | LFS HOST | 14 |
| 2.4 | LFS LOADER | 14 |
| 2.5 | LFS HARVESTER | 14 |
| 2.6 | TPS HARVESTER | 14 |
| 2.7 | TPS HOST | 14 |
| 2.8 | POLFS | 15 |
| 2.9 | RDMC | 16 |
| 2.10 | CASH CENTRE | 16 |
| 2.11 | BRANCH | 16 |
| CHAPTER 3 – CHANGED COMPONENTS | | 17 |
| 3.1 | GENERAL | 17 |
| 3.2 | FINANCIAL DATA TO POLFS | 17 |

COMMERCIAL IN CONFIDENCE

| | | |
|--|---------------------------------------|-----------|
| 3.2.1 | Transaction Summarisation Overview | 17 |
| 3.2.2 | EPOSS Changes | 21 |
| 3.2.3 | End of Day Branch Summarisation | 23 |
| 3.2.4 | TPS Harvester | 29 |
| 3.2.5 | TPS Host | 30 |
| 3.3 | POUCH COLLECTION AND DELIVERY | 31 |
| 3.3.1 | Cash Pouch Delivery | 31 |
| 3.3.2 | Cash Pouch Collection | 31 |
| 3.4 | GENERATED CASH BALANCE | 31 |
| CHAPTER 4 – REFERENCE DATA | | 33 |
| 4.1 | TYPE A REFERENCE DATA REQUIREMENTS | 33 |
| 4.1.1 | Item | 33 |
| 4.1.2 | Item History | 33 |
| 4.1.3 | Item Transaction Mode | 34 |
| 4.1.4 | Item Transaction Mode Code | 34 |
| 4.1.5 | Cash Account Table Line | 34 |
| 4.1.6 | Product Mapping | 34 |
| 4.2 | TYPE C REFERENCE DATA REQUIREMENTS | 35 |
| 4.2.1 | Chart of Accounts Collection | 35 |
| 4.2.2 | Chart of Accounts Products Collection | 36 |
| 4.2.3 | Settlement Product Modes Collection | 37 |
| 4.2.4 | EPOSS Products Collection | 38 |
| 4.2.5 | Buttons | 41 |
| 4.2.6 | Desk Top Apps | 41 |
| 4.2.7 | Token Definitions | 41 |
| 4.3 | OTHER REFERENCE DATA REQUIREMENTS | 41 |
| 4.3.1 | Type D Reference Data | 41 |
| 4.4 | OTHER REFERENCE DATA ISSUES | 42 |
| CHAPTER 5 – SYSTEM QUALITY ATTRIBUTES | | 43 |
| 5.1.1 | Security | 43 |
| 5.1.2 | Scalability | 43 |
| 5.1.3 | Resilience | 43 |
| CHAPTER 6 – SPECIFICALLY EXCLUDED | | 44 |
| 6.1 | SERVICE LEVEL MEASUREMENT | 44 |
| 6.2 | BRANCH REPORTING REQUIREMENTS | 44 |
| CHAPTER 7 – MIGRATION | | 45 |
| 7.1 | OVERVIEW | 45 |
| 7.2 | FINANCIAL DATA TO POLFS | 45 |
| 7.3 | GENERATED CASH BALANCE TO SAP ADS | 46 |
| 7.4 | REMITTANCE IN/OUT | 46 |
| 7.4.1 | Remittance Out | 46 |
| 7.4.2 | Remittance In | 49 |

Formatted

~~7.5 EPOSS 49~~

0.8 TABLE OF FIGURES

[Figure 1 – End to End Release 1 Impacted Components](#)1313
[Figure 2 – Entity Relationship](#)1818

0.9 TABLE OF TABLES

[Table 1 – Example Products](#)2222
[Table 2 – Example Product Transactions](#)2222
[Table 3 – Settlement Transactions](#)2323
[Table 4 – Products](#)3333
[Table 5 – Product Histories](#)3434
[Table 6 – Item Transaction Mode](#)3634
[Table 7 – Item Transaction Mode Code](#)3634
[Table 8 – Product Mapping](#)3735
[Table 9 – Chart of Accounts Data](#)3936
[Table 10 – Chart of Accounts Products Collection](#)4037
[Table 11 – Settlement Product Modes Collection](#)4138

Chapter 1 - Introduction

1.1 PURPOSE

As part of the Horizon End-to-End Re-Architecting, seventeen individual system enhancement projects have been identified. Project 1 and Project 3 requirements are now mature and delivery of functionality to satisfy the requirements therein is expected within the BI3 S60 timescales.

This is referred to as E2E Release 1.

This document details the high level design of the requirements stated in the following documents:

- [CD001] Project 1 Conceptual Design
- [CD002] Project 3 Conceptual Design
- [DPR02] Release 1 End-to-End Design Proposal

Many of the impacts of the changes proposed are incurred within the LFS product. The High Level Design that documents these changes are included within the following document:

- [LFSHLD] LFS Release 1 Delta HLD

This document describes the High Level Design for Release 1 of those Horizon subsystems that are not a part of the Horizon Logistics Feeder System.

1.2 SCOPE

This document describes the High Level Design for the End to End Release 1 Projects 1 and 3 and is constrained to the boundaries of the Horizon automated system.

Since much of the LFS High Level Design is documented in a separate document ([LFSHLD]), it is not proposed to repeat such designs within this document. Where this is the case, the reader will be directed to the document as appropriate.

In addition, separate interface specifications exist that detail the nature and content of the data that flows across the Horizon system boundaries. Again, in order to prevent repetition within this document, references are made to [FSAIS] & [LFSAIS] as appropriate.

This document therefore delivers the High Level design for all parts of End to End Release 1 that are not considered within the scope of LFS. In essence, this reduces the scope of this document to Release 1, Project 1. However, an overview of the

entire Release 1 project is provided to give continuity with the Design Proposal ([DPR02]) and to cross-refer to the LFS designs.

1.3 READERSHIP

This document is intended for application developers concerned with development of E2E Release 1. It is also intended to provide a detailed understanding of the software impacts incurred by E2E Release 1 as an aid to devising testing strategies and scripts.

It is suggested that readers have already read and understood [DPR02] before continuing.

1.4 RELATED DOCUMENTS

See section 0.3 for a full list of referenced documents

Chapter 2 - Architecture

2.1 COMPONENTS

The following diagram illustrates the main components of the end-to-end Release 1

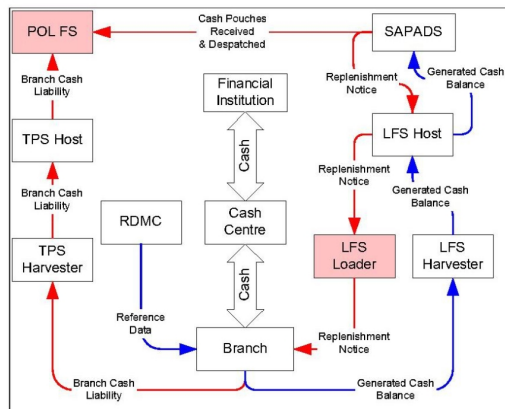


Figure 1 – End to End Release 1 Impacted Components

- ❑ Lines in Red represent new information flows
- ❑ Lines in Blue represent modified information flows
- ❑ Boxes represent subsystems or institutions and are described below. The red boxes are new sub-systems at End-to-End Release 1.
- ❑ The Direct flow of data from SAP ADS to POL FS is not within the boundary of the Horizon System

2.2 SAP ADS

External to the Horizon system, this is a SAP system that is responsible for the control of cash distribution to/from the Branches. SAP ADS will be responsible for providing details of intended deliveries of cash to Branches. The details will include the branch information, the number of cash pouches and the value of each cash denomination in each pouch. Each set of information regarding a delivery of Cash to a Branch is known as a Replenishment Delivery Notice. Replenishment Delivery Notices will be

automatically matched with deliveries of cash to Branches and will allow automated remittance-in of the cash at the point of delivery.

In addition, the total Cash balance at each Branch is calculated and sent back to SAP ADS to enable a better understanding of both the Branch and the overall Cash Holding. This information will piggy-back on the existing Cash Statement information flow.

2.3 LFS HOST

The LFS Host acts as a deliverer and receptor of information to/from SAP ADS. Changes will be needed to LFS to receive the new Replenishment Delivery Notices and to pass this information to a new LFS Loader process in order to deliver such information to the correct Branches.

The LFS Host also needs minor changes to enable it to accept the additional Cash Statement attribute that reflects the Branch Cash Balance.

Full details of the changes required to the LFS Host are contained within [LFSHLD]. Additionally, the changes and specifications of the interface between SAP ADS and the LFS Host are documented in [LFSAIS].

2.4 LFS LOADER

A new Agent loader process is required to deliver the Replenishment Delivery Notices from the LFS Host to the target Branches.

Full details of the new Loader process are described in [LFSHLD]

2.5 LFS HARVESTER

The existing LFS Harvester will be modified to additionally harvest the generated cash balance attribute each day.

Full details of the changes required are described in [LFSHLD]

2.6 TPS HARVESTER

The existing TPS Harvester Agent will be modified to harvest end of day cash (and Near-Cash) movement summaries to the TPS Host system. Full details of the changes required to this Harvester are described in Section 3.2.4 within this document.

An overview of the accounting process and the mechanisms of transaction summarisation are described in section 3.2 of this document.

2.7 TPS HOST

The TPS Host process is responsible for delivery of transactional data and summaries to both the POL TIP system and the Data Warehouse. This functionality will be extended to pass the new Cash Movement Summaries to the POL Financial System (POL FS).

Details of the changes required are summarised in Section 3.2.5 of this document and will be further clarified during the update of [TPSHLD].

The output format of the cash movement summaries generated by TPS Host is fully described in [FSAIS].

2.8 POL FS

The POL Financial System is a new SAP system that will eventually replace the existing set of CBDB financial systems. This system is out of scope of this document however a brief description is in order.

The financial system will receive cash movement data from the Branches via the TPS sub-system, from the Cash Centres (probably via SAP ADS) and from the Financial Institutions (probably manually entered from statements).

The POL FS implements a double-entry bookkeeping system such that each posting to an account must have an equal and opposite posting to another (or same) account. The net result of this mechanism is that the balance of the accounts is always zero.

The aim at Release 1 is to account for Cash Holdings at both the Branches and at the Cash Centres. At Release 2, this accounting will be extended to account for all transactions.

The following Branch transactions therefore need to be identified:

- Cash/Cheques Received during payment for goods during Serve Customer
- Cash Paid as benefits, banking transactions, reversals during Serve Customer
- Debit Card payments for services (Bureaux & Non-Bureaux)
- Foreign exchange Transactions
- Cash Received in Pouches from Cash Centres
- Cash Returned in Pouches to Cash Centres
- Cheques returned to the Cheque clearance centre.
- Cash movement between stock units (although these movements do not affect the balance, they should be included for completeness)
- Discrepancies in cash position arising from the difference between the Horizon calculated cash position and the manually declared cash position

These transactions will be identified by Product reference data (as described in section 3.2), summarised, mapped to the relevant Chart of Accounts code and sent to the POL FS on a daily basis.

