Liquidlogic

protocol

West Yorkshire Child Protection System Integration Specification	
Date	26th March 2004
Version	1.0
Author	Rob Davis and Liz Leach
Status	ISSUE

Document Distribution

Document History

Date	Version	Produced By	Comments
26/03/2004	1.0	Rob Davis and Liz Leach	Create Document

Table of Contents

1 Introduction

This document details the technical architecture that is required by the West Yorkshire Child Protection System (WY CP System) to provide multi-agency and multi-authority sharing of child protection data to assist in the protection of children across West Yorkshire.

It does not detail the child protection processes or the data sharing protocols as these are covered in other documents.

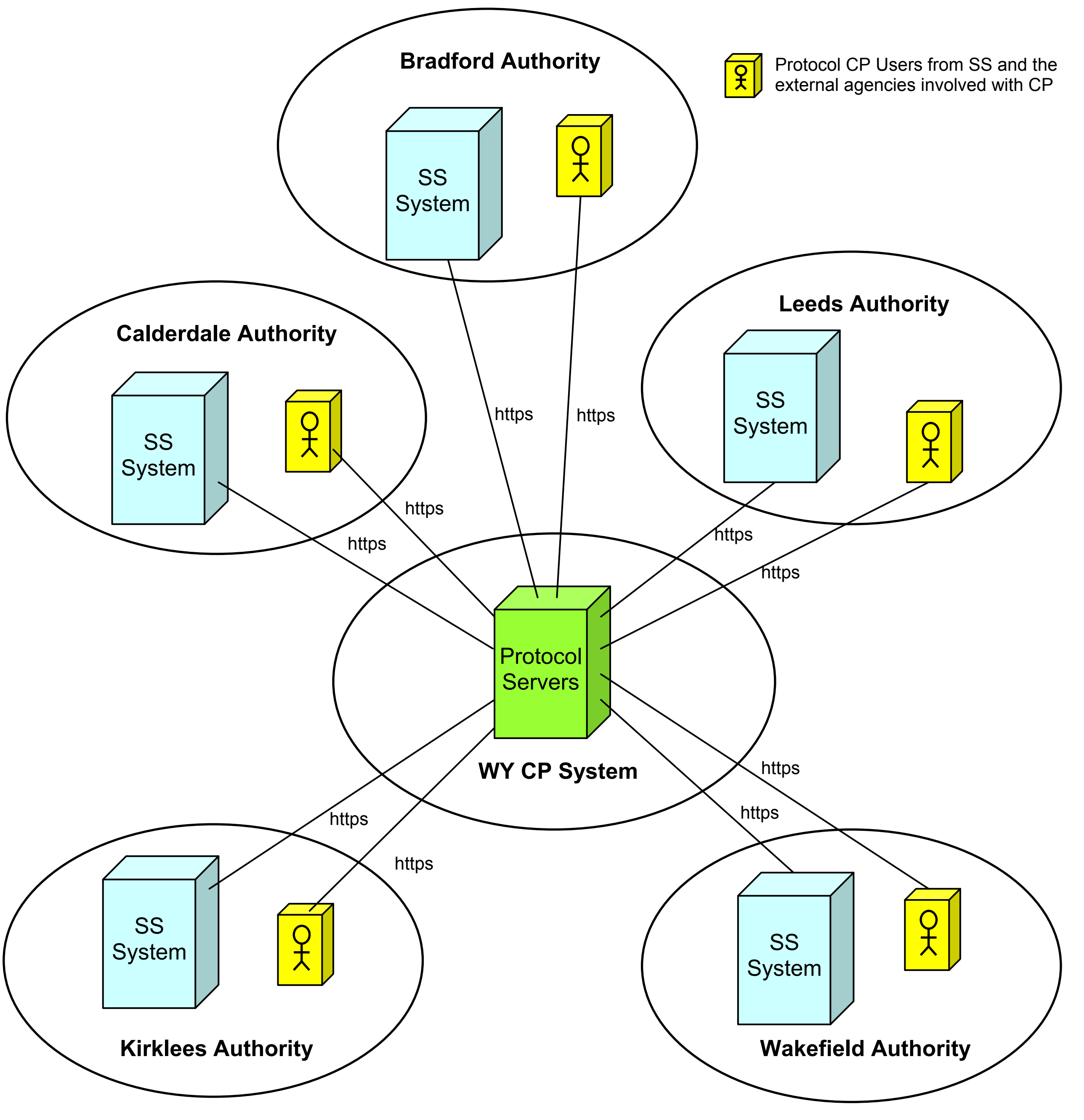
It does detail the technology components within the centrally hosted WY CP System, the communication/integration paths, the mechanisms of integration with the core Social Service systems and the data sets held within the WY CP System.

This document is aimed at a technical audience that is involved with the integration between the WY CP System and the core Social Services systems.

2 WY CP System Overall Architecture

The WY CP System will be hosted centrally and accessed by professionals throughout West Yorkshire that are involved with child protection. The centrally hosted WY CP System connects to each of the core Social Services systems that are hosted within the five West Yorkshire authorities. This architecture is illustrated in Figure 1 below.

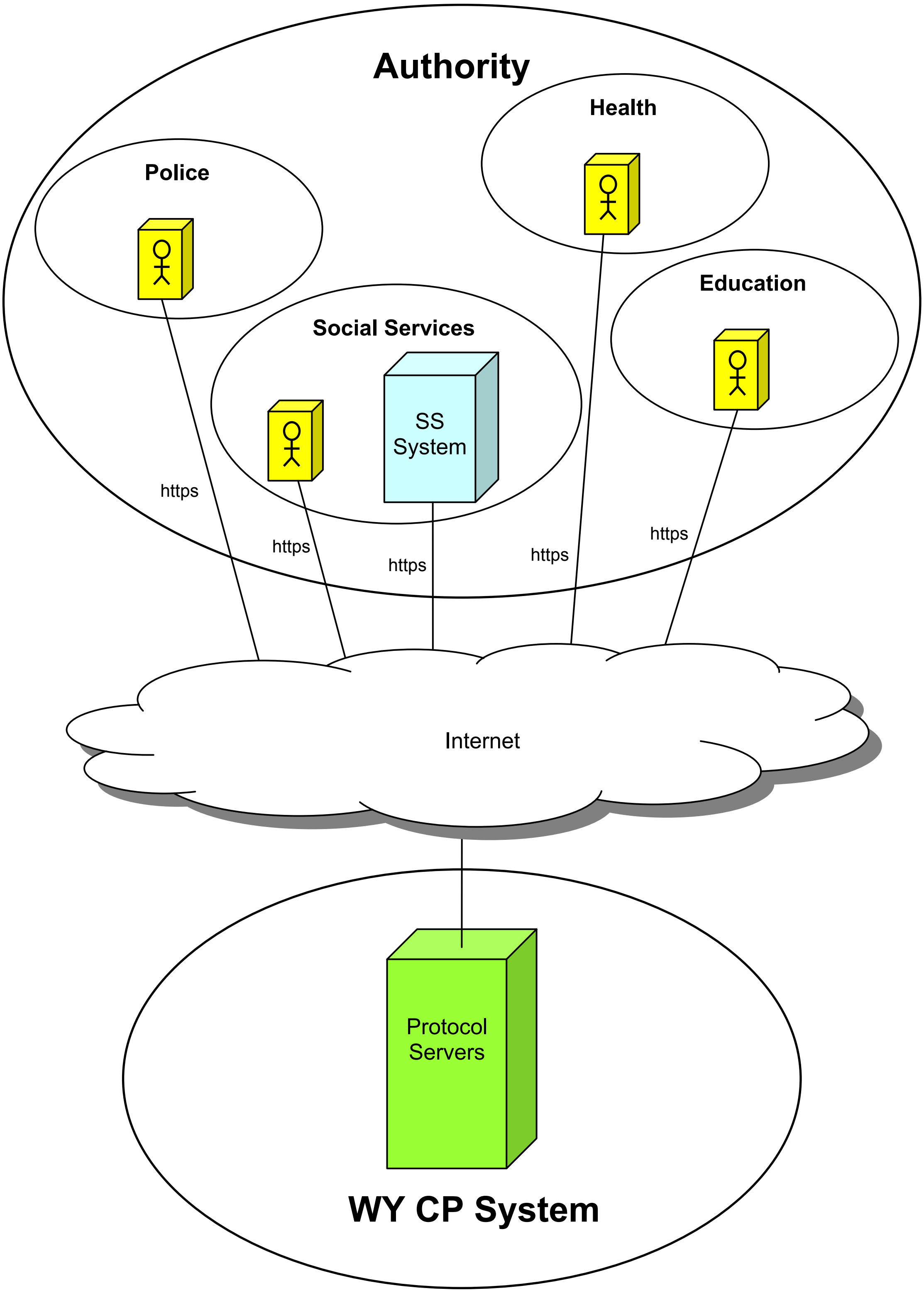
Fig 1. Multi-Authority Participation within West Yorkshire



Protocol is the name of the core application developed by Liquidlogic, which provides the workflow engine, the integration engine required to interface with the core social services systems and the CP user interface used by the professionals.

Within each authority there are various agencies that participate in the protection of children, namely Social Services (child protection unit, children's advice teams, assessment teams and emergency duty teams), Police (child protection unit), Health (A&E, paediatrics, health visitors and child protection teams) and Education (schools, education child protection officers and education social workers). This is illustrated in Figure 2.

