1 Introduction

I am Gareth Idris Jenkins. I am employed by Fujitsu Services Ltd who have been contracted by Post Office Ltd to provide the Horizon systems operating in Post Offices around the country. However I understand that my role is to assist the court rather than represent the views of my employers or Post Office Ltd.

I graduated from Cambridge University with a degree in Mathematics in 1973 and was awarded an MA by Cambridge University in 1997. I was employed by ICL in September 1973 and have worked for that company ever since (though its name was changed to Fujitsu Services about 10 years ago). During my time with ICL / Fujitsu I have held a number of roles in customer support, development, design and architecture. During the early 1990s I was involved with representing ICL in developing Systems Management Standards and in 1992 I was the head of the UK delegation on Systems Management at the International Standards Organisation conference in Ottawa, Canada. In the late 1990s I become a Distinguished Engineer within ICL. Distinguished Engineers, were about 100 or so of the senior technical staff within the company (out of about 6000 to 7000 technical staff).

I am a member of the British Computer Society (MBCS), a Chartered Engineer (CEng) and a Chartered IT Professional (CITP).

Since 1996 I have been working on the Horizon project in association with Post Office Ltd. My initial role was in the integration of the Riposte messaging system which is responsible for storing all data in the Post Office branches and replicating it to the Data Centres. I was also responsible for the design of the interface between Horizon and Streamline which processes all Credit and Debit Card payments for Post Office Ltd. More recently I've been involved in projects associated with interfacing data from Horizon to Post Office's back end accounting systems.

The purpose of this report is to provide some further background information and relate this to the current case.

1.1 Document Structure

Section 2 of the document describes the Horizon system at a high level, giving a time-line for its development, the Business scope and Architecture diagrams for both the original Horizon System and the current Horizon Online system.

Section 3 then summarises my views on the overall integrity of the Horizon system.

2 The Horizon System

2.1 Timeline

Fujitsu were originally awarded a contract in 1996 to provide a Horizon System to Post Office Ltd. The following provides some key dates and functional changes:

- Horizon Pilot 1996
- Horizon Rollout 1999 – 2002
- Network Banking 2003
- EMV 2004
- Cash Account removed 2005

- Data Centre Migration 2009
- HNG-X Rollout 2010

Horizon Online (or HNG-X) was a major re-implementation of Horizon. It was a complete re-implementation of the business functionality at the counter and utilised a central Database to hold details of all transactions rather than the MessageStore used by the original Horizon system.

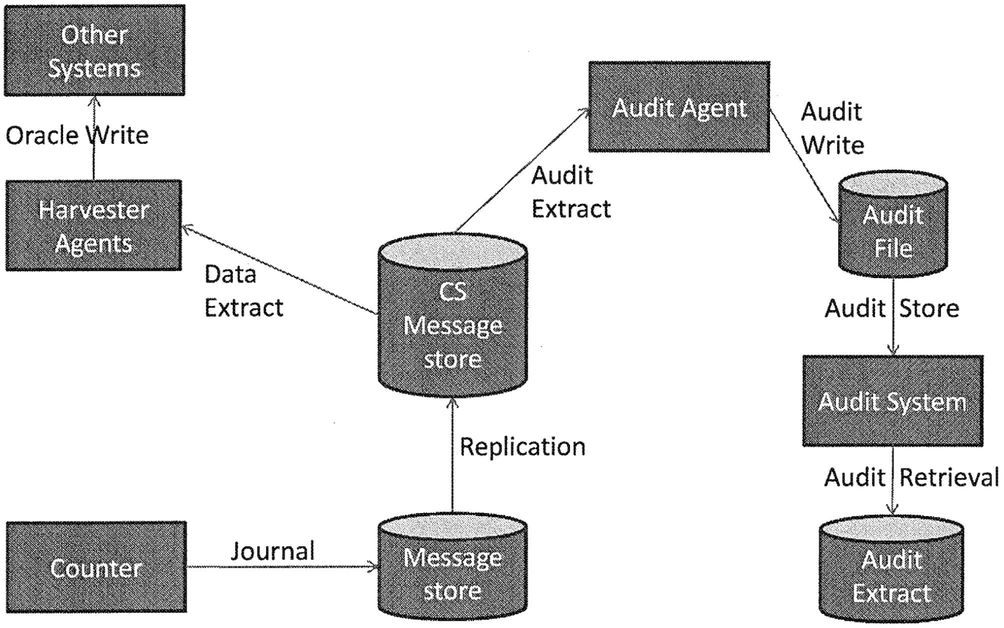
All Post Office Branches migrated from the original Horizon to Horizon Online between January and September 2010. Historical transactions were made visible in the new system as part of the migration process.