In order to compile balancing double-entry accounts, both the cash movements and the product movements require capture such that they net to zero. Since the product movements will not be captured at Release 1 then a dummy 'net-out' balancing movement needs to be generated and sent to PO LFS. This balancing figure will simply be equivalent and opposite to the sum of all the transactions sent to POL FS.

2.9 RDMC

The RDMC/RDDS system is responsible for delivering reference data updates to the Horizon estate. At Release 1, the reference data updates will be constrained to Type-C Data, this means that there is no impact on the RDMC/RDDS software deliverables.

The new requirements for reference data are described in Chapter 4

2.10 CASH CENTRE

The POL Cash Centres are not a part of the Horizon system and are therefore out of scope of this document. A brief description follows.

The cash centre is responsible for delivering cash to the Branches and receiving cash back from the Branches. Cash is packed in Pouches for delivery (either way) and contains a known (and recorded) amount for each denomination.

The despatch and receipt of each pouch is recorded in SAP ADS and the Pouch Id and Total Value passed from SAP ADS to POL FS. POL FS will match these data to the despatch and receipt of Pouches at the Branches to ensure that no Pouch is 'lost' in transit. Matching will be by Pouch Id that is recorded on the transaction detail that is sent to POL FS.

2.11 BRANCH

The branch counter systems require changes in the following areas:

1. End of Day Summarisation of Cash for POL FS
2. Generation of the daily total Cash Balance for return to SAP ADS
3. Automated Remittance-in of cash at the point of delivery via reference to the Replenishment Delivery Notices
4. Deferral of remittance-out of cash until the actual point of collection by the courier
5. Reversal of remittance-out prior to collection
6. Electronic preparation of a group of pouches awaiting collection by courier

Item 1 is fully described in Section 3.2. All of the other changes to the counter systems are detailed in [LFSHLD].

Chapter 3 - Changed Components

3.1 GENERAL

This section describes all the Horizon subsystem components that require change as a result of both the new Requirements defined for E2E Release 1 Projects 1 and 3 and as a result of guidance provided in the End-to-End Release 1 Design Proposal.

3.2 FINANCIAL DATA TO POL FS

At E2E Release 1, all Cash and Near Cash movements are to be posted to the new POL Financial system on a daily basis. Most of the movements of cash will be summarised at a Branch level by mapping each transacted Product to a POL FS Chart of Accounts reference.

Summarisation will be performed at each branch at the end of each trading day and these resulting summaries will be harvested by the TPS Agent and routed to POL FS via the TPS Host System Database.

3.2.1 Transaction Summarisation Overview

The POL FS will present Cash and Near Cash movements against a number of different Chart of Accounts Accounting Codes. Each C-of-A Account will consist of the summarisation of transactions of differing Products transacted in each Branch.

The summarisation process is to be performed daily and is to only summarise transactions performed on each trading day. The data summarised is therefore a daily movement figure rather than a balance figure.

Not all transactions are to be summarised. At E2E Release 1, only Cash and Near-Cash Products require summarisation to one of the following accounts:

- Summary: Cash on Hand movement
- Summary: Cheques on Hand movement
- Transactions: Cash Rem In Value
- Summary: Bureau Card transactions Value
- Summary: Non-Bureau Card transactions Value
- Summary: Cheque Rem Out Value
- Transactions: Cash Rem Out Value
- Summary: Forex on Hand movement (in £)
- Balance (sum of all transactions posted to the above accounts)

For each Product that will be summarised, the C-of-A Account to which it will be mapped will be recorded within the Product definition.

Some C-of-A accounts need individual transactional detail rather than summaries. Products that are mapped to these accounts will not be summarised but will instead be transferred to POL FS as individual transactions. Examples of these transactions are:

- Remit Out Cash to Automated Distribution Centre
- Remit In Cash from Automated Distribution Centre

Where individual transactional detail is passed to POL FS, an agreed Transaction Identifier must accompany it. In the case of cash remittances, this will be the cash Pouch Id. Future requirements for individual transactional details are unknown and it should therefore be possible to use Reference Data to specify which transaction attribute should be passed to POLS FS as the Transaction Identifier.

The volume of these individual non-summarised transactions on any trading date is expected to be low.

At Release 2, the mapping of cash transactions to the POL FS Chart of Accounts is to be extended to all Products

The logical reference data required to support this is as follows:

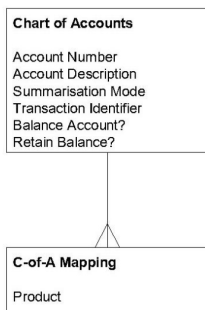


Figure 2 – Entity Relationship

Where:

Account Number: Chart of Accounts Account number. It is assumed that this is numeric

Account Description: The description of the POL FS Account (Not really needed, other than as documentary info).

Summarisation Mode: Currently, the two modes that are required are either summarisation of all daily transactions to a single figure or no summarisation at all.

Transaction Identifier: If the Summarisation Mode indicates that the transactions are not to be summarised, then this

| | |
|-------------------|---|
| | indicates which piece of attribute grammar is stored as the transaction identifier. |
| Balancing Account | A Boolean indicator to highlight which CofA Account the Cash Balancing Figure should be posted to. This attribute is only required for Release 1 whilst the dummy balancing figure is required. Only <i>one</i> CofA Account should have this Boolean set to TRUE |
| Retain Balance | A Boolean flag that determines whether a balance figure should be maintained for the CofA account within each branch. This should always be set to TRUE if Cash Indicator is set to TRUE (see section 3.2.3.5). |
| Product: | A POL Product code |

The above assumes that the mechanism for transaction summarisation is consistent for any C-of-A account. If a C-of-A account requires both summarised and individual transaction details to be posted to it, then the Summarisation Mode will need to be stated within the C-of-A Mapping.

Since there is already a definition of each Product, a modification to this definition to include the C-of-A mapping would be ideal. The Product definition is delivered as Type-A Reference Data as 'Item' and this is transformed into the 'EPOSSProducts' collection that is then made available to the counter.

However, changes to Type A reference data are likely to be difficult to implement within Release 1 timescales. It is therefore proposed that the reference data be delivered as Type-C.

If the Reference data is delivered as Type-C data, then the EPOSSProducts collection cannot initially be used to store the new Chart of Accounts Product Mapping. A new collection will need to be devised that has a similar structure to the EPOSSProducts collection so that, when the data is eventually delivered as Type-A, the code changes required at the counter to use the EPOSSProducts collection (rather than the new collection defined below) will be minimal.

Refer to section [4.3.24.2.2](#) for details of the CofAProducts Collection.

In addition to the CofAProducts collection, a ChartOfAccounts collection also requires definition.

Refer to section [4.3.14.2.1](#) for details of the ChartOfAccounts Collection.

There are two ways in which the transactions can be summarised at the end of each day:

1. The new End-of-day summarisation process can read all transactions performed during the day and perform a lookup on the CofAProducts collection to determine the C-Of-A mapping and the ChartOfAccounts collection is used to determine whether summarisation is required or individual transaction detail is required.
2. The EPOSS Retail Broker is modified to de-normalise the *mapping* onto the transaction itself at the time that the transaction is written to the message store. The End-of-Day summarisation process therefore only needs to scan those

transactions that have a mapping attribute and referral to reference data is required only for the ChartOfAccounts Collection.

There are a number of benefits to option 2:

- ❑ The EPOSS Code already reads the EPOSSProducts collection at the time of performing a transaction. The data is therefore already available.
- ❑ The End of Day summarisation process is simplified and more efficient.
- ❑ If a reference data change is made during a period when a gateway machine is faulty, then summarisation of prior-days transactions will use the most recent reference data if option 1 above is chosen. However, Option 2 guarantees that the correct reference data is used as long as the reference data is available at the Branch

The downside to option 2 is that the Transaction itself is increased in size. However, this increase is only due to the addition of a single Chart of Accounts Code.

Overall, Option 2 is the better proposal and is the chosen option.

3.2.1.1 Settlement Products

Sessions that are performed in various modes settle automatically to a default Settlement Product that is defined in the Collection 'ModeParameters'. These Modes are:

- ❑ Parcel Traffic
- ❑ Revaluation Down
- ❑ Revaluation Up
- ❑ Remit-in Auto Distribution
- ❑ Remit-in Client
- ❑ Remit-out Auto Distribution
- ❑ Remit-out Client
- ❑ Remit Out Data Centre
- ❑ Transfer In
- ❑ Transfer Out

Additional

In many of these modes, the Session may consist of both Cash and Stock transactions and the total value of the Session is settled as a single balancing transaction to the Settlement product.

Due to the need to report *all* Cash transactions to POL FS in order to ultimately provide balancing double-entry accounts, the Cash portion of all Settlement transactions requires separate identification from any Stock portion. It is therefore necessary to have a separate Settlement Product for Cash than for Stock for those Product/Mode combinations that allow Cash and Stock to be mixed within the same Session (and which auto-settle).

The Product/Mode combinations that require Settlement to a non-default Product are the exception to the rule. It is therefore proposed that the *default* Settlement Product is retained within the 'ModeParameters' Collection and that, exceptionally/optionally,

Formatted

COMMERCIAL IN CONFIDENCE

a different Settlement Product is defined within the ProductModes Collection. However, since we are not changing Type-A reference data at Release 1, the Exceptional/Optional Settlement Product will be defined in a CofAProductModes collection.

Refer to Section [4.3.34.2.3](#) for full details of the CofAProductModes Collection.

3.2.1.2 Transfer Out/In

Due to various complexities within the generic EPOSS core code, the use of multiple settlement products during Transfer Out/In is not desirable. However, the settlement of cash during transfers out/in needs to be differentiated from the settlement of stock otherwise an imbalance would be posted to POL FS.

However transfers of stock between stock units do not affect the overall stock holding or cash balance of the branch. These transfers are therefore of no accounting use.

It is therefore proposed that Transfer Out/In transactions are ignored during the summarisation process (Section 3.2.3) and are therefore not reported to POL FS.

The use of transfer in (TI) and transfer out (TO) modes within the CofAProductModes collection is therefore expressly prohibited.

3.2.1.3 Discrepancies

Discrepancies between roll-over declarations and the values deduced within the Horizon Branch System are posted to a Discrepancy Surplus or a Discrepancy Shortage Product.

For Release 1, it is proposed that Discrepancy Surplus and Discrepancy Shortage Products are ignored and not posted to POL FS.

3.2.2 EPOSS Changes

Each transaction is to be written with the C-of-A mapping account as an additional attribute *EPOSSTransaction.CofA*.