Fig 2. Multi-Agency Participation within a single Authority

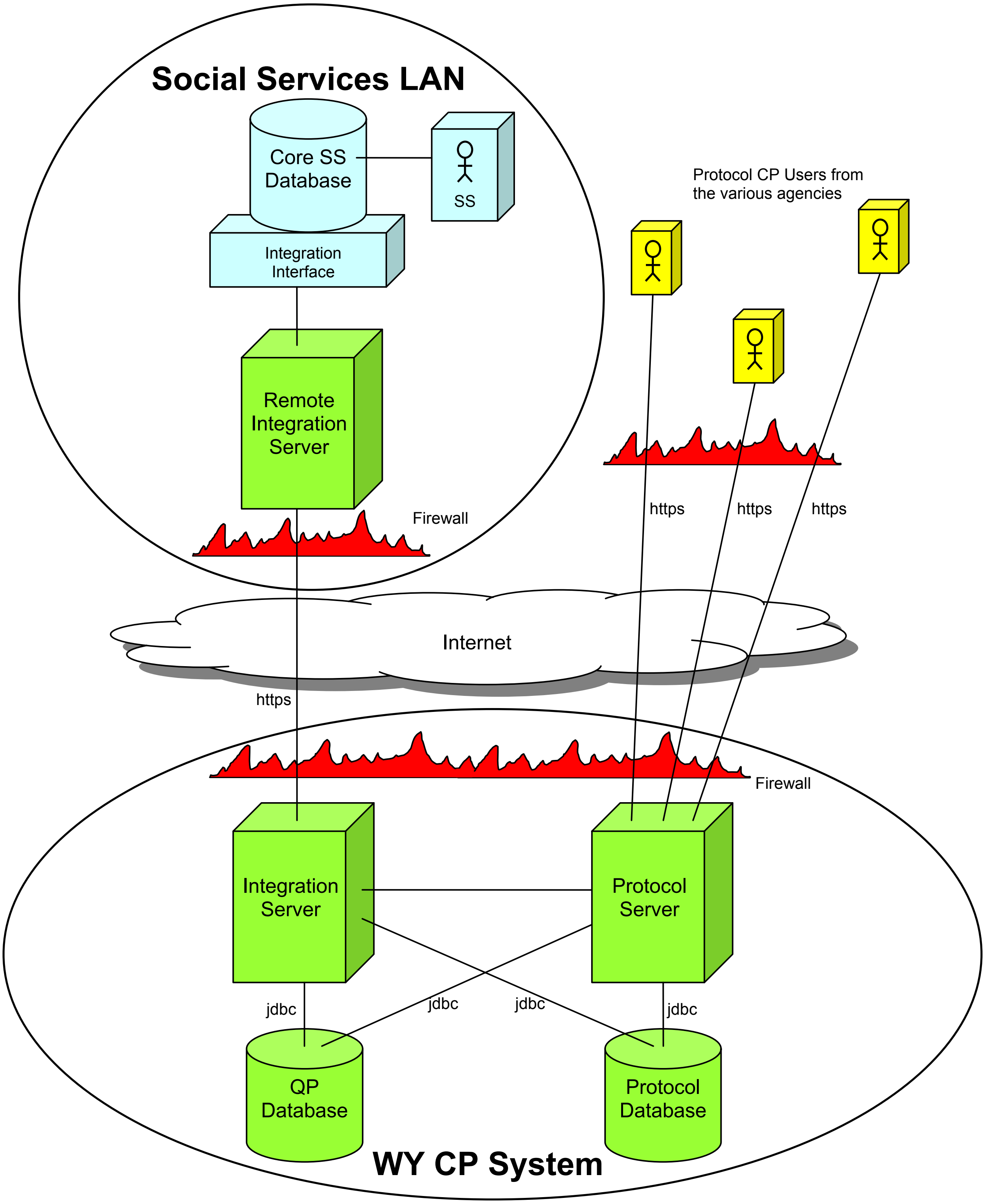


3 WY CP System Integration Architecture

3.1 Architecture Components

The components of the integration architecture reside within the hosted WY CP System environment and within each of the five Social Services LANs. Figure 3 shows how the main components within the integration architecture interconnect. The main components are described after the diagram.

Fig 3. Integration Architecture between a SS System and the WY CP System for a single Authority



3.1.1 Protocol Client

This is the user interface used by the professionals throughout West Yorkshire participating in the WY CP System. The Protocol client can be either a standard internet browser (IE, Netscape) or a thick client. The Protocol client establishes an https connection via the internet to the Protocol Server that is hosted within the WY CP System environment.

3.1.2 SS Client

This is the user interface used by Social Services professionals to access their local Social Services system.

3.1.3 Protocol Server

This is the combined Web Server and Application Server developed by Liquidlogic to facilitate the requirements of the WY CP System. It is developed in Java and runs under Sun's J2SE.

3.1.4 Protocol Database

This holds the core Protocol data such as users, groups, departments, agencies, authorities, security access rights, profiles, diaries, etc., within the WY CP System. It also holds the workflow elements such as tasks within worktrays for users, workgroups, departments, etc., along with all appropriate auditing. The Protocol database will physically be held in the same database as the QP database, but is shown separate within the diagrams as a logical architecture rather than a physical architecture.

3.1.5 Quality Protects (QP) Database

This holds the Child Protection specific data relating to the service users within the core Social Services databases of all five authorities. There will be a single QP database within the hosted WY CP Systems that will hold the data from all the authorities. Although the service users from each core Social Services databases will all be held together within the QP database, services users appearing on more than one core Social Services database will not be merged together into a single record with a single set of demographic data. These service user records from multiple authorities for a particular child will be associated with each other via a link table, and so will keep the QP data corresponding to each authority quite distinct and manageable.

The QP database schema adopted by Liquidlogic to hold the Child Protection specific data is modelled upon the Quality Protects schema recommended as an outcome from the Department of Health's Quality Protects Programme.

The data that is held within the QP databases is to be synchronised dynamically with the corresponding Social Service database. This includes the following datasets: service users, aliases, addresses, relationships, cp registrations, cp enquiries, cp offences and security relating to individual service users. These are detailed comprehensively later in this document. There may be some child protection data that needs to be held within the QP database that is not held on some of the core Social Services database.

The datasets that are not held within the QP database, and that are accessed directly from the core Social Service database are: languages, contacts, case notes, legal status, referrals, assessments, investigations, strategy discussions, interviews, case conferences, reviews, involved professionals, organisations and warnings.

3.1.6 Integration Server (IS)

This is the integration engine developed by Liquidlogic to integrate with the remote Social Services systems. It provides the mapping algorithm functionality required to map the QP database schema to the core Social Services database schema, and provides the mechanism to keep both data sources synchronised.

3.1.7 Remote Integration Server (RIS)

This is another Liquidlogic integration engine, but is designed to be hosted remotely from the Protocol Server. It connects directly to the Social Services system via the provided integration interface and connects back to the Integration Server hosted alongside the Protocol Server. It can exposes a web services interface to allow the Social Services system to pass in data changes that have occurred within the Social Services database to be synchronised with the WY CP System. It is a lightweight server that does not need a database.

It does not hold any mapping algorithms locally, as this is carried out by the centrally hosted Integration Server, which dynamically sends through the pre-format messages required to integrate with the specific Social Services system.

It is configured with the required technologies to connect to the Social Service system integration interface. The authentication details needed to connect to the integration interface of the Social Service system are securely encrypted within the Remote Integration Server.

3.1.8 Social Services System

The Social Services system can comprise of 2 components, the core Social Services database and the integration interface required to access the data within the database. The database can be accessed directly, where the integration interface is effectively within the database. Alternatively, connection to an external integration interface may be needed in order to access the data within the database. The details of the integration interface for each authority will be specific the each Social Services system and are included later on in this document.

3.2 Integration Security

Firewalls protect the Integration Servers and the Remote Integration Servers from attack via the internet. The Social Services firewall can be configured to only allow the IP address of the centrally hosted WY CP System to enter through a specified port number.

Both the Integration Server and the Remote Integration Server will have a server certificate signed by an appropriate CA, and will each hold the public key certificate of the other as trusted.

All http communications are over secure sockets (https), using the 128 bit triple-strength data encryption standard (3-DES).

3.3 Integration Configuration

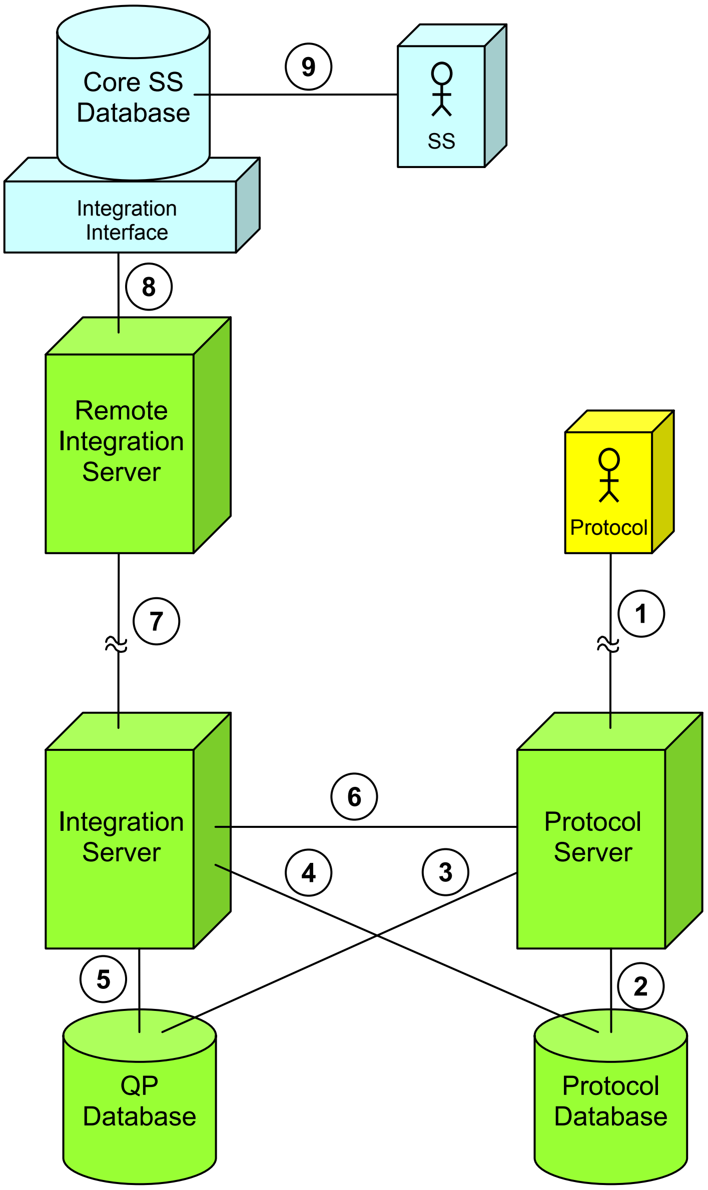
The schemas of the data needed to be accessed within the Social Services systems are held within the corresponding Integration Server as XML entities. These entities hold the information relating to the mechanism of integration and the mapping algorithms employed between the QP and the SS databases. There will also be a QP to SS mapping for data that is not held within the QP database (i.e. data that is accessed directly on the SS database and not duplicated within the WY CP System). This enables a common CP workflow application to be developed without the need to worry about the differences in the Social Services system behind the scenes.

These XML entities are specific to each authority, and are maintained independently. Any changes to a Social Services database or integration interface need to be reflected within theses entities, but will not affect the integration to the other systems.

3.4 Communication and Integration Paths

This section follows through the integration scenarios for a single authority, considering read/write from a Protocol user and from a Social Services system user. Figure 4 shows the step numbers referred to in these scenarios.

Fig 4. Stepping through the Integration Scenarios for a single Authority



3.4.1 Protocol User Request Scenarios

(The numbers listed below relate to the steps identified in Figure 4 above)

- **Read from Protocol/QP databases** (1,2,3,1)
  1 – Protocol client requests data to be read from Protocol/QP databases.
  2 – Protocol Server reads from Protocol database.
  3 – Protocol Server reads from QP database.
  1 – Data returned to Protocol client.

- **Write to Protocol database** (1,2,1)
  1 – Protocol client requests data to be written to Protocol database.
  2 – Protocol Server writes to Protocol database.
  1 – Outcome response returned to Protocol client .

- **Write to QP database** (1,3,5,4,7,8,7,5)
  1 – Protocol client requests data to be written to QP database.
  3 – Protocol Server writes to QP database.
  5 – Integration Server picks up changes made to the QP database.
  4 – Integration Server reads any required data from Protocol database.
  7 – Integration server executes the mapping algorithm, generates the integration message in the form appropriate for the specific Social Services system integration interface (i.e. XML GovTalk message, SQL statement, etc) and passes it to the Remote Integration Server.
  8 – Remote Integration Server connects to the Social Services system integration interface to pass in the data. If the write is a create (rather than modify or delete), then the new unique identifier for the created record is received back by the Remote Integration Server assuming successfully created.
  7 – The Remote Integration Server passes back the outcome to the Integration Server, and the new unique identifier if the write was a create.
  5 – The Integration Server updates the QP database record with the unique identifier created by the Social Service system if the write was a create.

- **Read direct from Core SS database** (1,6,7,8,7,6,1)
  1 – Protocol client requests data to be read direct from core SS database.
  6 – Protocol Server passes request directly onto the Integration Server.
  7 – Integration server executes the mapping algorithm, generates the integration message in the form appropriate for the specific Social Services system integration interface (i.e. XML GovTalk message, SQL statement, etc) and passes it to the Remote Integration Server.
  8 – Remote Integration Server connects to the Social Service system integration interface to request the data.
  7 – The Remote Integration Server passes back the requested data to the Integration Server.
  6 – The Integration Server passes back the requested data to the Protocol Server.
  1 – The Protocol Server passes back the requested data to the Protocol client.