2.2 Business Scope

The Business scope of Horizon is:

- Point Of Sale Application
- Transaction Recording
    - All such transactions are Audited
- Posting Summary Transactions to POL SAP (Post Office Ltd's back end accounting system)
- Posting Detailed Transactions to Credence (Post Office Ltd's back end Management Information system)
- Posting Remuneration Data to HR-SAP (Royal Mail Group's back end Payroll system)
- Delivering Client Data to Post Office Ltd's Clients (ie 3rd parties that Post Office Ltd acts as an agent for such as Local Authorities and Utility companies etc)

2.3 Architecture Diagrams



1

Figure 1 – Horizon Data Flows

The Horizon system was designed to store all data locally on the counter's hard disk in what is referred to as the messagestore. Once the data has been successfully stored there it is then replicated (copied) to the hard disks of any other counters in the branch (and in the case of a single counter branch to the additional external storage on the single counter). Data is also passed on from the gateway counter to the Horizon data centre using similar mechanisms where it is stored in the CS Messagestore.

The replication process is designed such that should the data fail to be copied immediately (for example due to a failure on the local IT network within the branch or another counter being switched off or the branch being disconnected from the data centre), then further attempts are made to replicate the data at regular intervals until it is finally copied successfully. Once the data reaches the Data Centre a further copy is taken by the Audit Agent which writes it to an Audit File which is added into the audit trail where it is available for retrieval for up to 7 years. Data in the audit trail is "sealed" with a secure checksum that is held separately to ensure that it has not been tampered with or corrupted.

Other systems can also access the data from the CS Messagestore via Harvester Agents. However such systems are outside the scope of the integrity of the Audit trail.

Every record that is written to the transaction log has a unique incrementing sequence number. This means it is possible to detect if any transitions records have been lost.

While a customer session is in progress, details of the transactions for that customer session are normally held in the computer's memory until the customer session (often known as the "stack") is settled. At that point all details of the transactions (including any methods of payment used) are written to the local hard disk and replicated (as described above). It should be noted that double entry bookkeeping is used when recording all financial transactions, ie for every sale of goods or services, there is a corresponding entry to cover the method of payment that has been used. When a "stack" is secured it is written in such a way that either all the data is written to the local hard disk or none of it is written. This concept of "atomic writes" is also taken into account when data is replicated to other systems (ie other counters, external storage or the data centre).

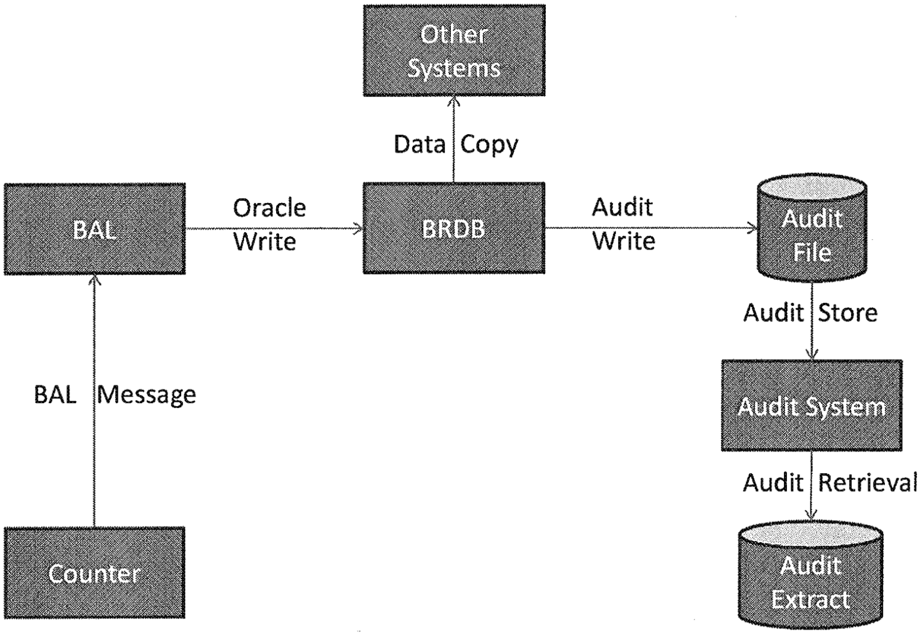
The data for a stack will have been successfully secured to the local hard disk before the screen is updated indicating that a new customer session can be started. Note that although an attempt will have been made to replicate the data to an external system at this time, there is no guarantee at this point that such replication will have been successful. For example if there is a Network Failure followed by a Terminal failure there is a slight risk that transactions in the intervening period could be lost.