In addition, where the transaction Mode causes a Session to be settled to a default Settlement Product, then each Product on the transaction stack will need individual inspection to determine whether the value for the Product Transaction should settle to an exceptional Settlement Product. An example follows:

Consider the following example Products:

| Product Number | Description |
|----------------|--|
| 1 | CashCheques |
| 2 | First Class Stamps |
| 3 | £10 Postal Orders |
| 4 | Second Class Stamps |
| 5 | Txfr ROAD Dummy Product |
| 6 | Txfr ROAD Cash Cheque Product |

Formatted

COMMERCIAL IN CONFIDENCE

Table 1 – Example Products

And assume that the Default Settlement Product for Transaction Mode '~~Transfer ROADOut~~' is Product number 5. However, we wish to account for ~~Cash-Cheques~~ separately to the way in which we account for Stock and we wish to post the ~~cash Cheques~~ to the Chart of Accounts account code '~~00001-00009~~'. We would therefore make the following entries in the CofAProductModes Collection:

```
<Collection:CofAProductModes>
<ObjectName:1>
<Data:
  <Modes:
    <Mode:
      <M:TOROAD>
      <V:1>
      <S:6>
    >
  >
>
```

As can be seen, the ObjectName '1' (Product 1), when transacting in Transfer-out mode ('~~ROADTO~~'), has an entry in this mode indicating that the product is settled to Product 6 ('S:6').

When performing the following Transaction in Transfer-Out Mode:

| Product Code | Value |
|--------------|-------|
| 1 | £100 |
| 2 | £25 |
| 3 | £10 |
| 1 | £20 |
| 4 | £36 |

Table 2 – Example Product Transactions

The EPOSS system should determine the total values by each Settlement Product. Since Cash Settles to Product 6 and the other Products Settle to the Default Settlement Product (Product 5), then the Settlement Transactions are as follows:

| Product Code | Value |
|--------------|-------|
| 5 | £71 |
| 6 | £120 |

Table 3 – Settlement Transactions

Initially, this will require access to the new CofAProductModes collection. In the fullness of time, the CofAProductModes collection will be replaced by the ProductModes collection.

Note: It is unclear whether transaction creation is centralised within EPOSS Core or whether multiple areas of code will require change

I suspect it is common from EPOSS, NBS and DCS, but is separate for APS, OBCS and Mails. Need to get Chris or the Developers to confirm this.

3.2.3 End of Day Branch Summarisation

A new End of Day Branch Summarisation process is required. The existing End-of-Day Reconciliation module will be cloned and modified to meet the requirement since it contains existing infrastructure that scans all messages for each outstanding Trading Date. This caters for scenarios where catch-up is required if previous end-of-day processes have failed to run. It should be noted that catch-up should be attempted for all days in arrears up to the point where transactions are archived (the existing Daily Reconciliation only attempts catch-up for a limited period in arrears). The most recent EOD Branch Summarisation Trailer will be located via reference to a new Persistent Object that is updated at the end of every daily summarisation to point to the most recent trailer record.

The End of Day Reconciliation is documented in [EOD]. An extract from this document is as follows:

Single Day's Reconciliation

A further sub-process within Daily Reconciliation acts to perform the reconciliation for a single day between two end of day markers, processing all messages between these markers. At system installation there will be only a single marker however to delineate the first day's processing. This is taken into account.

The process to perform daily reconciliation reads all messages for a day and uses selection criteria to determine whether a message constitutes an EPOSS Transaction for reconciliation. Selection of a transaction has that message submitted to two aggregation processes as a result. The process is ended with the writing of the day's reconciliation details.

```

Read and Cache ModeParameters Reference Data
Locate End of Day Marker for yesterday
If not Found then
    ???Find a first message
Locate End of Day Marker for Day of Reconciliation
If not Found then
    Record System Error to Event Log
    Abandon EPOSS End of Day
EndIf
TxnSelected = Select_Transactions(Messages)
For each Message written between the two Markers
    Read Message
    TxnValidated = Validate (Transaction)
    Perform Transaction Accumulation
    Perform Mini Cash Account Accumulation
Loop
Write Daily Reconciliation Details

```

The above pseudo-code logic will be modified as follows:

```

Locate End of Day Marker for Day_of_Summarisation - 1
If not Found then
    ???Find a first message
Locate End of Day Marker for Day_of_Summarisation
If not Found then
    Record System Error to Event Log
    Abandon EPOSS End of Day

```

```

EndIf
TxnSelected = Select_Transactions(Messages)
For each Message written between the two Markers
  Read Message
  TxnValidated = Validate (Transaction)
  Perform Chart of Accounts Accumulation
Loop
Write Chart of Accounts Summaries
Write Chart of Accounts Trailer
Update Cash Statement Persistent Object

```

The new functions (Shown in bold) are described in more detail below.

3.2.3.1 Perform Chart of Accounts Accumulation

This function initially processes only those transactions that have attribute *EPOSSTransaction.CofA* present on the Riposte Message.

Note: In future releases, all transactions (other than those in TI or TO mode) should be mapped to a Chart of Accounts code and an error would then be raised if the EPOSSTransaction.CofA attribute did not exist on any one transaction. In Release 1, the transaction scan will simply pick-up the transactions that have the CofA attribute present and are not in modes TI or TO.

If the Summarisation Mode as found by lookup of the ChartOfAccounts Collection indicates that transaction detail is to be sent to POL FS, then each **non-zero value transaction** (*EPOSSTransaction.ValueCount != 0*) will be reproduced (duplicated) in the message store within the end-of-day Branch Summarisation. The following message will be written for each of these transactions:

```

<Application:CofASummary>
<EPOSSTransaction:
  <TranType:CofATransaction>
  <EODDate:dd-mmm-yyyy>
  <EODTime:hh:mm:ss>
  <Id:Node_Id>
  <Num:Txn_Num>
  <CofAMapping:CofACode>
  <TxnId:TransactionId>
  <MessageCount:Messages>
  <QtyCount:Quantity>
  <ValueCount:Value>
  <CofASummaryTrailerId:Identifier>
>

```

Where:

dd-mmm-yyyy: Trading Date associated with the transaction
hh:mm:ss: End of Day Time
Node_Id: Message.Id (From the original transaction within the message store)
Txn_Num: Message.Num (From the original transaction within the message store)
CofACode: Chart of accounts code recorded in *EPOSSTransaction.CofA*
TransactionId: The value taken from an attribute within the transaction. The specific attribute itself is determined by which chart of accounts code the transaction is mapped-to. The attribute is

determined by the value held in Collection *ChartOfAccounts* Attribute *TxnAttribute*

Messages: This will always be set to '1'. This value is documentary only and is not passed to POL FS.

Quantity: This is set to the *EPOSSTransaction.Qty* attribute. This value is documentary only and is not passed to POL FS.

Value: This is set to the *EPOSSTransaction.SaleValue* attribute

Identifier: This is the unique identifier associated with the current CofA Summary

If the Summarisation Mode indicates that the transaction is to be *summarised* before sending to POL FS, then the transaction will not be duplicated within the CofA Summary.

Regardless of Summarisation Mode each transaction will be added to an internal (in-memory) summary for the correct C-of-A account. Against each Chart of Accounts code, the values for Transaction Quantity, Transaction Value and the count of the contributing records will be accumulated.

3.2.3.2 Write Chart of Accounts Summaries

On completion of the message store scan, all the *non-zero value* (*EPOSSTransaction.ValueCount != 0*) internal C-of-A summaries with a Summarisation Mode of 'S' will be written to the message store. The following message will be written for each summary:

```
<Application:CofASummary>
<EPOSSTransaction:
  <TranType:CofASummary>
  <EODDate:dd-mmm-yyyy>
  <EODTime:hh:mm:ss>
  <CofAMapping:CofACode>
  <MessageCount:Messages>
  <QtyCount:Quantity>
  <ValueCount:Value>
  <CofASummaryTrailerId:Identifier>
>
```

Where:

dd-mmm-yyyy: Trading Date associated with the transaction

hh:mm:ss: End of Day Time

CofACode: Chart of accounts code recorded in *EPOSSTransaction.CofA*

Messages: The count of the number of messages that were used to create this summary accumulation. This value is documentary only and is not passed to POL FS.

Quantity: The sum of the *EPOSSTransaction.Qty* attributes. This value is documentary only and is not passed to POL FS.

Value: The sum of the *EPOSSTransaction.SaleValue* attributes

Identifier: This is the unique identifier associated with the current CofA Summary

Having written all the CofA summary records to the message store, a balancing transaction needs to be written with a value that is equal to and opposite to the sum of all Summarised and non-Summarised transactions. The Chart of Account code to

which this 'Balancing Transaction' is written is the code within the *ChartOfAccounts* Collection that has attribute *Data.BalanceFlag* = 'Y'.

3.2.3.3 Write Chart of Accounts Trailer

The Attribute Grammar of the Chart of Accounts Trailer Message, written last is as follows:

```
<Application:CofASummary>
<EPOSTransaction:
  <TranType:CofATrailer>
  <EODDate:dd-mmm-yyyy>
  <EODTime:hh:mm:ss>
  <PreviousTrailer:PrevTrailerId>
  <CofASummaryTrailerId:Identifier>
>
```

Where:

dd-mmm-yyyy: Trading Date associated with the transaction
 hh:mm:ss: Time of transaction
 PrevTrailerId: The Id of the previous CofA Summary Trailer record. This will be used by the Agent during catch-up following EOD failure for one or more days.
 Identifier: This is the unique identifier associated with the current CofA Summary

The total messages, total quantity and total value attributes are recorded for audit purposes. In Release 1, they are additionally used in conjunction with the CofACode by the TPS Harvester to generate the balancing transaction for the Summarised Cash transactions.

3.2.3.4 Write Trailer Persistent Object

A Trailer object is required to point to the most recent Chart of Accounts trailer record. This prevents processor-binding searches for the latest trailer record.

```
<Collection:CofABSP>
<ObjectName:Trailer>
<EPOSTransaction:
  <TranType:CofATrailer>
  <EODDate:dd-mmm-yyyy>
  <EODTime:hh:mm:ss>
  <CofASummaryTrailerId:Identifier>
  <PreviousTrailer:PrevTrailerId>
>
```

Where:

dd-mmm-yyyy: Trading Date associated with the transaction
 hh:mm:ss: Time of transaction
 PrevTrailerId: The Id of the previous CofA Summary Trailer record. This will be used by the Agent during catch-up following EOD failure for one or more days.
 Identifier: This is the unique identifier associated with the current CofA Summary

3.2.3.5 Update Cash Statement Persistent Objects

The balance of Cash on Hand at each Branch needs to be recorded at the point in time when the cash movement figures are compiled for POL FS. This Cash on Hand figure will be used within the LFS Cash Statement message as the *Data.GeneratedCash* figure.

The balance figure will be derived from the previous days' balance plus the movement for today.