- **Write direct to Core SS database** (1,6,7,8,7,6,1)

  1 – Protocol client requests data to be written direct to core SS database.
  6 – Protocol Server passes request directly onto the Integration Server.
  7 – Integration server executes the mapping algorithm, generates the integration message in the form appropriate for the specific Social Services system integration interface (i.e. XML GovTalk message, SQL statement, etc) and passes it to the Remote Integration Server.
  8 – Remote Integration Server connects to the Social Service system integration interface to pass in the data.
  7 – The Remote Integration Server passes back the outcome to the Integration Server.
  6 – The Integration Server passes back the outcome to the Protocol Server.
  1 – The Protocol Server passes back the outcome to the Protocol client.

3.4.2 Social Services System User Request Scenarios

(The numbers listed below relate to the steps identified in Figure 4 above)

- **Read from Core SS database** (9)

  9 – Social Services system client requests data to be read from core SS databases.

Option A:
This option applies where the SS integration interface allows for the Remote Integration Server to "PULL" the data changes that have occurred within the core SS database.

- **Write to Core SS database** (9,7,8,7,5,7,8,7)

  9 – Social Services system client requests data to be written to core SS databases (create, modify or delete).
  7 – Integration Server makes a scheduled request to the Remote Integration Server for a list of any core SS databases changes that have been made since the last scheduled request (only changes that need to be synchronised with the QP database).
  8 – Remote Integration Server connects to the Social Services system integration interface to request any relevant changes. These changes are stored in "Transfer Tables" on the core SS database, and will be marked as having been received by the Remote Integration Server, but will not be deleted at this stage.
  7 – Remote Integration Server passes any changes back to the Integration Server.
  5 – Integration Server transforms the data received from the Social Services specific integration interface (i.e. XML GovTalk message, SQL statement, etc) into an appropriate format, executes the mapping algorithm and updates the QP database accordingly (create, modify or delete).
  7 – Integration Server passes back outcome to Remote Integration Server.
  8 – If the outcome is successful, then the Remote Integration Server requests via the integration interface that the marked records in the "Transfer Tables" be deleted.
  7 – The Remote Integration Server sends the outcome back to the Integration Server.

Option B:
This option applies where the SS integration interface needs to "PUSH" the data changes that have occurred within the core SS database to the Remote Integration Server.

- **Write to Core SS database** (9,8,7,5,7,8)

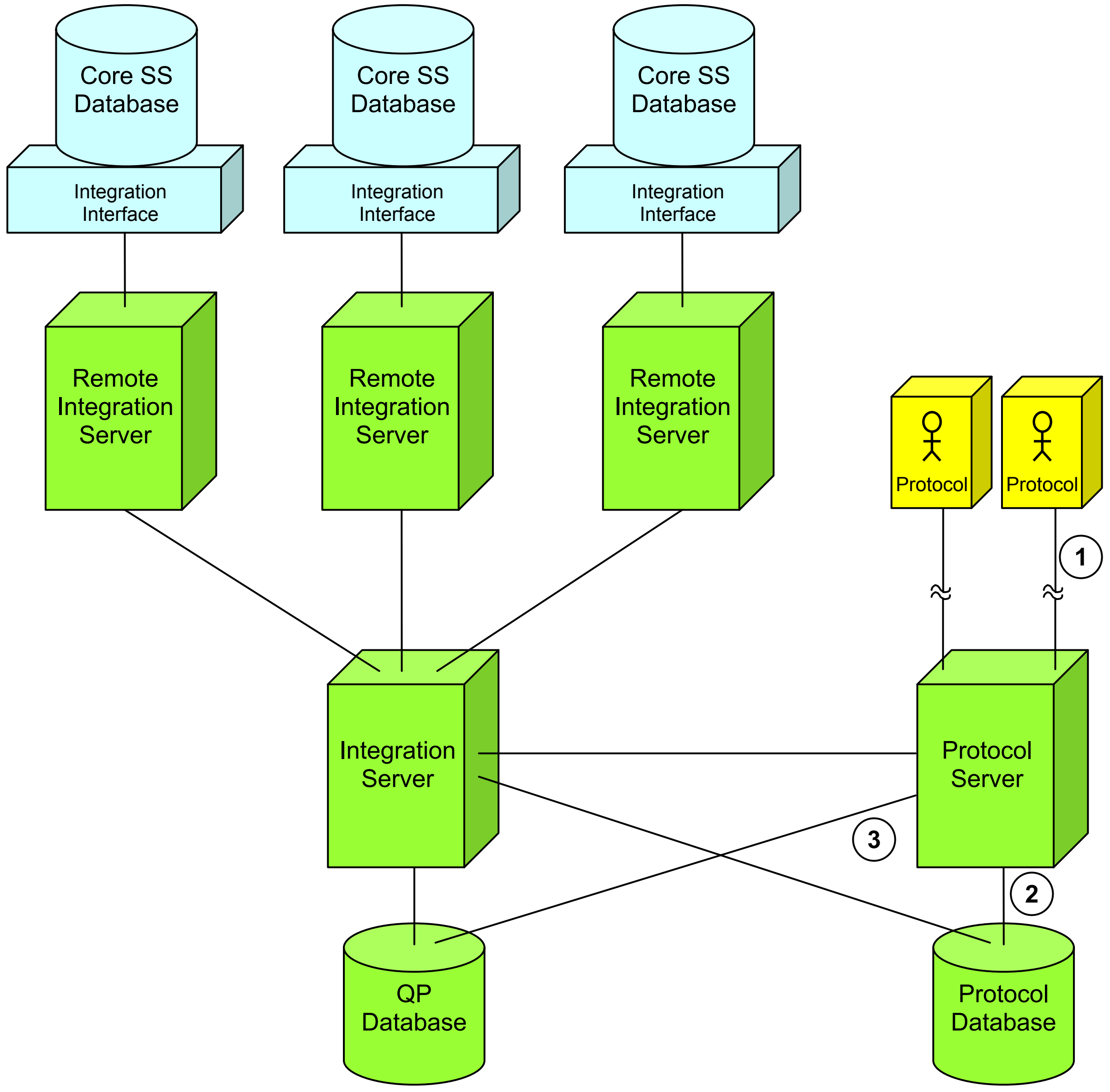
  9 – Social Services system client requests data to be written to the core SS databases (create, modify or delete).
  8 – Social Services system integration engine connects to the web services interface exposed by the Remote Integration Server to pass the data changes that need to be synchronised into the QP database.
  7 – Remote Integration Server passes the web services message back to the Integration Server.
  5 – Integration Server validates the message and applies the changes to the QP database.
  7 – Integration Server passes back response to Remote Integration Server.
  8 – Remote Integration Server completes web services transaction back to the Social Services system integration engine with the appropriate response.

3.5 Multi-Authority View for CP Registry Enquiries

Figure 5 shows integration to three Social Services systems.

When a professional within any of the authorities makes a child protection registry enquiry against the WY CP System via a Protocol client (1), then the Protocol Server can view the data within the Protocol and QP database (2 and 3). All the data required is within the QP database to service the CP registry enquiry request (names, aliases, date of births, genders, addresses, siblings, current/previous registrations, and previous CP registry enquiries). The QP database is optimised for these searches, and so a very prompt response will be able to be delivered back to the Protocol user.

Fig 5. Multi-Authority View for CP Registry Enquiries across three Authorities



3.6 Physical Architecture

The diagrams shown so far within this document have illustrated the execution architecture as opposed to the physical architecture. The following describes the physical architecture of the main components of the WY CP System:

3.6.1 Protocol and Integration Servers

The Protocol and Integration Servers will be hosted as a farm of servers within a clustered environment. The https requests coming from Protocol clients from within the five authorities and the requests coming from the Remote Integration Servers will be distributed amongst the servers within the cluster via a load balancing mechanism. The cluster allows for horizontal scalability, which provides the fail-over, high availability and performance requirements.

3.6.2 QP and Protocol Databases

The QP and Protocol databases will be hosted on a clustered pair of database servers, configured with passive standby.

3.6.3 Remote Integration Servers

Each Remote Integration Server will be a single resilient server hosted within the Social Services LAN where it can directly access the Social Services system integration interface. It must have direct access to the internet through the local firewall, where the firewall is configured with the agreed port number and IP address. The internet connection must be open for both incoming and outgoing https connections.

3.7 Systems Down

If one or more of the Social Services systems are down or the communications to them are down for whatever reason, the WY CP System will still be able to service child protection registry enquiry requests across all five authorities from professionals throughout West Yorkshire. The datasets that are not held on the centrally hosted QP databases for these Social Services systems will obviously not be available until the systems and communications are back up again.

Any changes made to the data held within the QP databases whilst the Social Services systems are down will be automatically synchronised to the Social Services system when it comes back up again.

Visa versa, if the WY CP System is down for any period of time, any changes made within the core SS databases will be synchronised into the QP databases when the WY CP Systems comes back up again.

4 Authority Specific Integration Design

4.1 Bradford Integration Design

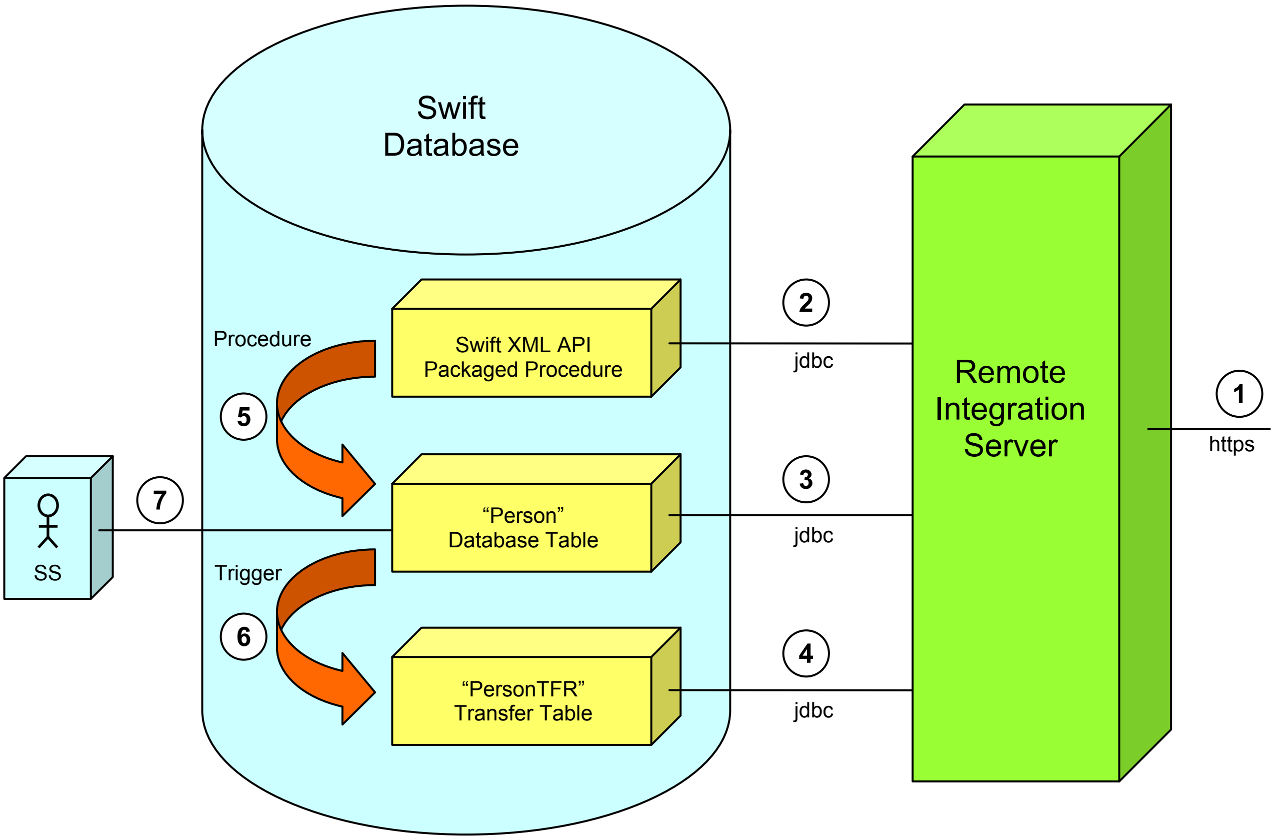
The Social Services system used within Bradford Authority is Swift v16 supplied by Anite PLC. The API being exposed is within the Swift database, which is accessed via a jdbc database connection.