All data that is written includes a "checksum" value (known as a CRC) which is checked whenever the data is read to ensure that it has not been corrupted. Any such corruptions detected on reading will result in failures being recorded in the event logs which are held on the local hard disk for a few days for immediate diagnosis and also immediately sent through to the data centre where they are held for 7 years.

Any failures to write to a hard disk (after appropriate retries) will result in the counter failing and needing to be restarted and so will be immediately visible to the user.

Whenever data is retrieved for audit enquiries a number of checks are carried out:

1. The audit files have not been tampered with (ie the Seals on the audit files are correct)
2. The individual transactions have their CRCs checked to ensure that they have not been corrupted.
3. A check is made that no records are missing. Each record generated by a counter has an incremental sequence number and a check is made that there are no gaps in the sequencing.



1

Figure 2 – Horizon Online Data Flows

Horizon Online is designed to store all data in an online database known as the Branch Database (BRDB). In particular no data concerning Business Transactions is retained at the counter other than in the memory of the Counter Business Application.[1]

Transactions are carried out locally on the Horizon Online counters and a Basket is built up during a Customer Session. Each transaction will result in a Basket Entry consisting of one or more Accounting Lines. At the end of a Customer Session when the Basket has been completed and all Settlement items (or Tender lines) have been processed and added into the Basket as further Accounting Lines, such that the total value of the Basket is zero, the entire Basket is sent to the Data Centre as a BAL Message where the Branch Access Layer (BAL) processes the message and all the Accounting Lines are recorded and committed to the BRDB as part of a single Oracle Commit. This means that either **all** the transactions within a Basket are successfully written or **none** of them are. Once the Accounting Lines have been successfully committed a response is returned to the counter indicating this success and this then allows any receipts to be printed. The Basket is deemed to be fully completed once all relevant receipts have been successfully printed. Note that if there are no receipts to be printed, then the screen is updated to show the top level menu indicating successful completion of the previous Basket.

[1] In order to support recovery, the identifier of the last successfully completed Basket is recorded on the Hard disk at the counter. However this is not classed as Business Data.

The Oracle Commit also includes an Audit of the data originally transmitted from the counter to the BRDB. This data is digitally signed at the counter using a key generated as part of the Log On process. It is this audit record that is used to provide the extract of transactions used for Litigation support.

Any auditable message from the counter is stored, together with its Digital Signature and other key attributes in an "Audit table" (known as the Message Journal) in BRDB. Each night after midnight, the contents of this table for the previous day are copied from the BRDB to a number of serial files.

> *A number of files are generated due to the volume of data processed each day. All data from a given Branch will be concentrated into a small number of these files for ease of retrieval.*

At this point a check is made that indeed there are no missing or duplicate jsns for any counter and should any be found an alert is raised.

> *Note that this could only happen as a result of a bug in the code or by somebody tampering with the data in BRDB and this check is included specifically to check for any such bugs / tampering.*

These files are then copied to the Audit system where they are sealed with digital seals. They are held there for a period of 7 years during which time they may be retrieved and filtered to produce the relevant audit data for a particular Branch.

The audit record may also include application events that have been accumulated at the counter since the last auditable message was sent to the Data Centre. All major activities that affect the Branch also have an audit of the data sent from the counter to the Data Centre included in the audit log.

Each Audit record includes the following identification:

- Branch identifier (i.e. FAD Code)
- Counter identifier
- Sequence Number (known as a Journal Sequence Number or jsn)
- Counter timestamp

Within any counter (i.e. for a given Branch Id / Counter Id combination), the jsn will always increase by exactly one for each successive audit record. This enables a check to be made that there are no records missing from the audit trail when they are retrieved.

The transactions in a basket are constructed using the principle of double-entry bookkeeping. This means that in addition to the Accounting Lines that relate to the actual business transactions, separate Accounting Lines are also generated for the tender items (such as Cash, Cheques or Credit / Debit Cards), resulting in the total value of all Accounting Lines in a Basket adding up to zero. When the contents of a Basket are written to BRDB a check is made that the net value of all the accounting lines is indeed zero and should it not be, then an alert is raised and the basket is discarded and an error response returned to the counter.

> *Note that this could only happen as a result of a bug in the code and this check is included specifically to check for any such bugs.*

Baskets are also built up during Back Office Sessions and such Back Office baskets are handled in a similar way to Customer Baskets.