Since there is a requirement to store a balance figure for cash, then the mechanism should be extended to store a balance figure for each chart of accounts code. This makes the process more generic and aids the testing of the product. Each balance figure is stored in a *local* persistent object as follows:

```
<Collection:CofABalance>
<ObjectName:CofACode>
<RData+
  <Data:
    <EODDate:dd-mmm-ccyy>
    <Balance:BalanceAmt>
  >
  >
<Version:>
<Deleted+>
```

Where:

CofACode: Chart of accounts code which this balance figure represents.

dd-mmm-ccyy: Trading Date when the balance figure was last calculated.

BalanceAmt: The value of the balance for this Chart of Accounts Code.

Following the calculation of the End of Day Branch Summarisation, the CofABalance persistent objects will be updated with the total movement figures. If the persistent object did not exist prior to execution (due to the introduction of a new CofA Code) then the value of the persistent object is assumed zero and a new persistent object created with a value equivalent to the current days' movement figure.

Note: The balance consists of the previous days' balance figure plus the value of the transactions performed today. Therefore in cases where the End of Day Branch Summarisation process has failed to run for one or more days, then catch-up must be performed starting with the last successful run and must always operate in chronological order. This will be performed on a day-by-day basis such that a set of CofABalance messages and a trailer record will be written for each Trading Date

Note: Balance persistent objects will only be created/maintained for those Chart of Account codes where Collection ChartOfAccounts.Data.BalanceFlag = 'Y'. There is only ever expected to be one such persistent object required.

3.2.3.6 Migration Processing

The migration to the End of Day Branch Summarisation process will be controlled by a Soft-launch Product. The presence of the Product will be checked during start-up of the process and if the Product does not exist, then the Summarisation process will remove all persistent objects referred to in section 3.2.3.5 and then exit without further action.

If the Soft-launch Product does exist, then the process needs to check whether this is the first time that the process has been executed (1st run after migration). This is done by checking for the presence of one or more persistent objects that are described in Section 3.2.3.5. If one or more of these objects exists, then this process must have previously executed and no further migration issues need to be considered. If there are no Summarisation Persistent objects, then this must be the first execution of the process following Migration and the following migration activities must be performed:

For each row in CofAProducts Collection, check the Summarisation Mode as found by lookup of the ChartOfAccounts Collection indicates that transaction Summary is to be sent to POL FS. For each Product that requires summarisation, read the associated Container #1 record from the most recent Cash Account Rollover balance figure. Where this message exists, summarise all transactions for the Product that lie between the Cash Account Rollover Trailer and the start of the current day and add both the Rollover Balance and the Transaction Total to the in-memory Summary totals (See Section 3.2.3.1).

Note: At the point of Migration, ALL the products that are present in the CofAProducts collection will be migrated/Bforward from the previous Cash Account rollover balances. However, if products are subsequently added to the CofAProducts collection, it will be assumed that they are new products and no further migration activity will take place.

At Release 2, the migration of stock Products will need to be considered. It is assumed that the CofA mapping will be available via Type A Reference Data and the migration will provide Balance figures for all products that do NOT exist in the CofAProducts collection.

3.2.4 TPS Harvester

The existing TPS harvester will be enhanced to process the new end-of-day summary records. This harvester uses a secondary scan to identify all summaries that exist between the end-of-day marker and the end-of-day trailer record. The trailer record for the C-of-A summaries will be identified by:

```
<Application:CofASummary> and  
<EPOSSTransaction.TranType:CofATrailer> and  
<EPOSSTransaction.EODDate:dd-Mon-yyyy>
```

A tertiary scan of the Gateway Counter (node 1) is initiated to find the Summary message(s) associated with the C-of-A Trailer. The low marker for this scan is the EOD Marker message itself and the high marker is the Summary Trailer.

The Summary messages are recognised by having the same values for:

```
<Application:> and
<EPOSSTransaction.CofASummaryTrailerId:> and
<EPOSSTransaction.EODDate:>
```

as the Summary Trailer.

The Summary messages to be harvested are further recognised by the value of <EPOSSTransaction.TranType:>. Depending on the value of *EPOSSTransaction.TranType*, the harvested record will be placed in either a TPS C-of-A Summary table or a TPS C-of-A Transaction Detail table.

| TranType | Harvested as | TPS Table |
|-----------------|--------------------------------------|--------------------------|
| CofATransaction | Individual Transaction detail record | TMS_RX_COFA_TRANSACTIONS |
| CofASummary | Chart of Accounts Summary record | TMS_RX_COFA_SUMMARIES |

The transactions that are harvested will be inserted into one of two TPS interface tables. The attribute mapping for these tables is as follows:

| Column | Type | Size | Dec | Def | Null? | Checks | Num | Attribute Name |
|---------------------------------|------------|--------------------------|-----|-----|----------|--------|----------|-------------------------------|
| TMS_RX_COFA_TRANSACTIONS | | TranType:CofATransaction | | | | | | |
| TRADING_DATE | DATE | | | | NOT NULL | | | *eod date |
| INSERT_DATE | DATE | | | | NOT NULL | | | SYSDATE |
| GROUP_ID | NUMBER | 6 | | | NOT NULL | | | GroupId |
| COUNTER_POSITION | NUMBER | 2 | | | NOT NULL | | | EPOSSTransaction.Id |
| RIPOSTE_MESSAGE_NUMBE | NUMBER | 10 | | | NOT NULL | | | EPOSSTransaction.Num |
| C_OF_A_CODE | NUMBER | 6 | | | NOT NULL | | | EPOSSTransaction.CofAMapping |
| TRANSACTION_QUANTITY | NUMBER | 9 | | | NOT NULL | | Rip, Num | EPOSSTransaction.QtyCount |
| TRANSACTION_AMOUNT | NUMBER | 12 | 2 | | NOT NULL | | Rip, Num | EPOSSTransaction.ValueCount |
| TRANSACTION_ID | VARCHAR2() | 32 | | | NOT NULL | | | EPOSSTransaction.TxnId |
| TMS_RX_COFA_SUMMARIES | | TranType:CofASummary | | | | | | |
| TRADING_DATE | DATE | | | | NOT NULL | | | *eod date |
| INSERT_DATE | DATE | | | | NOT NULL | | | SYSDATE |
| GROUP_ID | NUMBER | 6 | | | NOT NULL | | | GroupId |
| C_OF_A_CODE | NUMBER | 6 | | | NOT NULL | | | EPOSSTransaction.CofAMapping |
| TOTAL_TRANSACTIONS | NUMBER | 6 | | | NOT NULL | | Rip, Num | EPOSSTransaction.MessageCount |
| TOTAL_TRAN_QUANTITIES | NUMBER | 9 | | | NOT NULL | | Rip, Num | EPOSSTransaction.QtyCount |
| TOTAL_TRAN_AMOUNTS | NUMBER | 12 | 2 | | NOT NULL | | Rip, Num | EPOSSTransaction.ValueCount |

Note: The above table will eventually be used to update document [TPSMAP]. A full description of the meanings of the columns and contents can be found within that document.

3.2.5 TPS Host

The TPS Host system will present two new views to the TPS Harvester into which the C-of-A transaction summaries and the C-of-A individual transactions will be written during the harvesting process. These tables are defined in the previous section.

A new TPS process will extract and transform the data and write it to one or more flat files on the TPS Host operating system. The format of these files and the associated error processing is defined in the POL FS Application Interface Spec [FSAIS]. The two TPS tables (defined in the previous section) will be joined using a UNION ALL, sorted and written to the Branch Ledger Entry Contents Record and the Branch Ledger Entry Detail Record as defined in [FSAIS].

Scheduling and housekeeping of the data is similar to the existing TPS TIP schedules and housekeeping tasks as defined in the TPS HLD [TPSHLD].

3.3 POUCH COLLECTION AND DELIVERY

The new POL Financial System will provide a tighter and more timely mechanism of cash management. To better facilitate this, the liability for cash needs to be precisely aligned with the delivery and collection of cash to/from each Branch.

Cash therefore needs to become part of the overall Branch balance at the point of delivery and will only reduce the branch balance at the point of collection.

This document only outlines the changes that are required, full design details are documented in:

[LFSHLD] LFS High Level Design

3.3.1 Cash Pouch Delivery

Each pouch must be recorded onto the system at the point of receipt by scanning or manually entering the barcode. The process will be changed such that the receipt of each cash pouch is matched with the associated Replenishment Delivery Notice that advises the Pouch Contents. The remittance-in of the Pouch contents will then be automated from the details of the Replenishment Delivery Notice.

3.3.2 Cash Pouch Collection

The original LFS process remitted-out the cash at the point of packing each Cash Pouch. However, collection of the pouches may be some time thereafter and therefore the liability of the Branch is reduced artificially early.

The process will change such that the cash value of the pouch being packed is transferred to a suspense account until the point of Pouch collection.

An additional process will be implemented that will enable pouches to be grouped together before collection and all associated receipts pre-printed. The collection process will therefore change such that a group of pouches is identified and collected as a single transaction. The cash balance of the branch will be reduced at the point of collection by transferring each Pouch value from the suspense account to a cash Remittance-out account.

3.4 GENERATED CASH BALANCE

A better understanding of the accuracy of LFS declared Overnight Cash Holdings that are sent to SAP ADS is required to ensure that Branches report their cash holdings accurately. Appending an automatically calculated Branch Cash Balance to the manually declared figures will do this.

The End of Day Branch Summarisation process defined in Section 3.2.3 maintains the Branch Cash Balance as values in a local Persistent object *CofABalance*. These values give the overall Branch balance that is appended to the existing *CashStatement* message for harvesting to the LFS host and onward transmission to SAP ADS.

This document only outlines the changes that are required, full design details are documented in:

[LFSHLD] LFS High Level Design Delta

Chapter 4 - Reference Data

All new reference data structures will be delivered via Type-C reference data in order to prevent impact to the RDMC/RDDS delivery mechanisms. There will however be additional items of Type-A reference data and this will need to be agreed with POL.

4.1 TYPE A REFERENCE DATA REQUIREMENTS

4.1.1 Item

New product definitions are required to achieve the Accounting requirements and also to implement the Softlaunch capability.

| Item Code | Item Type | Description |
|-----------|-----------|---|
| CIP | 1 | Cash in Pouches. This product will be defined as a cash product that is used to retain cash within the Branch until Pouch collection. It will be mapped to the Cash Account as a Suspense item in Table 2. This is a Core Product. |
| ACC | 1 | Authorised Collectors Card non-Core Product. This dummy product will be distributed only to those branches that do not require to validate the ACC Card of Cash Collection Couriers. These are the Branches where collection of cash is managed by Royal Mail Special Delivery. |
| SOF1 | 1 | Soft Launch Product 1. This non-Core Product will be used to invoke the Summarisation of data to POL FS (See Section 7.2) |
| SOF2 | 1 | Soft Launch Product 2. This non-Core Product will be used to invoke the changes to the remittance Out/In functionality (See Section 7.4) |

Table 4 – Products

Note: The Item Code to be replaced by the actual Product Code once POL have told us the codes to assign.

Note that **all** both softlaunch products **and the ACC product** must be registered in the RDDS table *svc_control_products*.