The data read is achieved by direct queries against the actual Swift database tables that contain the required data.

The data write is achieved by passing a GovTalk XML message into a Swift database packaged procedure called “SXAPI.SXAPI_Interface”. See appendix A for an example GovTalk XML messages to create a person record within Swift. The outcome is passed back via a parameter of the packaged procedure as a GovTalk XML message.

Figure 6 illustrates the flow of the integration to the Swift database, identifying the step numbers referred to in the read/write scenarios described after the diagram. One example database table called “Persons” has been used throughout the scenarios and within the diagram.

Fig 6. Specific Integration with the Bradford SS Core System



4.1.1 Protocol Server Request Scenarios

(The numbers listed below relate to the steps identified in Figure 6 above)

- Read from "Person" database table (1,3,1)
  - 1 – Remote Integration Server receives an https request from the Integration Server to read data from the "Person" table. The sql statement will be already formed by the Integration Server and ready to execute.
  - 3 – Remote Integration Server uses this sql statement to directly query the "Person" table via a jdbc connection to the Swift database.
  - 1 – The results from the query are returned back to the Integration Server.

- Write to "Person" database table (1,2,5,2,6,4,1)
  - 1 – Remote Integration Server receives an https request from the Integration Server to write the person data. The GovTalk XML message will be already formed by the Integration Server and ready to pass into the Swift XML API packaged procedure.
  - 2 – Remote Integration Server passes this GovTalk XML message into the Swift XML API packaged procedure.
  - 5 – The Swift XML API packaged procedure validates the request and if authorised will insert, update or delete from the "Person" table as appropriate.
  - 2 – The resulting GovTalk XML message will be received back from the Swift XML API packaged procedure by the Remote Integration Server. If the request was to create a person, then the unique identifier for the new person will be contained within this GovTalk XML message.
  - 6 – The transfer database trigger will fire against the "Person" table and insert a record into the "PersonTFR" database table.
  - 4 – The Remote Integration Server deletes this transfer record from the "PersonTFR" table to prevent the change feeding back to the Integration Server.
  - 1 – Remote Integration Server passes back the resulting GovTalk XML message to the Integration Server.

4.1.2 Swift Client Request Scenarios

(The numbers listed below relate to the steps identified in Figure 6 above)

- Read from “Person” database table (7)
  7 – Swift client reads data from “Person” table.

- Write to “Person” database table (7,6,1,4,1,1,4,1)
  7 – Swift client writes data to the “Person” table.
  6 – The transfer database trigger will fire against the “Person” table and inserts a record into the “PersonTFR” database table.
  1 – Remote Integration Server receives a request from the Integration Server for a list of changes (inserts, updates, deletes) that have been made to the “Person” table that need to be synchronised back into the centrally hosted QP database. This request contains the sql statement already formed in order to carry out the query against the “PersonTFR” table.
  4 – Remote Integration Server executes the sql query against the “PersonTFR” table. Theses transfer records are marked as been sent to the Integration Server by an update sql statement on the “PersonTFR” table.
  1 – Remote Integration Server passes the list of “PersonTFR” table records back to the Integration server.
  1 – Remote Integration Server receives a request from the Integration Server to delete the “PersonTFR” records if all is successful. This request contains the sql statement already formed in order to carry out the delete against the “PersonTFR” table.
  4 – Remote Integration Server uses this delete statement to delete the marked “PersonTFR” records.
  1 – Remote Integration Server sends outcome back to the Integration Server.

4.2 Wakefield Integration Design

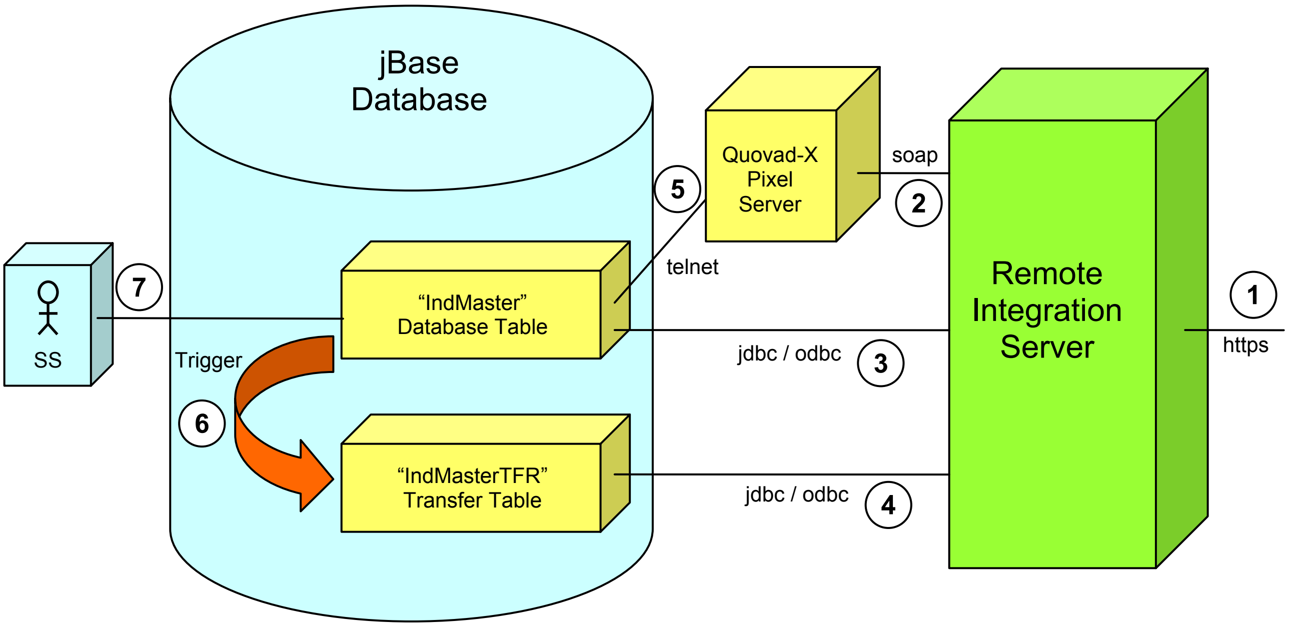
The Social Services system used within Wakefield Authority is jBase. Two APIs are exposed to the Remote Integration Server, one for reading and the other for writing.

The read is achieved by queries against the jBase database tables that contain the required data via the ISG Navigator odbc driver.

The write is achieved by connecting to the Pixel SOAP Server supplied by Quovad-X and calling the appropriate functions. See appendix B for an example WSDL that describes the connection and functions. The Pixel Server navigates around the jBase database via a telnet session in order to insert, update and delete from the jBase database.

Figure 7 illustrates the flow of the integration to the jBase database, identifying the step numbers referred to in the read/write scenarios described after the diagram. One example database table called "IndMaster", which is the master client index table, has been used throughout the scenarios and within the diagram.

Fig 7. Specific Integration with the Wakefield SS Core System



4.2.1 Protocol Server Request Scenarios

(The numbers listed below relate to the steps identified in Figure 7 above)

- Read from "IndMaster" database table (1,3,1)

  1 – Remote Integration Server receives an https request from the Integration Server to read data from the "IndMaster" table. The sql statement will be already formed by the Integration Server and ready to execute.

  3 – Remote Integration Server uses this sql statement to directly query the "IndMaster" table via a jdbc/odbc connection to the jBase database.

  1 – The results from the query are returned back to the Integration Server.

- Write to "IndMaster" database table (1,2,5,2,6,4,1)

  1 – Remote Integration Server receives an https request from the Integration Server to write the person data. The SOAP XML message will be already formed by the Integration Server ready to pass to the Web Services SOAP server.

  2 – Remote Integration Server connects to the web services soap server exposed by the Quovad-X Pixel Server and passes in the SOAP XML message.

  5 – The Quovad-X Pixel Server validates the request and if authorised will insert, update or delete from the "IndMaster" table as appropriate. This is achieved via telnet navigation.

  2 – The resulting SOAP XML message will be received back from the Web Services SOAP server by the Remote Integration Server. If the request was to create a person, then the unique identifier for the new person will be contained within this SOAP XML message.

  6 – The transfer database trigger will fire against the "IndMaster" table and insert a record into the "IndMasterTFR" database table.

  4 – Remote Integration Server deletes this transfer record from the "IndMasterTFR" table to prevent the change feeding back to the Integration Server.

  1 – Remote Integration Server passes back the resulting SOAP XML message to the Integration Server.

4.2.2 jBase Client Request Scenarios

(The numbers listed below relate to the steps identified in Figure 7 above)

- Read from "IndMaster" database table (7)
  7 – jBase client reads data from "IndMaster" table.

- Write to "IndMaster" database table (7,6,1,4,1,1,4,1)
  7 – jBase client writes data to the "IndMaster" table.
  6 – The transfer database trigger will fire against the "IndMaster" table and inserts a record into the "IndMasterTFR" database table.
  1 – Remote Integration Server receives a request from the Integration Server for a list of changes (inserts, updates, deletes) that have been made to the "IndMaster" table that need to be synchronised back into the centrally hosted QP database. This request contains the sql statement already formed in order to carry out the query against the "IndMasterTFR" table.
  4 – Remote Integration Server executes the sql query against the "IndMasterTFR" table. Theses transfer records are marked as been sent to the Integration Server by an update sql statement on the "IndMasterTFR" table.
  1 – Remote Integration Server passes the list of "IndMasterTFR" table records back to the Integration server.
  1 – Remote Integration Server receives a request from the Integration Server to delete the "IndMasterTFR" records if all is successful. This request contains the sql statement already formed in order to carry out the delete against the "IndMasterTFR" table.
  4 – Remote Integration Server uses this delete statement to delete the marked "IndMasterTFR" records.
  1 – Remote Integration Server sends outcome back to the Integration Server.