4.1.2 Item History/Item Version

This defines the Product attributes.

Formatted

| ITEM ID | LONG NAME | LOGISTICS ACCOUNTING ITEM | LOGISTICS INVENTORY ITEM | RETAIL PRICE | MULTIPLE VALUE ALLOWED | MINIMUM VALUE | MAXIMUM VALUE | MINIMUM QTY | MAXIMUM QTY | NUMBER OF RECEIPTS | OVERRIDE PRICE ALLOWED | STAFF DISCOUNT ALLOWED | CUSTOMER DISCOUNT ALLOWED | EXISTING REVERSAL ALLOWED | NEW REVERSAL ALLOWED | VALUE STOCK | REFUND ALLOWED | VOIDABLE | IS A CORE ITEM | CAN CHANGE SIGN |
|---------|-------------------------|---------------------------|--------------------------|--------------|------------------------|---------------|---------------|-------------|-------------|--------------------|------------------------|------------------------|---------------------------|---------------------------|----------------------|-------------|----------------|----------|----------------|-----------------|
| CIP | Cash in Pouches | N | N | £0.00 | £0.00 | £0.01 | £99999999 | 1 | 99999 | 0 | N | N | N | N | N | N | N | Y | N | N |
| ACC | Non-ACC Courier | N | N | £0.00 | £0.00 | £0.00 | £0.00 | 40 | 99999 | 0 | N | N | N | N | N | N | N | Y | N | N |
| SOF1 | E2E Softlaunch Product1 | N | N | £0.00 | £0.00 | £0.00 | £0.00 | 40 | 99999 | 0 | N | N | N | N | N | N | N | Y | N | N |
| SOF2 | E2E Softlaunch Product2 | N | N | £0.00 | £0.00 | £0.00 | £0.00 | 40 | 99999 | 0 | N | N | N | N | N | N | N | Y | N | N |

Table 5 – Product Histories

4.1.3 Transaction Mode

Formatted: Bullets and Numbering

Three new transaction modes are required to implement the new remit-out functionality described in [LFSHLD]. These are as follows

| Transaction Mode Code | Description |
|-----------------------|--|
| 26 | Remit-out Cash from stock unit |
| 27 | Reverse Remit-out Cash from stock unit |
| 28 | Remit-out – despatch cash pouches |

4.1.4 Item Transaction Mode

Formatted: Bullets and Numbering

A definition of which Items (products) may be transacted in which modes is required for each valid product/mode combination. For modes 26 (ROSP) and 27 (RISP), all Cash denomination products need to be mapped to the new modes.

Mode 28 (RODP) is only used to transact against the new Cash in Pouches product (product 5610).

Mappings for Cash (product 1) and Cash in Pouches (product 5610) for modes 26 & 27 are only required to maintain referential integrity with the 'Item Transaction Mode Code' entries that provide the cash account mappings.

| Item Id | Item Description | Mode Code | Accounting Sense |
|---------|-----------------------|-----------|------------------|
| 1 | Cash | 26 | - |
| 5610 | Cash In Pouches Stock | 26 | - |
| 356 | #100 Banknote | 26 | - |
| 357 | #2 Coin | 26 | - |
| 655 | #50 Banknote | 26 | - |
| 656 | #20 Banknote | 26 | - |
| 657 | #10 Banknote | 26 | - |
| 658 | #5 Banknote | 26 | - |
| 659 | #1 Banknote | 26 | - |

Fujitsu Services

End to End Release 1 - High Level Design

Ref.: EA/HLD/002

Version: 1.00-3

Date: 19/01/2004

Formatted

COMMERCIAL IN CONFIDENCE

| | | | |
|------|-----------------------|----|---|
| 660 | #1 Coin | 26 | - |
| 661 | 50p Coin | 26 | - |
| 662 | 1p Coin | 26 | - |
| 663 | 2p Coin | 26 | - |
| 2189 | 5p Coin | 26 | - |
| 2190 | 10p Coin | 26 | - |
| 2191 | 20p Coin | 26 | - |
| 2343 | #5 Coin | 26 | - |
| 3320 | Unusable Note | 26 | - |
| 3321 | Unusable Coin | 26 | - |
| 3322 | Unusable Note #100 | 26 | - |
| 3323 | Unusable Note #50 | 26 | - |
| 3324 | Unusable Note #20 | 26 | - |
| 3325 | Unusable Note #10 | 26 | - |
| 3326 | Unusable Note #5 | 26 | - |
| 3327 | Unusable Note #1 | 26 | - |
| 3328 | #2 Coin-Full(Brown) | 26 | - |
| 3329 | #1 Coin-Full(Red) | 26 | - |
| 3330 | 50p Coin-Full(Yellow) | 26 | - |
| 3331 | 20p Coin-Full(Green) | 26 | - |
| 3332 | 10p Coin-Full(Grey) | 26 | - |
| 3333 | 5p Coin-Full(Pink) | 26 | - |
| 3334 | 2p Coin-Full(Blue) | 26 | - |
| 3335 | 1p Coin-Full(Orange) | 26 | - |
| 3336 | #2 Partial Coin Bag | 26 | - |
| 3337 | #1 Partial Coin Bag | 26 | - |
| 3338 | 50p Partial Coin Bag | 26 | - |
| 3339 | 20p Partial Coin Bag | 26 | - |
| 3340 | 10p Partial Coin Bag | 26 | - |
| 3341 | 5p Partial Coin Bag | 26 | - |
| 3342 | 2p Partial Coin Bag | 26 | - |
| 3343 | 1p Partial Coin Bag | 26 | - |
| 3344 | Unusable Coin #5 | 26 | - |
| 3345 | Unusable Coin #2 | 26 | - |
| 3346 | Unusable Coin #1 | 26 | - |
| 3347 | Unusable Coin 50p | 26 | - |
| 3348 | Unusable Coin 20p | 26 | - |
| 3349 | Unusable Coin 10p | 26 | - |
| 3350 | Unusable Coin 5p | 26 | - |
| 3351 | Unusable Coin 2p | 26 | - |
| 3352 | Unusable Coin 1p | 26 | - |
| 1 | Cash | 27 | + |
| 5610 | Cash In Pouches Stock | 27 | + |
| 356 | #100 Banknote | 27 | + |
| 357 | #2 Coin | 27 | + |
| 655 | #50 Banknote | 27 | + |
| 656 | #20 Banknote | 27 | + |
| 657 | #10 Banknote | 27 | + |
| 658 | #5 Banknote | 27 | + |
| 659 | #1 Banknote | 27 | + |
| 660 | #1 Coin | 27 | + |
| 661 | 50p Coin | 27 | + |
| 662 | 1p Coin | 27 | + |
| 663 | 2p Coin | 27 | + |
| 2189 | 5p Coin | 27 | + |
| 2190 | 10p Coin | 27 | + |
| 2191 | 20p Coin | 27 | + |
| 2343 | #5 Coin | 27 | + |
| 3320 | Unusable Note | 27 | + |
| 3321 | Unusable Coin | 27 | + |
| 3322 | Unusable Note #100 | 27 | + |
| 3323 | Unusable Note #50 | 27 | + |
| 3324 | Unusable Note #20 | 27 | + |
| 3325 | Unusable Note #10 | 27 | + |
| 3326 | Unusable Note #5 | 27 | + |
| 3327 | Unusable Note #1 | 27 | + |
| 3328 | #2 Coin-Full(Brown) | 27 | + |
| 3329 | #1 Coin-Full(Red) | 27 | + |
| 3330 | 50p Coin-Full(Yellow) | 27 | + |
| 3331 | 20p Coin-Full(Green) | 27 | + |

COMMERCIAL IN CONFIDENCE

Formatted

| | | | |
|------|-----------------------|----|---|
| 3332 | 10p Coin-Full(Grey) | 27 | + |
| 3333 | 5p Coin-Full(Pink) | 27 | + |
| 3334 | 2p Coin-Full(Blue) | 27 | + |
| 3335 | 1p Coin-Full(Orange) | 27 | + |
| 3336 | #2 Partial Coin Bag | 27 | + |
| 3337 | #1 Partial Coin Bag | 27 | + |
| 3338 | 50p Partial Coin Bag | 27 | + |
| 3339 | 20p Partial Coin Bag | 27 | + |
| 3340 | 10p Partial Coin Bag | 27 | + |
| 3341 | 5p Partial Coin Bag | 27 | + |
| 3342 | 2p Partial Coin Bag | 27 | + |
| 3343 | 1p Partial Coin Bag | 27 | + |
| 3344 | Unusable Coin #5 | 27 | + |
| 3345 | Unusable Coin #2 | 27 | + |
| 3346 | Unusable Coin #1 | 27 | + |
| 3347 | Unusable Coin 50p | 27 | + |
| 3348 | Unusable Coin 20p | 27 | + |
| 3349 | Unusable Coin 10p | 27 | + |
| 3350 | Unusable Coin 5p | 27 | + |
| 3351 | Unusable Coin 2p | 27 | + |
| 3352 | Unusable Coin 1p | 27 | + |
| 5610 | Cash In Pouches Stock | 28 | - |

4.1.3 Item Transaction Mode

Formatted: Bullets and Numbering

This defines the modes in which the Item may be transacted.

| Item Code | Transaction Mode Code | Horizon Mode | Accounting Sense |
|-----------|-----------------------|--------------|------------------|
| CIP | 25 | ROAD | - |
| CHP | 15 | HK | - |

Table 6 – Item Transaction Mode

4.1.44.1.5 Item Transaction Mode Code

Formatted: Bullets and Numbering

A definition of which Items (products), when transacted in certain Modes, map to lines on the Cash Account.

| Cash Account Line Code | Item Id | Transaction Mode Code |
|------------------------|---------|-----------------------|
| 2050 | 1 | 26 |
| 2050 | 1 | 27 |
| 5037 | 5610 | 26 |
| 5037 | 5610 | 27 |
| 5037 | 5610 | 28 |
| 8001 | 5610 | 28 |

New Cash Account Mappings will be as follows:

| Item Code | Transaction Mode Code | Cash Account Code |
|-----------|-----------------------|-------------------|
| CIP | 25 | 5034 |
| CHP | 15 | 5034 |

Table 7 – Item Transaction Mode Code

4.1.54.1.6 Cash Account

Table Line

Formatted: Bullets and Numbering

It is proposed that the unused Cash Account Table Line 8_16_2_X_0_9 is used to present the Cash in Pouches product on the Cash Account. This line maps to Cash Account Code 5034. The description of this line (line name) should be updated to 'Cash in Pouches' (or agreeable similar name).