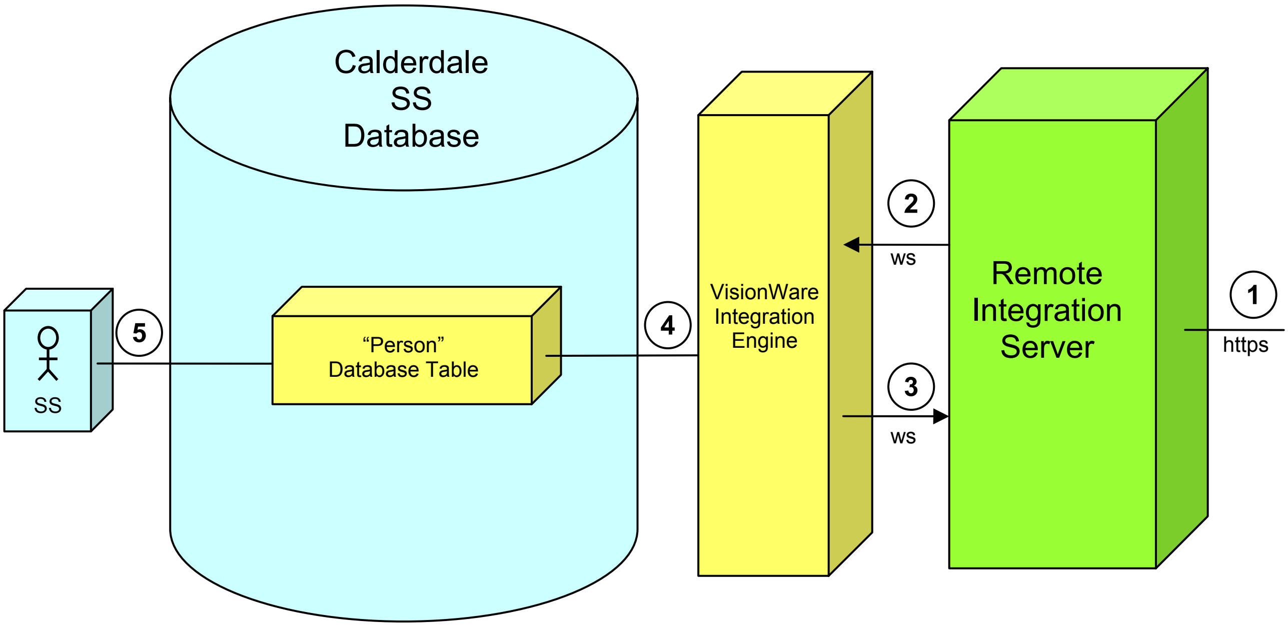
4.3 Calderdale Integration Design

The Social Services system used within Calderdale Authority will be integrated via the VisionWare integration engine. The VisionWare integration engine will expose a web services interface to allow the Remote Integration Server to push/pull data, and the Remote Integration Server will expose a web services interface to allow the VisionWare integration engine to push data.

It is important to note that the VisionWare integration engine needs to maintain reliable messaging when passing data changes into the web services interface being exposed by the Remote Integration Server. It is also important that the data elements within the Calderdale SS database have appropriate version/checksum validation in order to maintain the integrity of the data and prevent update from both systems to the same data.

Figure 8 illustrates the flow of the integration to the Calderdale Social Services database, identifying the step numbers referred to in the read/write scenarios described after the diagram. A made up example database table called "Person" has been used throughout the scenarios and within the diagram.

Fig 8. Specific Integration with the Calderdale SS Core System



4.3.1 Protocol Server Request Scenarios

(The numbers listed below relate to the steps identified in Figure 8 above)

- Read from “Person” database table (1,2,4,2,1)

  1 – Remote Integration Server receives an https request from the Integration Server to retrieve data from the “Person” table.
  2 – Remote Integration Server connects to the web services interface exposed by the VisionWare integration engine and sends the message to retrieve data from the “Person” table. (Remote Integration Server “PULL” from VisionWare integration engine)
  4 – VisionWare integration engine retrieves the required data from the Calderdale SS database.
  2 – Message response sent back to the Remote Integration Server.
  1 – The message response is returned back to the Integration Server.

- Write to “Person” database table (1,2,4,2,1)

  1 – Remote Integration Server receives an https request from the Integration Server to write data to the “Person” table.
  2 – Remote Integration Server connects to the web services interface exposed by the VisionWare integration engine and sends the message to write data to the “Person” table. (Remote Integration Server “PUSH” to VisionWare integration engine)
  4 – VisionWare integration engine writes the required data to the Calderdale SS database.
  2 – Message response sent back to the Remote Integration Server.
  1 – The message response is returned back to the Integration Server.

4.3.2 Calderdale SS Client Request Scenarios

(The numbers listed below relate to the steps identified in Figure 8 above)

- Read from “Person” database table (5)

  5 – Client reads data from “Person” table.

- Write to “Person” database table (5,4,3,1,3)

  5 – Client writes data to the “Person” table.
  4 – This change is picked up by the VisionWare integration engine.
  3 – The VisionWare integration engine establishes a connection to the web services interface exposed by the Remote Integration Server and passes these changes in as a message. If the connection is unable to be established or a fail to send response is received, the message must be reliable and retried appropriately and ultimately sent to a dead message queue. (VisionWare integration engine “PUSH” to Remote Integration Server)
  1 – The Remote Integration Server sends the message to the Integration Server.
  3 – The response is returned back to VisionWare integration engine.

4.4 Kirklees Integration Design

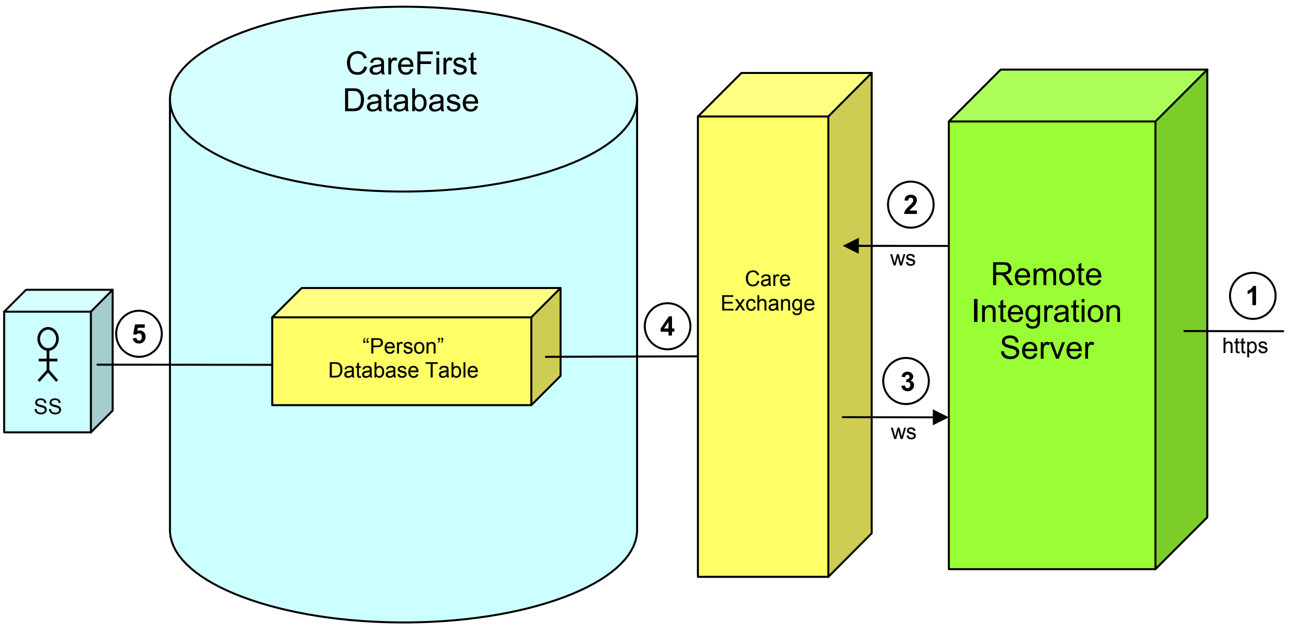
The Social Services system used within Kirklees Authority is CareFirst supplied by OLM and will be integrated via CareExchange supplied by OLM. CareExchange will expose a web services interface to allow the Remote Integration Server to push/pull data, and the Remote Integration Server will expose a web services interface to allow CareExchange to push data.

If CareExchange does not have the required level of integration available in time, then the integration mechanism detailed within the Bradford Integration Design will need to be implemented.

It is important to note that CareExchange needs to maintain reliable messaging when passing data changes into the web services interface being exposed by the Remote Integration Server. It is also important that the data elements within the CareFirst database have appropriate version/checksum validation in order to maintain the integrity of the data and prevent update from both systems to the same data.

Figure 9 illustrates the flow of the integration to the Kirklees Social Services database, identifying the step numbers referred to in the read/write scenarios described after the diagram. A made up example database table called "Person" has been used throughout the scenarios and within the diagram.

Fig 9. Specific Integration with the Kirklees SS Core System



4.4.1 Protocol Server Request Scenarios

(The numbers listed below relate to the steps identified in Figure 9 above)

- Read from “Person” database table (1,2,4,2,1)
  1 – Remote Integration Server receives an https request from the Integration Server to retrieve data from the “Person” table.
  2 – Remote Integration Server connects to the web services interface exposed by CareExchange and sends the message to retrieve data from the “Person” table. (Remote Integration Server “PULL” from CareExchange)
  4 – CareExchange retrieves the required data from CareFirst.
  2 – Message response sent back to the Remote Integration Server.
  1 – The message response is returned back to the Integration Server.

- Write to “Person” database table (1,2,4,2,1)
  1 – Remote Integration Server receives an https request from the Integration Server to write data to the “Person” table.
  2 – Remote Integration Server connects to the web services interface exposed by the CareExchange and sends the message to write data to the “Person” table. (Remote Integration Server “PUSH” to CareExchange)
  4 – CareExchange writes the required data to CareFirst.
  2 – Message response sent back to the Remote Integration Server.
  1 – The message response is returned back to the Integration Server.

4.4.2 CareFirst Client Request Scenarios

(The numbers listed below relate to the steps identified in Figure 9 above)

- Read from “Person” database table (5)
  5 – Client reads data from “Person” table.

- Write to “Person” database table (5,4,3,1,3)
  5 – Client writes data to the “Person” table.
  4 – This change is picked up by CareExchange.
  3 – CareExchange establishes a connection to the web services interface exposed by the Remote Integration Server and passes these changes in as a message. If the connection is unable to be established or a fail to send response is received, the message must be reliable and retried appropriately and ultimately sent to a dead message queue. (CareExchange “PUSH” to Remote Integration Server)
  1 – The Remote Integration Server sends the message to the Integration Server.
  3 – The response is returned back to CareExchange.

4.5 Leeds Integration Design

The Social Services system used within Leeds Authority is supplied by Siebel. The Siebel integration engine will expose a web services interface to allow the Remote Integration Server to push/pull data, and the Remote Integration Server will expose a web services interface to allow the Siebel integration engine to push data.

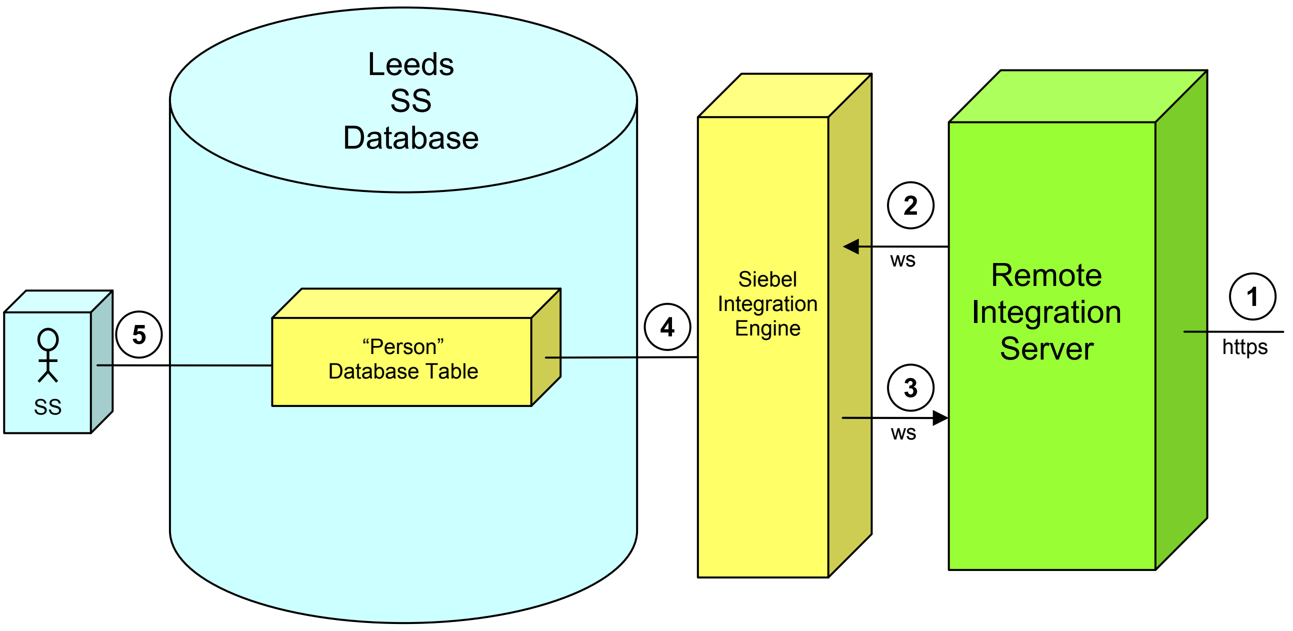
If the Siebel integration engine does not have the required level of integration available in time, then the integration mechanism detailed within the Bradford Integration Design will need to be implemented in order to achieve the read/write integration required.

If neither of the read/write integration mechanisms will be available in time then a read only real-time link to the SS database will need to be implemented. If this is not available in time then an automated daily export from the Leeds Social Services database will need to be imported into the QP database hosted by the WY CP System. This will result in the data within the QP database being read only.

It is important to note that the Siebel integration engine needs to maintain reliable messaging when passing data changes into the web services interface being exposed by the Remote Integration Server. It is also important that the data elements within the SS database have appropriate version/checksum validation in order to maintain the integrity of the data and prevent update from both systems to the same data.

Figure 10 illustrates the flow of the integration to the Leeds Social Services database, assuming that the Siebel integration engine is available with a web services interface, identifying the step numbers referred to in the read/write scenarios described after the diagram. A made up example database table called "Person" has been used throughout the scenarios and within the diagram.

Fig 10. Specific Integration with the Leeds SS Core System



4.5.1 Protocol Server Request Scenarios

(The numbers listed below relate to the steps identified in Figure 9 above)

- Read from “Person” database table (1,2,4,2,1)
  - 1 – Remote Integration Server receives an https request from the Integration Server to retrieve data from the “Person” table.
  - 2 – Remote Integration Server connects to the web services interface exposed by the Siebel integration engine and sends the message to retrieve data from the “Person” table. (Remote Integration Server “PULL” from the Siebel integration engine)
  - 4 – The Siebel integration engine retrieves the required data from the SS database.
  - 2 – Message response sent back to the Remote Integration Server.
  - 1 – The message response is returned back to the Integration Server.

- Write to “Person” database table (1,2,4,2,1)
  - 1 – Remote Integration Server receives an https request from the Integration Server to write data to the “Person” table.
  - 2 – Remote Integration Server connects to the web services interface exposed by the Siebel integration engine and sends the message to write data to the “Person” table. (Remote Integration Server “PUSH” to the Siebel integration engine)
  - 4 – The Siebel integration engine writes the required data to the SS database.
  - 2 – Message response sent back to the Remote Integration Server.
  - 1 – The message response is returned back to the Integration Server.

4.5.2 Leeds SS Client Request Scenarios

(The numbers listed below relate to the steps identified in Figure 9 above)

- Read from “Person” database table (5)
  - 5 – Client reads data from “Person” table.

- Write to “Person” database table (5,4,3,1,3)
  - 5 – Client writes data to the “Person” table.
  - 4 – This change is picked up by the Siebel integration engine.
  - 3 – The Siebel integration engine establishes a connection to the web services interface exposed by the Remote Integration Server and passes these changes in as a message. If the connection is unable to be established or a fail to send response is received, the message must be reliable and retried appropriately and ultimately sent to a dead message queue. (Siebel integration engine “PUSH” to Remote Integration Server)
  - 1 – The Remote Integration Server sends the message to the Integration Server.
  - 3 – The response is returned back to the Siebel integration engine.

5 Integration Data Sets

This section details the data sets which have been integrated as part of the Bradford Child Protection System and will form the West Yorkshire CP System integration requirements.

Some information that requires integration will relate to the security, confidentiality or disclosure of specific data sets and/or fields. For data that falls into this category and requires integration, the West Yorkshire Child Protection System Information Sharing Protocol will dictate how this information is protected and shared between authorised users of the system. This will include access to child protection registration information, confidential addresses, warning indicators and details, and the child protection offender register.

5.1 Service User / Person Information

The service user data set contains demographic and other personal information relating to children and adults and professionals that are known to Social Services.
The information contained within this data set includes forenames, surname, date of birth, gender, title, NHS No, ethnicity, religion etc.

5.1.1 Aliases

The aliases data set contains details of any other names by which a service user is known. This will include legal name changes (such as married name) as well as other names or variants on their legal name by which a service user is known.

The Liquidlogic application will use any available aliases within its search facilities and therefore this is a key area for integration. This ensures that children or adults who are known to use different names or variations on names are more likely to be identified by a user.

5.1.2 Languages

The language data set will identify the language and communication issues for a specific service user. This will include languages known by a service user, their fluency in those languages and their preferred/primary language.

Protocol can cater for both a single “primary” language identifier stored against a service user or more commonly, a list of known languages and related information.

5.1.3 Service User Security

Some core systems have the ability to include or exclude certain users from being able to access information relating to specific service users held on the system. Protocol uses a confidential security field to reflect this security control for the equivalent Protocol users where appropriate.

Typically any such security will only apply to Social Services professionals who have user accounts on the core social services system. This security therefore works in conjunction with the restricted security access within Protocol that Social Services may use to dynamically grant or revoke privileges to share information with professionals external to Social Services.

5.2 Addresses

The address set includes both actual physical addresses and also the cross referencing of service users to those addresses. Each of these is described in more detail below.

5.2.1 Address Gazetteer

The address gazetteer will include a list of all local addresses covered by the CSSR in addition to specific addresses for individuals for whom that CSSR is responsible.

Most core social services systems provide access to a local address gazetteer containing the addresses covered by that particular CSSR. The Protocol CP application will use these addresses to ensure that addresses are correctly identified. Protocol CP will not allow the creation of new addresses or the modification of existing addresses; this functionality will only be available within the core system.

5.2.2 Service User Addresses

This data set allows the identification of all current and previous addresses relating to a service user. The data in this set will contain information that associates addresses in the gazetteer with a specific service user. Some core systems may also contain additional information to indicate the primary address for an individual or that an address is confidential.

Child protection register enquiries performed by the Protocol CP system will check both current and previous addresses for matching children and siblings. To facilitate this the Protocol CP system requires access to a list of all addresses, both current and previous, for service users.

5.3 Service User Contact Details

Contact information will normally include a list of contact telephone numbers, both current and historical, for a service user.

5.4 Service User Case notes

Case notes recorded against a service user or a specific episode or case for a service user and will include details of the type of note, the professional who added the note and the detailed comment of the note.

5.5 Relationships

The relationship data set identifies a defined relationship between two service users e.g. Father/Son. In addition to the relationship type between the two service users other information may also be stored as part of the relationship record including parental responsibility, principal carer indicators etc. Relationships may also identify service users as being members of a family and conversely identify all members of a particular family.

The Protocol CP application uses the relationship information to generate a pictorial representation of the family tree in addition to displaying lists of known children/parents for selection at key process points. The Protocol application will always use two records for each relationship so the relationship can be identified from either party i.e. If a relationship is created between Person A and Person B, the relationship records that would be created would be as follows:

Person A is the FATHER of Person B

Person B is the SON of Person A

5.6 Child Protection Register

The child protection register data set identifies any child who is currently or has previously been on the child protection register. Information will include the start and end date of registrations and the current registration type.

This information is used to identify if a child (or a sibling at the same address) has a current or previous registration during an enquiry process. Further information regarding the child protection registration including the currently responsible key worker, any registration type changes, associated case conference, strategy discussions etc would be queried directly from the core social service system.

5.7 Legal Status / Looked After Child

The legal status data set is a record of any current or previous legal actions taken in respect of a child including looked after child. This will include the start and end date of the legal status, the legal category and the reason for the start and end of the legal status.

5.8 Contacts

The contact data set is a record of any contact made by Social Services with a service user. This will include date and time of the contact, the contact type and outcome, the source of the contact and the professional who processed the contact.

The Protocol CP application will generate a child protection register enquiry contact record when a professional makes an enquiry against the child protection register. The application will also query existing child protection register enquiry contacts to determine if one or more previous enquiries have been made for a child. The application can be configured to make an automatic referral if two or more register enquiries are made for the same child.

5.9 Referrals

The referral data set is a record of any referrals that have been made, as a result of a contact. The referral data set details will include date and time of the referral, referral reason and outcome and the professional who processed the referral. Other family members may also be linked to the referral record.

5.10 Assessments

The assessment data set is a record of any assessments that have been planned or completed. Assessment data will include the planned and completion assessment date and time, assessment reason and outcome and details of the professional(s) responsible for the assessment.

5.11 Child Protection Investigations

The child protection investigation data set is a record of any child protection investigations planned or undertaken by Social Services. The investigation data set will include details on the start, planned and completion date of the investigation, the professionals involved in the investigation and their roles and the children who were considered as part of the investigation.

5.11.1 Strategy Discussions

The strategy discussion data set is a record of completed or planned discussions which were / are to be undertaken as part of a child protection investigation. Data will include details on the completed/planned date, the professionals involved and the agreed outcome.

5.11.2 Interviews

A record of any completed or planned interviews which were / are to be undertaken as part of a child protection investigation. Data will include details on the completed/planned date, the professionals involved, the interview participants and the outcome.

5.12 Child Protection Case Conferences / Reviews / Core Group Meetings

The case conference / meeting data set contains details of any case conferences, reviews or core group meetings that have been planned or completed. This data set will include planned and/or actual conference date, due date, conference location and outcome. The conference data set will also allow access to reports submitted for the conference and the minutes of the conference.

5.12.1 Conference - Linked Professionals

The conference - linked professionals will identify any professionals who were invited to the conference and the professionals who attended the conference. Specific roles for some professionals may also be identified including the conference chairperson, responsible caseworker etc.

5.12.2 Conference - Linked Participants

The conference – linked participants will identify any other non-professional participants who were invited/attended the conference.

5.13 Reviews

The review data set is a record of any (non child protection) reviews that have been planned or completed for a service user. Review data will include the planned and completion review date, the review type and outcome and the professional(s) responsible for the review.

5.14 Involved Professionals

This data set should identify any professionals or other individuals that are involved with a case. Examples would include the responsible caseworker and organisation for a specific service user or family.

5.15 Organisations / Department

The organisations data set will include details on the name, address and contact information for an organisation or department.

5.16 Warnings

This data set includes any warning, hazards or service indicators against a service user, which should be flagged to professionals involved with this service user.

5.17 Offender Register

This data set identifies service users who are currently or have previously been on the child protection offender register. Details will include the type of offence, date of conviction, imprisonment, parole, and release dates.