Formatted

4.1.6 Product Mapping

Formatted: Bullets and Numbering

The mapping of the products to Accounting Nodes is as follows:

| Item Code | Mapping Type | Accounting Node |
|-----------|-----------------|-----------------|
| CIP | Primary Mapping | 740 |
| | | |

Table 8 – Product Mapping

This will lead to a primary mapping on the product of:

4.2 **<PM:<L1:><L2:740><L3:3005><L4:3016><L5:3017>>TYPE B REFERENCE DATA REQUIREMENTS**

Formatted: Bullets and Numbering

4.2.1 Product Mapping

The mapping of the products to Accounting Nodes is as follows:

| Item Code | Mapping Type | Accounting Node |
|-----------|-----------------|-----------------|
| CIP | Primary Mapping | 740 |
| | | |

Table 9 – Product Mapping

This will lead to a primary mapping on the product of:

<PM:<L1:><L2:740><L3:3005><L4:3016><L5:3017>>

4.2.4.3 TYPE C REFERENCE DATA REQUIREMENTS

Formatted: Bullets and Numbering

The following sections define data that will be delivered to the Counters as Type C reference data. It should be noted that this should all be defined as temporal reference data.

4.2.14.3.1 Chart of Accounts Collection

Formatted: Bullets and Numbering

The Chart of Accounts collection is a new collection that defines the Chart of Account Codes and additional parameters that are used during the transaction summarisation process.

<Collection:ChartOfAccounts>
 <ObjectName:CofACode>
 <Data:
 <Description:Description>
 <SummaryMode:SummarisationMode>
 <TxnAttribute:TransactionIdentifier>

COMMERCIAL IN CONFIDENCE

Formatted

<BalanceAccount:BalancingAccount>
<RetainBalance:BalancingIndicator>
<OpeningBal:OpeningIndicator>

Formatted

>
>
Where:

- CofACode* C-of-A Account Code
- SummarisationMode* Method of transaction summarisation required by this Account (Either 'Summary' or 'Individual')
- TransactionIdentifier*: If the Summarisation Mode indicates that the transactions are not to be summarised, then this indicates which piece of attribute grammar is stored as the transaction identifier.
- BalancingAccount*: This will either be 'Y' or 'N' and will define which CofACode will be used to post the Release 1 balancing transaction to. The balancing transaction is simply the sum of all transactions posted on a single accounting day. Only *one* of the CofACode records will have this attribute set to 'Y'.
- BalancingIndicator*: This will either be 'Y' or 'N' and will define whether a daily balance will be held within a local persistent object for this CofACode (see section 3.2.3.5).

OpeningIndicator: Y or N defining whether an opening balance is generated on migration.

Formatted

The full set of data that should be delivered for testing purposes is as follows. The actual Chart of Accounts codes to be used in Live will be advised by POL at a later stage.

| C of A Code | Description | Summary Mode | Txn Attribute | Balance Account | Retain Balance | Opening Balance |
|-----------------|------------------------------------|--------------|---------------|-----------------|----------------|-----------------|
| 000045 51000 | Cash on Hand movement | S | | N | Y | Y |
| 000025 52001 | Cheques on Hand movement | S | | N | N | Y |
| 000035 53001 | Cash Rem In Value | T | Note 1. | N | N | N |
| 000045 55056 | Bureau Card transactions Value | S | | N | N | N |
| 000055 55066 | Non-Bureau Card transactions Value | S | | N | N | N |
| 000065 53040 | Cheque Rem Out Value | S | | N | N | N |
| 000075 53002 | Cash Rem Out Value | T | Note 1. | N | N | N |
| 000085 52100 | Forex on Hand movement(in £) | S | | N | N | Y |
| 553010 | Northern Ireland Cheque Rem-Out | S | | N | N | N |
| 000099 99999 | Balance | S | | Y | N | N |

Table 109 – Chart of Accounts Data

Notes:

Formatted

- The transaction attribute for these transactions will be "EPOSStransaction.BlackBoxData.PouchId"

4.2.24.3.2 Chart of Accounts Products Collection

Formatted: Bullets and Numbering

This is delivered as a new Type C collection that provides the mapping between the product and a Chart of Accounts Code. This will eventually be replaced once the Type A Product data is enhanced with the additional Chart of Accounts Code at some point beyond Release 1.

```
<Collection:CofAProducts>
<ObjectName:ProductCode>
<Data:
  <CofA:Mapping>
```

>

Where:

ProductCode ——— Horizon unique Product Identifier. This is otherwise known as 'Item' within RDT, 'Prod id' within RDMC & RDDS and 'ObjectName' within Riposte Collection EPOSSProducts

Mapping The C-of-A Account to which this Product/Mode combination is mapped

The mappings required at Release 1 ~~are as follows:~~ will be defined by POL.

| Product Code | Product Description | C-of-A Code |
|--------------|--------------------------------------|-------------|
| 1 | Cash | 00001 |
| 2 | Cheque | 00002 |
| 6 | Foreign-Currency-Sterling-Equivalent | 00008 |
| 2568 | Debit-Card | 00004 |
| 4703 | Pre-Pack 50 US Notes | 00008 |
| 4704 | Pre-Pack 150 US Notes | 00008 |
| 4705 | Pre-Pack 350 US Notes | 00008 |
| 4709 | Pre-Pack 50 EU Notes | 00008 |
| 4710 | Pre-Pack 150 EU Notes | 00008 |
| 4711 | Pre-Pack 350 EU Notes | 00008 |
| 4816 | FRTS-Customer-Collection | 00004 |
| 4818 | BdeC-Pre-Order-Buy-back | 00004 |
| 5045 | Debit-Card | 00005 |
| 5046 | Debit-Card | 00005 |
| 5047 | Debit-Card | 00005 |
| 5048 | Debit-Card | 00005 |
| 5049 | Debit-Card | 00005 |
| 5050 | Debit-Card | 00005 |
| 5051 | Debit-Card | 00005 |
| 5052 | Debit-Card | 00005 |
| 11205 | Rem-Out-Cheques-to-DC | 00006 |

COMMERCIAL IN CONFIDENCE

Formatted

| New Products Required | | |
|-----------------------|-----------------------|-------|
| | | |
| GIP | Cash-in-Pouches | 00001 |
| CTI | Cash-in-Transit (In) | 00003 |
| CTO | Cash-in-Transit (Out) | 00007 |

Table 10 – Chart of Accounts Products Collection

4.2.34.3.3 Settlement Product Modes Collection

Formatted: Bullets and Numbering

This collection defines which settlement product should be used for each Product/Mode combination. An entry for this collection is only required when the settlement product for this Product/Mode combination differs from that which is defined in the ModeParameters Collection.

```
<Collection:CofAProductModes>
<ObjectName:ProductCode>
<Data:
  <Modes:
    <Mode:
      <M:Mode>
      <V:Version>
      <S:SettlementProduct>
    >
    <Mode:
      .....
      .....
    >
  >
>
```

Where:

ProductCode ———Horizon unique Product Identifier. This is otherwise known as ‘Item’ within RDT, ‘Prod id’ within RDMC & RDDS and ‘ObjectName’ within Riposte Collection EPOSSProducts.

Mode Transaction Mode allowed for the Product

Version Version of the Product/Mode

SettlementProduct The product code to which this Product/Mode combination Settles. This is optional and only present if the Product/Mode combination does not Settle to the default Product as specified in the ‘ModeParameters’ Collection.

| Product Code | Product Description | Mode | Settlement Product | Settlement Product Description |
|--------------|---------------------|------|--------------------|--------------------------------|
| 1 | Cash | RIAD | 11218 | Cash in Transit (In) |
| 2 | Cheques | ROAD | 11205 | Rem Out Cheques DC |

Table 12.11 – Settlement Product Modes Collection

Formatted

Following Release 1, it is expected that this data will be made available in the ProductModes Collection via Type-A reference data (Item_Transaction_Mode).

4.3.4 Mode Parameters

Formatted: Bullets and Numbering

The reference data Collection 'ModeParameters' defines how the modes are used within the context of the Counter desktop. A new instance of the collection is required for each new Transaction Mode.

4.3.4.1 Mode ROSP (Mode 26)

Formatted: Bullets and Numbering

This mode will be set-up in a similar manner to the existing ROAD Mode (Mode 25) except that the settlement product will be different and there is no secondary mapping.

```
<Collection:ModeParameters>
<ObjectName:ROSP>
<ModeInfo:
  <Cmd:ChangeMode>
  <DASS:True>
  <MaxStackTotal:9999999.99>
  <Mode:ROSP>
  <MC:True>
  <LINVZero:True>
  <SettlementProduct:5610>
  <PostSettleTxn:True>
  <ShowNoRed:True>
  <SessionReceipt:108>
  <AlwaysPrintReceipt:True>
  <ReceiptTitle:Remittance Out Slip (Auto Distribution)>
  <CallApp:
    <InterfaceName:
      <LFSCconfirm:
        <CmdStr:
          <Cmd:ReadBarcode>
          <Type:LFSCollection>
        >
      >
    >
  >
  <ReceiptHotKey:Disabled>
  <ModeTitle:Rem Out to Pouches>
  <ReverseSense:True>
  <PermanentSense:Out>
  <PrimaryMappings:>
  <SecondaryMappings:>
>
```

4.3.4.2 Mode RISP (Mode 27)

Formatted: Bullets and Numbering

Fujitsu Services

End to End Release 1 - High Level Design

Ref.: EA/HLD/002

Version: 1.00-3

Date: 19/01/2004

COMMERCIAL IN CONFIDENCE

Formatted

This mode will be set-up in a similar manner to the existing RIAD Mode (Mode 24) except that the settlement product will be different and there is no secondary mapping. Also, the receipt will be similar in format to the existing Remit-Out receipt.

```

<Collection:ModeParameters>
<ObjectName:RISP>
<ModeInfo:
  <Item:>
  <Cmd:ChangeMode>
  <DASS:True>
  <MaxStackTotal:9999999.99>
  <Mode:RISP>
  <LINVZero:True>
  <MC:True>
  <SessionReceipt:108>
  <SettlementProduct:5610>
  <AlwaysPrintReceipt:True>
  <ReceiptTitle:Reverse Remittance-Out Slip>
  <CallApp:>
  <ReceiptHotKey:Disabled>
  <ModeTitle:Reverse RO SP>
  <ReverseSense:True>
  <PermanentSense:ln>
  <PrimaryMappings:>
  <SecondaryMappings:>
>

```

4.3.4.3 Mode RODP (Mode 28)

Formatted: Bullets and Numbering

This new mode 'Remit Out Despatch Pouches' will be set-up in a similar manner to the existing ROAD Mode (Mode 25) in terms of the accounting requirements. However, the difference between the RODP and ROAD will be that the RODP mode will not cause the LFS Pouch packing functionality to be invoked and will not cause an EPOSS receipt to be printed.

```

Collection:ModeParameters>
<ObjectName:RODP>
<ModeInfo:
  <Cmd:ChangeMode>
  <DASS:True>
  <MaxStackTotal:9999999.99>
  <Mode:RODP>
  <MC:True>
  <LINVZero:True>
  <SettlementProduct:CIT> (Horizon 'Cash in Transit (Out)' product)
  <PostSettleTxn:True>
  <ShowNoRed:True>
  <SessionReceipt:>
  <AlwaysPrintReceipt:>
  <ReceiptTitle:>
  <CallApp:>
>
  <ReceiptHotKey:Disabled>
  <ModeTitle:Rem Out Desp Pouch>

```

COMMERCIAL IN CONFIDENCE

Formatted

```
<ReverseSense:False>
<PermanentSense:Out>
<PrimaryMappings:>
<SecondaryMappings:
  <L1:>
  <L2:3048>
  <L3:3029>
  <L4:3027>
  <L5:3017>
>
```

4.2.44.3.5 EPOSSProducts Collection

Formatted: Bullets and Numbering

Products in the range 10000 to 19999 are not used by POL and may therefore be used by Fujitsu for the delivery of Type C Product Reference Data. ~~Four~~ Two new products will be defined by the development Team. These will take product numbers in the range 112.xx that will be allocated closer to the time of code delivery and test.