5.18 Other Information

During the process and integration analysis for each partner, it is possible that additional data sets will be identified other than those listed above. Some of these may not be directly required by the Protocol Child Protection application but are identified as bringing additional benefits for users if the information is accessible from the Protocol CP application. The Bradford Child Protection Pilot identified two additional data sets for integration:

- Paper Files
  This data set identifies the physical location of paper-based files.

- Other References
  Additional references which may assist in the identification of a particular child.

6 Integration Data Set Summary

This section summarises the list of data sets detailed in the previous section. It also gives an indication of where this information is stored in the Bradford Protocol CP implementation and an indication of the location for the West Yorkshire Child Protection system.

It is worth noting that where the core system does not hold certain information (for example CP Investigation Interviews), this information will be retained within the Protocol system.

Data Set	Stored in Protocol and Synchronised with Core System	Read/Written directly to core system
Service User	Yes (1)	Yes (2)
Service User Aliases	Yes (1)	Synchronised
Service User Languages	No	Yes
Service User Security	Yes	Synchronised for Read (3)
Address Gazetteer	Yes (1)	Synchronised for Read (4)
Service User Addresses	Yes (1)	Synchronised
Service User Contact Details	No	Yes
Service User Case Notes	No	Yes
Relationships	Yes	Synchronised
Child Protection Register	Yes (1)	Yes (5) Synchronised for Read
Legal Status / Looked after Child	No (6)	Yes
Contacts (Enquiries)	Yes (1) *	Yes
Referrals	No	Yes
Assessments	No	Yes
CP Investigations	No	Yes
CP Strategy Discussions	No	Yes
CP Investigation Interviews	No	Yes
CP Case Conferences and links	No	Yes
Reviews	No	Yes
Involved Professionals	No	Yes
Organisations	No	Yes - Read Only
Warnings	No	Yes
CP Offender Register	Yes (1)	Yes
Other Information	No	Yes

*Information is stored in Protocol whilst this data is being generated as part of an active child protection process. In some circumstances, it may not be possible to write a specific record to the core system at certain points during the process as all the mandatory information for that core system may not be available. At the end of the process, this information would be read directly from the core system

(1) This information is cached and synchronised by the Protocol application to enable searches to be optimised.

(2) New service users are created simultaneously in both the Protocol and the core social services system.

(3) Social Services core system security information is held by Protocol but will not be changed or written by Protocol. All changes must be made in the core Social Services system. These changes will then be reflected via the synchronisation into the Protocol database.

(4) Address gazetteer entries are synchronised into the Protocol database only. Address entries are not written back to the core Social Services gazetteer. All changes must be made in the core Social Services system. These changes will then be reflected via the synchronisation into the Protocol database.

(5) Basic registration information is held within the Protocol database to enable optimised searching. Detailed information is read directly from the core Social Services database. Any updates or new registration information would be written directly to the core Social Services database.

(6) Legal and looked after status information must be notified to Protocol as this may effect ongoing processes for a child. However, this information is not held within Protocol and will be retrieved directly from the core Social Services system as required.

7 Integration Data Set Details

This section contains the individual data fields which are expected to be included in each data set. It is important to note that the actual field names, the data lengths and the logic data structures will vary between each core system. This is expected and is handled by the Protocol application.

For example, the Protocol logical database schema is designed to cater for multiple languages to be held against each service user. Protocol has been integrated with systems that just use a primary language field held within the main service user table as well as systems that do hold multiple languages.

The following data types are used within these specifications:

Data Type	**Normal Size / Comments**
Number	Numeric field. All numeric fields will typically be 10 length (e.g. 99,999,999)
Date	Date only field
Timestamp	Date and Time field
Varchar	Variable length character field. Comment fields will typically be 2000 characters
Varchar (Picklist)	A coded value used to represent available options for a field. Picklist fields will typically be 1, 4 or 6 characters in length.

7.1 Service User / Person Details

Field Description	Expected Data Type / Size	Comments
Service User ID	Number	Field which uniquely identifies this service user
Service User Category / Classification(s)	Varchar (Picklist)	Type of Service User. Only applicable where multiple person "types" are stored in the same data set as service users e.g. Person, Professional, and Family.
Title	Varchar (Picklist)	
Forenames	Varchar	
Family Name	Varchar	
Gender	Varchar (Picklist)	
Unborn Indicator	Varchar (Picklist)	Flag to indicate if this service user is an unborn child
Date of Birth	Date	Date of birth, Estimated Date of Birth depending upon 'Unborn Indicator'
Birth Certificate Number	Varchar	
Place of Birth from Birth Certificate	Varchar	
NI Number	Varchar	
NHS Number	Varchar	
Unique Pupil Number	Varchar	
Religion	Varchar (Picklist)	
Practicing Religion	Varchar (Picklist)	
Ethnic origin	Varchar (Picklist)	
Immigration Status	Varchar (Picklist)	
Date of Death	Date	
Responsible CSSR	Varchar (Picklist)	
Interpreter Required	Varchar (Picklist)	Yes / No / Unknown

Aliases

Field Description	Expected Data Type	Comments
Alias ID	Number	Field which uniquely identifies this alias
Alias Description	Varchar (Picklist)	Type of alias. E.g. maiden name, alias, previous name etc
Alias Forenames	Varchar	Alias Forename(s)
Alias Family Name	Varchar	Alias Family Name
Start Date	Date	Start date for this alias (if known)
End Date	Date	End Date for this alias (if known)
Comments	Varchar	Additional information regarding this alias
Service User ID	Number	Service User to whom this alias relates (Foreign key to Service User)

Language Usage

Field Description	Expected Data Type	Comments
Language Usage ID	Number	Field which uniquely identifies this language usage
Language Description	Varchar (Picklist)	Language or type of communication
Fluency	Varchar (Picklist)	Fluency in this language
Preferred Language	Varchar (Picklist)	Specifies if this is the preferred language for the service user
Service User ID	Number	Service User to whom this language relates (Foreign key to Service User)

Security

Field Description	Expected Data Type	Comments
Security ID	Number	Field which uniquely identifies this security setting
Security Type	Varchar (Picklist)	Identifies if this security access right is granted or revoked from the specified user
User ID	Number	Unique identifier of the user to whom this security access right relates
Service User ID	Number	Service User to whom this security access right relates (Foreign key to Service User)

7.2 Addresses and Contacts

Address Gazetteer

Field Description	Expected Data Type	Comments
Address ID	Number	Field which uniquely identifies this address
Property Name	Varchar	Name of the property (if present)
Property No	Varchar	House No
Address Line 1	Varchar	First line of address
Address Line 2	Varchar	Second line of address
Town	Varchar	Town
County	Varchar	County
Postcode	Varchar	Postcode
Country	Varchar	Country
Valid Address	Varchar (Picklist)	Is this a current address

Service User Addresses

Field Description	Expected Data Type	Comments
Address Usage ID	Number	Field which uniquely identifies this address usage
Address ID	Number	Address to which this address usage related (Foreign key to Address Gazetteer)
Address Usage Type	Varchar (Picklist)	Address type e.g. Home Address, Care Address, Temporary Address etc
Start Date	Date	Start date for this address location
End Date	Date	End date for this address location
Main Address Indicator	Varchar (Picklist)	Flag to indicate if this is the main address for the service user
Confidential Indicator	Varchar (Picklist)	Flag to indicate if this address is confidential
Telephone No	Varchar	Telephone No relating to this address usage
Telephone Ext No	Varchar	Telephone No Ext relating to this address usage
Service User ID	Number	Service User to whom this address usage relates (Foreign key to Service User)

Contact Details

Field Description	Expected Data Type	Comments
Contact ID	Number	Field which uniquely identifies these contact details
Contact Type	Varchar (Picklist)	Contact type e.g. Home Telephone, Mobile Telephone, Work Telephone
Start Date	Date	Start date for these contact details
End Date	Date	End Date for these contact details
Main Contact Indicator	Varchar (Picklist)	Indicates if this is the main contact number for the service user
Current Contact Indicator	Varchar (Picklist)	Indicates if this is the current contact number for the service user. This would be used when the service user is a temporary location.
Telephone No	Varchar	Telephone No relating to this contact details
Telephone Ext No	Varchar	Telephone No Ext relating to this contact details
Service User ID	Number	Service User to whom these contact details relate (Foreign key to Service User)

7.3 Case Notes

Field Description	Expected Data Type	Comments
Note ID	Number	Field which uniquely identifies this note
Note Type	Varchar (Picklist)	Type of note
Note Date	Date	Date note was created
User ID	Number	User who created note
Note Summary	Varchar	Short note description
Note Details	Varchar	Note comments
Service User ID	Number	Service User to whom this note refers (Foreign key to Service User)

7.4 Relationships

Field Description	Expected Data Type	Comments
Relationship ID	Number	Field which uniquely identifies this relationship record
Relationship Type	Varchar (Picklist)	Description of relationship e.g. Mother, Sister, Son, Wife etc
Service User ID From	Number	Service User to whom this relationship is from (Foreign key to Service User)
Service User ID To	Number	Service User to whom this relationship is to (Foreign key to Service User)
Parental Responsibility Indicatory	Varchar (Picklist)	Indicator to show if the “From” Service User has parental responsibility for the “To” Service User
Principal Carer Indicator	Varchar (Picklist)	Indicator to show if the “From” Service User is a principal carer for the “To” Service User

7.5 Child Protection Register

Field Description	Expected Data Type	Comments
CP Registration ID	Number	Field which uniquely identifies this child protection registration
Registration Start Date	Date	Start date for this registration
Registration End Date	Date	End date for this registration
Registration Type	Varchar (Picklist)	Registration Type e.g. Neglect
Registration Reason	Varchar (Picklist)	Principal reason for the registration of this child
De-registration Reason	Varchar (Picklist)	Reason for the de-registration of this child
User ID	Number	Professional responsible for this registered child
Registration Comments	Varchar	Comments regarding this registration
Registration Conference ID	Number	Conference at which this registration was agreed (Foreign key to CP Conference)
Service User ID	Number	Service User to whom this registration refers (Foreign key to Service User)

7.6 Legal Status / Looked After Child

Field Description	Expected Data Type	Comments
Legal ID	Number	Field which uniquely identifies this child protection registration
Legal Status Start Date	Date	Start date for this registration
Legal Status End Date	Date	End date for this registration
Legal Type	Varchar (Picklist)	Legal status at a specific point in time
Legal Start Reason	Varchar (Picklist)	Reason for the start of this legal status
Legal End Reason	Varchar (Picklist)	Reason for the end of this legal status
Comments	Varchar	Comments relevant to this legal status
Service User ID	Number	Service User to whom this registration refers (Foreign key to Service User)

7.7 Contacts (Enquiries)

Field Description	Expected Data Type	Comments
Contact ID	Number	Field which uniquely identifies this contact
Contact Type	Varchar (Picklist)	Type of contact e.g. Child Protection Register Enquiry
Contact Date	Timestamp	Date and Time of Contact
Reporting Contact Description	Varchar	Name of the person reporting this contact
Reporting Contact ID	Number	Details of the professional reporting this contact (Foreign key to User table)
Reporting Contact Type	Varchar (Picklist)	Details of the agency / type for the person reporting this contact. (e.g. Health Visitor, Education etc)
Reporting Contact Confidential Indicator	Varchar (Picklist)	Specifies if the reporting professional/person should remain confidential or anonymous
Contact Method	Varchar (Picklist)	Method of contact (e.g. phone, letter etc)
Subject Aware Indicator	Varchar (Picklist)	Is the subject for this contact aware that this is being reported to Social Services
Adult Aware Indicator	Varchar (Picklist)	Is a responsible adult aware that this is being reported to Social Services
Contact Priority	Varchar (Picklist)	Priority for this contact
Contact Details	Varchar	Narrative for this contact
Contact Outcome	Varchar (Picklist)	Outcome for this contact
Contact Outcome Details	Varchar	Outcome narrative for this contact
Contact Outcome Date	Timestamp	Date and Time Contact Outcome is recorded
Responsible User ID	Number	Professional responsible for processing this contact (Foreign key to user table)
Responsible Dept ID	Number	Department responsible for processing this contact (Foreign key to Organisation / Department table)
Service User ID	Number	Service User to whom this contact refers (Foreign key to Service User)

Contact Linking

Identifies any other services users linked to a specific contact.