Cash in Transit (Return Cash to ADC)

This has also been named 'Cash Transfer Out' in some other documents.

Formatted

Formatted

```
<Collection:EPOSSProducts>
<ObjectName:CTO>
<Data:
  <PN:CTO>
  <SN:CshOut>
  <LN:Cash In Transit (Out)>
  <RN:CshOut>
  <MnV:0>
  <MxV:1000000>
  <MnQ:1>
  <MxQ:1>
  <FP:False>
  <RF:True>
  <RP:>
  <MV:0>
  <AS:True>
  <SE:In>
  <SR:False>
  <RcptNo:0>
  <MP:>
  <V:1>
  <RV:False>
  <S:>
  <VO:True>
  <I:False>
  <SDI:False>
  <CDI:False>
  <LINVI:False>
  <LACC:False>
  <ACCI:>
  <PM:
    <L1:>
    <L2:922>
    <L3:903>
    <L4:901>
    <L5:0>
  >
>
```

Cash in Transit (Receive Cash from ADC)

Formatted

This has also been named 'Cash Transfer In' in some other documents

Cheques in Transit (Return Cheques to ADC)

This is for Northern Ireland cheques that are despatched in ROAD Mode

```
<Collection:EPOSSProducts>
<ObjectName:CTI>
<Data:
  <PN:CTI>
  <SN:CshIn>
  <LN:Cash In Transit (In)>
  <RN:CshIn>
  <MnV:0>
  <MxV:1000000>
  <MnQ:1>
  <MxQ:1>
  <FP:False>
  <RF:True>
  <RP:>
  <MV:0>
  <AS:True>
  <SE:In>
  <SR:False>
  <ReptNo:0>
  <MP:>
  <V:1>
  <RV:False>
  <S:>
  <VO:True>
  <I:False>
  <SDI:False>
  <CDI:False>
  <LINVI:False>
  <LACCI:False>
  <ACCI:>
  <PM:
    <L1:>
    <L2:921>
    <L3:903>
    <L4:901>
    <L5:0>
  >
>
```

4.2.54.3.6 Buttons

New menu buttons are described in [DIALOGUE]. These are also defined in SD/DES/005.

Note: New buttons and the menu hierarchy cannot be defined until user dialogues have been agreed

Formatted: Bullets and Numbering

4.2.6 DeskTop Apps

There is possibly a requirement for new DeskTop Apps objects to support invocation of new LFS DLLs. The requirement for this will depend on whether the low level strategy for migration is to update the existing DLLs with new functionality or to replace the existing DLLs with new ones. The latter is probably the simpler migration

Formatted: Bullets and Numbering

Formatted

~~mechanism but will require the new DLLs to be described in the DeskTopApps Collection.~~

4.2.74.3.7 Token Definitions

Formatted: Bullets and Numbering

Two token definitions exist for the LFS counter application although both definitions are identical and relate to the same barcode. These definitions will not change at E2E Release 1.

A new token is required to define the LFS ACC Barcode. This will be defined in [LFSHLD]. The collection name is LFS PouchInputEvents.

4.3.7.1 AppConfig Collection

Formatted: Bullets and Numbering

The AppConfig collection is used by the SoftLaunch function to determine what affect the presence of the SoftLaunch Migration product will have on the menu hierarchy (which buttons will be enable/visible and which will be disabled/invisible).

Two soft-launch enabling non-core products (SOF1 & SOF2 defined in 4.1.1) will be used to dictate the date which branches migrate. Two AppConfig objects are therefore needed to control how these soft-launch products operate.

4.3.7.2 CofABSP Collection

Formatted: Bullets and Numbering

A new collection is required that will contain search criteria and application parameters for the new EOD Branch Summarisation process that is described in section 3.2.3. Two new objects will be defined; Parameters and RetrievalData.

4.3.7.3 ButtonList Collection

Formatted: Bullets and Numbering

This is a new Collection that will contain data that identifies buttons that are to be disabled dependent on where within the menu hierarchy transactions are being performed. Please refer to EP/LLD/012 for further details.

4.3.7.4 MessageDefs Collection

Formatted: Bullets and Numbering

New messages will be defined as a result to application changes. Many of these new messages are defined in [DIALOGUE]. However, this should not be treated as an exhaustive list since new messages and prompts come to light during the Low Level design and Development phases.

4.3.7.5 SchedulerTimeEvents Collection

Formatted: Bullets and Numbering

The new daily summarisation process needs to be scheduled to be run after the Daily Reconciliation process but before the LFS EOD process. This requires a new object in the SchedulerTimeEvents collection

4.3.7.6 SchedulerBusinessEvents Collection

Formatted: Bullets and Numbering

In order to make the LFS EOD process dependent on the completion of the new Summarisation process, a new object requires creation in the SchedulerBusinessEvents Collection.

4.34.4 OTHER REFERENCE DATA REQUIREMENTS

4.3.14.4.1 Type D Reference Data

The following Type-D reference data requires delivery to all counters.

4.4.1.1 DeskTopApps

There is possibly a requirement for new DeskTopApps objects to support invocation of new LFS DLLs. The requirement for this will depend on whether the low level strategy for migration is to update the existing DLLs with new functionality or to replace the existing DLLs with new ones. The latter is probably the simpler migration mechanism but will require the new DLLs to be described in the DeskTopApps Collection.

Note: At the time of writing, there has been no requirement for new DeskTopApps objects identified.

4.4.2 RDDS Meta Data

4.4.2.1 SVC Control Products

The Product Ids of both the SOF1 and SOF2 soft-launch products need to be registered in the RDDS table 'svc_control_products'.

4.4.2.2 POCL Trans Mode Conversions

Three new transaction modes will be introduced to support the new remittance functionality that is described in [LFSHLD]. These will require new rows in the RDDS table that converts the POL Mode number to the internal Horizon mnemonic code. The table is 'pocl_trans_mode_conversions' and the data required is as shown in the following SQL script:

```
insert into POCL_TRANS_MODE_CONVERSIONS values (26,'ROSP',0,SYSDATE);
insert into POCL_TRANS_MODE_CONVERSIONS values (27,'RISP',0,SYSDATE);
insert into POCL_TRANS_MODE_CONVERSIONS values (28,'RODP',0,SYSDATE);
```

4.4.2.3 POCL Trans Mode Exclusions

The new transaction modes need to be recorded in the POCL_trans_mode_exclusions table so that the Cash Account Table 5 node information is withheld from the counter application. The script needed to populate the data is as follows:

```
insert into POCL_TRANS_MODE_EXCLUSIONS values (15,26,'N',5,'X',SYSDATE);
insert into POCL_TRANS_MODE_EXCLUSIONS values (15,27,'N',5,'X',SYSDATE);
insert into POCL_TRANS_MODE_EXCLUSIONS values (15,28,'N',5,'X',SYSDATE);
```

4.4.2.4 Stage Modes

In order to allow graceful termination of non-core products, the table stage_modes requires additional data for the three new transaction modes.

```
insert into stage_modes values(0,'ROSP');
insert into stage_modes values(0,'RISP');
insert into stage_modes values(0,'RODP');
insert into stage_modes values(1,'ROSP');
```

Formatted

```
insert into stage modes values(1,'RISP');
insert into stage modes values(1,'RODP');
insert into stage modes values(2,'ROSP');
insert into stage modes values(2,'RISP');
insert into stage modes values(2,'RODP');
```

4.4.3 Global Objects

Formatted: Bullets and Numbering

New receipt definitions for Remittance-in and Remittance-out will require that the GlobalObjects.dat file is redistributed to all Branches.

4.3.1.1 AppConfig Collection

Formatted: Bullets and Numbering

The AppConfig collection is used by the SoftLaunch function to determine what affect the presence of the SoftLaunch Migration product will have on the menu hierarchy (which buttons will be enable/visible and which will be disabled/invisible).

```
<Collection:AppConfig>
<ObjectName:LFSQLClass>
<Data:
  <Enable_E2E_R1:
    <PN:SoftLaunchProduct>
    <1:
      <SysAttr:
        <Enable_E2E_R1:True>
      >
      <Item01194~MenuButtonDef.
        <Invisible:TrueFalse>
      >
      <Item01194~.....
    >
    <ERR:
      <SysAttr:
        <Enable_E2E_R1:False>
      >
      <Item01194~MenuButtonDef.
        <Invisible:TrueFalse>
      >
      <Item01194~.....
    >
  >
>
```

Where:

LFSQLClass: The name of a DLL where the End-to-End Release 1 SoftLaunch Query functionality is implemented (See [SFTLNCH]). It is expected that this will have the value of LFSCONFIRM.DLL (to be confirmed)

Enable_E2E_R1: This value will be passed to the End-to-End Release 1 SoftLaunch Query DLL but will not be used.

SoftLaunchProduct: This is the product number of the softlaunch product (This is product SOF2 described in section 4.1.1)

MenuButtonDef: Many menu buttons will be defined in this section. The actual number and content cannot be determined until the user interface and menu hierarchy have been agreed.

Formatted

~~TrueFalse: Indicating whether the menu item should be displayed or hidden.~~

4.44.5 OTHER REFERENCE DATA ISSUES

Formatted: Bullets and Numbering

Various ~~sett~~insections within this document highlight that the Chart of Account information and settlement information is directly related to the existing Riposte collections EPOSSProducts and ProductModes. It is suggested in this document that the additional data that is necessitated by Release 1 is merged with the existing reference data structures at future releases. However, doing so, would necessitate a massive redelivery of type A reference data. It may therefore be preferable to keep the new CofA reference data physically separate from the existing ~~Prouet~~Product and ProductMode data even though the two entities are logically the same. This will be considered further at the next release.

Chapter 5 - System Quality Attributes

5.1.1 Security

This document recommends changes to a product that already exists and introduces no new components. The existing security measures are therefore adequate as long as the development and delivery of the changed products adheres to the security standards that are already in-force.

5.1.2 Scalability

The end-of-day summarisation process, the harvesting and the TPS transformation processes are all based on well-tried and tested existing functionality. The new summarisation process is not expected add any significant additional volumes (in percentage terms) to the existing volume of transactions harvested through the correspondence layers and TPS.

The EoD Summarisation process will perform an additional scan of the Message Store (from the previous EoD Marker). This is expected to take approximately 2 minutes and will therefore add this time to the overall EoD process.

The initial scan following Migration will possibly take longer than 2 minutes depending on how many days have passed since the last Cash Account Rollover. However, this is a one-off activity.

5.1.3 Resilience

The changes described in this document either modify existing processes or implement new processes that are cloned from existing and proven technologies and techniques. The resilience of the system is therefore not impacted by the implementation of these changes and there is no requirement or need to increase the resilience.

Chapter 6 - Specifically Excluded

The following systems, sub-systems or processes are specifically excluded from change or impact at End to End Release 1 due to inadequately defined or lack of requirement.

6.1 SERVICE LEVEL MEASUREMENT

It is assumed that there is no requirement to report on the timeliness of delivery of information to POL FS within projects 1 and 3. This will be considered as part of project 2.

6.2 BRANCH REPORTING REQUIREMENTS

POL have been offered the opportunity to have the following view and report capability:

- View/Report Replenishment Delivery Notices (Received & Pending)
- View/Report Packed Cash Pouches (Pending and Collected)

No reply has been received from POL and these functions are therefore not considered to be within the scope of End-to-End Release 1.

Chapter 7 - Migration

7.1 OVERVIEW

There are three external interfaces that will require change at Release1.

SAPADS: New Inbound Replenishment Delivery Notices File
Modified Outbound Cash Statement File
POL FS: New Financial Summary File

As usual, the data centres will be migrated first followed by the Counters/Branches. Data centre migration can be performed as two separate activities; LFS & TPS:

LFS: LFS Host, LFS FTMS, LFS Harvester and LFS Replenishment Delivery Notice Loader.

TPS TPS Host, TPS Harvester (plus interface to POL FS)

Immediately following Migration of the Data Centres and before migration of the Counters:

New Inbound Replenishment Delivery Notices Files will be processed fully and transferred to Branches. This data is benign and will be ignored by counter software.

The new attribute 'Generated Cash Balance' will not be available to the migrated LFS Cash Statement harvester. This agent will therefore not populate the LFS Host with this data and the LFS Host will pass this NULL value back to SAPADS. This new attribute must therefore be optional at all times.

The migrated TPS Harvester will not find any Financial Summaries and will therefore not populate the new TPS Chart of Accounts Tables. TPS will therefore generate no transactions for POL FS and consequently will generate no data transmission files. An End of Transmission file will still be generated indicating that there are no files to process. The POL FS system must accept these files or a mechanism must be in place to automatically assume acknowledgement of receipt of these files.

The counter software will be delivered and then invoked separately using 'Soft-launch' products. This will be considered in more detail below.

7.2 FINANCIAL DATA TO POL FS

The financial summaries are generated by the process described in Section 3.2.3. This process will be invoked via changes to the Counter Scheduler reference data and will therefore execute each night following delivery of new Release 1 executables and the

associated reference data. Before performing any processing, the presence of the Soft-launch product will be checked and, if present, the summarisation process will continue. If the Soft-launch product is not available, then the summarisation process will remove the local persistent objects that are described in Section 3.2.3.5 and then exit. This will enable the Summarisation process to be switched on and off by the Soft-launch product. The absence of the Soft-launch Product removes the Persistent objects and the absence of the persistent objects will cause the Summarisation process to generate opening balances on initial execution following Soft-launch product availability.

The desktop Softlaunch application therefore does not need to be aware of this Softlaunch product nor does it need to perform any new migration processing.

This functionality cannot be softlaunched before the new remittance functionality described in 7.4

7.3 GENERATED CASH BALANCE TO SAP ADS

This functionality will be invoked immediately following delivery of the software. The Generated Cash Balance attribute will be derived from the values in the local Persistent objects that are created by the POL Financial Summarisation process. If the Persistent Objects do not exist, then the process will simply not create the new attribute within the Cash Statement message. The absence of the Generated Cash Balance attribute will not be an issue to the LFS Harvester since the attribute is optional. This process does not therefore need to consider any soft-launch product.

7.4 REMITTANCE IN/OUT

A single new Soft-launch non-core product will implement the revised processes for delivery and collection of cash pouches. In order to fully understand this section of the document, the reader should be familiar with the new requirements for Pouch Collection/Delivery and Remittance Out/In as described in [LFSHLD].

The existing SoftLaunch desktop start-up function will test the presence of the SoftLaunch product and also test the availability of the Release 1 code. If both are available to the Desktop then a new SoftLaunch Property will be set to TRUE to indicate that the Branch/Counter is now fully migrated. SoftLaunch will use the new Reference Data defined in section 0.1.1.14.3.1.1. This function must be softlaunched before the softlaunch of the process described in 7.2.

The new LFS DLLs (or modified LFS DLLs) will test the value of the SoftLaunch Property to determine the migration status. Low Level design will determine whether completely new DLLs are required or whether the existing DLLs will be updated.

The following exceptional conditions may be present at the point of migration that require consideration.

7.4.1 Remittance-Out

At the point of migration, there may exist a situation whereby Cash Pouches have been packed but not yet despatched. In these instances, the cash will have been

remitted-out of the office since the ROAD transaction that was performed at the point of packing the Pouch will have been settled to the normal ROAD settlement Product (11216). Since the ROAD Settlement Product is not mapped to the normal root node of the Cash Account Hierarchy, then the liability of the Branch is reduced.

Following migration, the packing of a Cash Pouch would settle the ROAD transaction to a new Cash-in-Pouches product that is mapped to table 2 of the Cash Account thereby retaining the value of the pouch as a suspense item.

Following migration, there may well be both types of Cash Pouch that have been prepared and are ready for collection.

The new collection process requires that the Cash Pouches are placed into a Collection Group. The fact that the Group may contain Cash Pouches that were prepared both before and after migration will not cause any problems since the structure of the messages in the message store does not differ (it is only the settlement product that differs).

At the point of collection, the system should also not functionally distinguish between Cash Pouches that were packed before migration and Cash Pouches that were packed post-migration. However, the net affect of collecting the Cash Pouches will be the same as shown below:

7.4.1.1 Pre-Migration Cash Pouch

At the point of packing the Cash Pouch containing £100, the following net transactions are made:

| Product | Value |
|---|--------|
| Product 1 (cash): | - £100 |
| Product 11216 (ROAD Settlement Product) | + £100 |

Since Product 11216 is not mapped to the Cash Account, then the liability of the Branch is reduced.

Note: It is assumed here that the EPOSS functionality described in section 3.2.2 migrates at the same time as the LFS Remittance functionality.

At the point of Pouch Collection, a separate transaction is performed against the Settlement Product that was used in the original transaction (in this case, Product 11216). This new transaction will be settled to the new Cash ROAD Settlement Product (See section 4.3.54.2.4):

| Product | Value |
|--|--------|
| Product 11216 (ROAD Settlement Product) | - £100 |
| Product CTO (Cash ROAD Settlement Product) | + £100 |

Since both the Original ROAD Settlement Product and the Cash ROAD Settlement Product are not mapped to the Cash Account, the transaction has no affect on the branch balance. However, this transaction *will* be available for harvesting to the new POL FS as a Cash In Transit (Out) transaction.

To ensure that this transaction can legally be performed, and that it follows all of the generic EPOSS transaction rules, Reference Data needs to be available to allow Product 11216 to be transacted in Housekeeping mode. This is done by providing a Product Mode mapping to the Counter in the CofAProductModes Collection (See section [4.3.34.2.3](#)).

Note: It must be ensured that the presence of CofAProductModes for a Product/Mode combination causes EPOSS to accept that Product/Mode combination as a valid transaction.

7.4.1.2 Post-Migration Cash Pouch

At the point of packing the Cash Pouch containing £100, the following net transactions are made:

| Product | Value |
|-------------------------------|--------|
| Product I (cash): | - £100 |
| Product CIP (Cash In Pouches) | + £100 |

Since the 'Cash in Pouches' Product is mapped to Cash Account table 2, then the liability of the Branch is retained and the value of the Pouch will be shown as a suspense item.

At the point of Pouch Collection, a separate transaction is performed against the Settlement Product that was used in the original transaction (in this case, Product 'Cash in Pouches'). This new transaction will be settled to the new *Cash* ROAD Settlement Product (See section [4.3.54.2.4](#)):

| Product | Value |
|--|--------|
| Product CIP (Cash In Pouches) | - £100 |
| Product CTO (Cash ROAD Settlement Product) | + £100 |

Since the Cash ROAD Settlement Product is not mapped to the Cash Account, the transaction has the affect of reducing the branch balance. The transaction will be available for harvesting to the new POL FS as a Cash In Transit (Out) transaction.

7.4.1.3 Summary

The net affect of performing Collection of Cash Pouches that were packed either before or after migration is the same.

- The value of Cash is reduced
- The liability of the Branch is reduced
- The value of the Pouch is settled to the Cash ROAD Settlement product and reported to POL FS as Cash in Transit.

There are no other migration considerations.

7.4.2 Remittance-In

At the point of migration, there may exist a situation whereby Cash Pouches have been received into the Branch but not yet remitted-in. In these instances, the cash cannot be remitted-in using the original mechanisms since the manual Remittance-in function will be disabled following migration.

A mechanism must therefore be provided to remit-in the cash manually.

Attempting to process the Cash Pouch through the Pouch Delivery function a second time will perform this. The Pouch Delivery function will recognise the attempt to process a duplicate Pouch Id from the presence of the Pouch Id in an existing *PouchDelivery* message. If this message does not contain the attribute *Data.Manual* then the original delivery was pre-migration. Processing will then continue as normal (either allowing manual entry of the Cash Pouch value or retrieving the value from an existing Replenishment Delivery Notice) until final transaction. The final transaction will then commit the Remittance In of the Pouch value but will refrain from writing a duplicate Pouch Delivery message.

Full details of the process are described in [LFSHLD].

7.5 EPOSS

Changes to EPOSS settlement functionality are described in section 3.2.1.1. ~~There is no reason why these changes cannot be implemented immediately the code changes are delivered to the counter. The functional changes are invoked by the delivery of the new Collections CofAProductModes and CofAProducts~~The EPOSS multiple settlement function must not be invoked until the point of softlaunch of the functions described in 7.4. The reason for this is that the new EPOSS settlement function will begin transferring cash to a new Cash-in-Pouches suspense product before there is the functionality available to subsequently move this value to the Cash-in-transit product. The launch of the EPOSS functionality must therefore coincide with the launch of the Remittance functionality via product SOF2.

It should be noted that the softlaunch of the POL FS feed can only be done once it is guaranteed that all cash & near cash transactions between the last Cash Account Rollover and the current date have the CofA mapping written as part of the transaction content. It is advised therefore that the softlaunch of the POL FS feed is not performed until after a week of code and reference data availability at the counters.