Field Description	Expected Data Type	Comments
Contact Link ID	Number	Field which uniquely identifies this contact link
Contact ID	Number	Contact ID (Foreign key to contact)
Service User	Number	Service User to whom this contact is also linked (Foreign key to Service User)

7.8 Referrals (following a contact)

Field Description	Expected Data Type	Comments
Referral ID	Number	Field which uniquely identifies this referral
Contact ID	Number	Contact ID to whom this referral is related (Foreign key to Contact)
Referral Type	Varchar (Picklist)	Type of referral
Referral Status	Varchar (Picklist)	Status of this referral
Referral Date	Timestamp	Date and Time of referral
Referral Reason	Varchar (Picklist)	Reason for this referral
Referral Outcome	Varchar (Picklist)	Outcome for this referral
Referral Outcome Details	Varchar	Outcome narrative for this referral
Referral Outcome Date	Timestamp	Date and Time Referral Outcome is recorded
Referral Close Date	Timestamp	Date and Time referral is closed
Responsible User ID	Number	Professional responsible for processing this referral (Foreign key to user table)
Responsible Dept ID	Number	Department responsible for processing this referral (Foreign key to Organisation / Department table)
Service User ID	Number	Service User to whom this referral refers (Foreign key to Service User)

7.9 Assessments

Assessment Details

Field Description	Expected Data Type	Comments
Assessment ID	Number	Field which uniquely identifies this assessment
Referral ID	Number	Referral ID to whom this assessment is related (Foreign key to Referrals)
Assessment Type	Varchar (Picklist)	Assessment Type
Planned Start Date	Date	Assessment planned start date
Planned End Date	Date	Assessment planned end date
Actual Start Date	Date	Assessment actual start date
Actual End Date	Date	Assessment actual end date
Assessment Priority	Varchar (Picklist)	Assessment Priority
Assessment Status	Varchar (Picklist)	Assessment Status
Delay Reason	Varchar (Picklist)	Reason why assessment has been delayed
Comments	Varchar	Assessor comments
Recipient Comments	Varchar	Comments made by the assessor
Carer Comments	Varchar	Comments made by the carer
Assessment Outcome	Varchar (Picklist)	Outcome of the assessment
Service User ID	Number	Service User to whom this assessment refers (Foreign key to Service User)

Assessment Linked Professionals / Service Users / Organisations & Department

Identifies any professionals, agencies, departments or other service users involved in this assessment

Field Description	**Expected Data Type**	**Comments**
Assessment Link ID	Number	Field which uniquely identifies this assessment
Assessment ID	Number	Assessment to which this link refers (Foreign key to assessment)
User ID	Number	Professional linked to this assessment (Foreign key to user table)
Service User ID	Number	Other Person linked to this assessment (Foreign key to Service User)
Department ID	Number	Department or agency linked to this assessment (Foreign key to Organisation / Department table)
Role	Varchar (Picklist)	Capacity in which this professional / person is linked to this assessment
Involvement Start Date	Date	Date at which this person became involved
Involvement End Date	Date	Date at which this person stopped being involved
Comments	Varchar (Picklist)	Comments

7.10 Child Protection Investigations

Investigation Details

Field Description	Expected Data Type	Comments
Investigation ID	Number	Field which uniquely identifies this investigation
Referral ID	Number	Referral ID to whom this assessment is related (Foreign key to Referrals)
Investigation Type	Varchar (Picklist)	Investigation Type
Planned Start Date	Date	Investigation planned start date
Planned End Date	Date	Investigation target end date
Actual Start Date	Date	Investigation actual start date
Actual End Date	Date	Investigation actual end date
Investigation Status	Varchar (Picklist)	Investigation Status
Comments	Varchar	Comments
Investigation Outcome	Varchar (Picklist)	Outcome of the investigation

Investigation Service User Recipients

Identifies the service users who are/were considered as part of this child protection investigation

Field Description	Expected Data Type	Comments
Investigation Link ID	Number	Field which uniquely identifies this investigation link
Investigation ID	Number	Investigation to which this link refers (Foreign key to child protection investigation)
Service User ID	Number	Service User linked to this investigation (Foreign key to Service User)
Comments	Varchar	Comments

Investigation Linked Professionals / Service Users / Organisations & Department

Identifies any professionals, agencies, departments or other service users involved in this investigation

Field Description	Expected Data Type	Comments
Investigation Link ID	Number	Field which uniquely identifies this investigation link
Investigation ID	Number	Investigation to which this link refers (Foreign key to child protection investigation)
User ID	Number	Professional linked to this child protection investigation (Foreign key to user table)
Service User ID	Number	Other Person linked to this child protection investigation (Foreign key to Service User)
Department ID	Number	Department or agency linked to this child protection investigation (Foreign key to Organisation / Department table)
Role	Varchar (Picklist)	Capacity in which this professional / person is linked to this child protection investigation
Involvement Start Date	Date	Date at which this person became involved
Involvement End Date	Date	Date at which this person stopped being involved
Comments	Varchar	Comments

Strategy Discussions

Identifies strategy discussions linked to child protection investigations

Field Description	Expected Data Type	Comments
Strategy ID	Number	Field which uniquely identifies this strategy discussion
Investigation ID	Number	Investigation to which this strategy discussion is linked (Foreign key to child protection investigation)
Strategy Discussion Date	Date	Date of the strategy discussion
Strategy Discussion Status	Varchar (Picklist)	Strategy discussion status
Strategy Discussion Category	Varchar (Picklist)	Strategy discussion category (e.g. Phone discussion, meeting discussion etc)
Location	Varchar (Picklist)	Location of the strategy discussion meeting (if applicable)
Comments	Varchar	Comments regarding the strategy group meeting Further details regarding the decisions made within the strategy discussion may be documented and stored in an external electronic document attached to this strategy discussion record. As part of entering the details of the strategy discussion, the Protocol system will allow interviews etc to be arranged.

Strategy Discussion Linked Professionals

Identifies the professionals involved in the strategy discussions

Field Description	Expected Data Type	Comments
Strategy Link ID	Number	Field which uniquely identifies this strategy discussion link
Strategy ID	Number	Strategy discussion to which this link refers (Foreign key to strategy discussion)
User ID	Number	Professional linked to the strategy discussion (Foreign key to user table)
Role	Varchar (Picklist)	Capacity in which this professional is linked to this child protection strategy discussion
Comments	Varchar	Comments
Strategy Discussion Date	Date	Date of the strategy discussion

Interview Details

Field Description	Expected Data Type	Comments
Interview ID	Number	Field which uniquely identifies this interview
Investigation ID	Number	Investigation to which this interview is linked (Foreign key to child protection investigation)
Interview Type	Varchar (Picklist)	Interview Type (e.g. Child Interview, Child Interview with Parent present, Parent Interview etc)
Planned Interview Date	Timestamp	Planned interview date
Actual Interview Date	Date	Actual Interview date
Interview Status	Varchar (Picklist)	Interview Status (e.g. planned, completed, cancelled)
Delay Reason	Varchar (Picklist)	Reason for delay or cancellation
Location	Varchar	Location for the interview
Interview Description	Varchar	Reason / Purpose for the interview
Comments	Varchar	Comments Further details containing either a transcript or notes from the interview may be documented and stored in an external electronic document attached to this interview record.

Interview Service User Recipients

Identifies the service users who are/were participants of this child protection interview

Field Description	Expected Data Type	Comments
Interview Link ID	Number	Field which uniquely identifies this interview link
Interview ID	Number	Investigation to which this interview relates (Foreign key to Interview Details)
Service User ID	Number	Service User linked to this interview (Foreign key to Service User)
Comments	Varchar	Comments

Interview Linked Professionals

Identifies any professionals involved in this interview

Field Description	Expected Data Type	Comments
Interview Link ID	Number	Field which uniquely identifies this interview link
Interview ID	Number	Interview to which this link refers (Foreign key to child protection interview)
User ID	Number	Professional linked to this child protection interview (Foreign key to user table)
Role	Varchar (Picklist)	Capacity in which this professional / person is linked to this child protection interview
Attended Indicator	Varchar (Picklist)	Indicates if the invited professional attended or sent apologies for the interview
Comments	Varchar	Comments

7.11 Child Protection Case Conferences / Reviews / Core Group Meetings

Meeting Details

Field Description	Expected Data Type	Comments
Meeting ID	Number	Field which uniquely identifies this meeting
Meeting Type	Varchar (Picklist)	Type of meeting (e.g. Conference, Review, Core Group)
Meeting Status	Varchar (Picklist)	Status of meeting (e.g. To be planned, scheduled, completed, cancelled etc)
Due Date	Date	Date by which the meeting is due
Planned Date	Timestamp	Planned meeting date and time
Actual Date	Date	Actual meeting date
Delay Reason	Varchar (Picklist)	Reason why meeting is late or has been cancelled
Location	Varchar	Location of the meeting
Outcome Comments	Varchar	Comments regarding the outcome of the meeting that can be distributed to invited professionals prior to the meeting minutes. The minutes of the meeting will document the details, outcome and actions from the meeting and will be stored in an external document attached to this meeting record
Minutes Distribution Date	Date	Date the minutes were distributed (this will be the date the minutes document is electronically sent via Protocol to professionals on the system)
Comments	Varchar	Any other comments relevant to this meeting
Responsible Professional User ID	Number	Professional responsible for requesting the meeting (Foreign key to user table)
Responsible Agency/Department ID	Number	Department/Agency of the professional responsible for requesting the meeting (Foreign key to Organisation / Department table)

Meeting Service User Recipients

Identifies the service users (children and parents) who were invited/attended the conference

Field Description	Expected Data Type	Comments
Meeting Link ID	Number	Field which uniquely identifies this meeting link
Meeting ID	Number	Case Conference / Review / Core Group to which this link refers (Foreign key to child protection case conference)
Service User ID	Number	Service User linked to this meeting (Foreign key to Service User)
Comments	Varchar	Comments
Attendance Indicator	Varchar (Picklist)	Indicator to show if invited recipient attended conference
Meeting Outcome	Varchar (Picklist)	Decision of the meeting in respect of this service user (only applicable for children)

Meeting Linked Professionals / Organisations & Department

Identifies any professionals who have been invited/attended the meeting including the chair of the conference

Field Description	Expected Data Type	Comments
Meeting Link ID	Number	Field which uniquely identifies this meeting link
Meeting ID	Number	Case Conference / Review to which this link refers (Foreign key to child protection case conference)
User ID	Number	Professional invited to attend this conference (Foreign key to user table)
Department ID	Number	Department invited to attend this conference (Foreign key to Organisation / Department table)
Role	Varchar (Picklist)	Capacity in which this professional / person is linked to this conference
Report Submitted Date	Date	Date at which this person became involved
Attended Indicator	Varchar (Picklist)	Indicates if the invited professional attended or sent apologies for the meeting
Comments	Varchar	Comments

7.12 Reviews

Field Description	Expected Data Type	Comments
Review ID	Number	Field which uniquely identifies this review
Review Type	Varchar (Picklist)	Type of review
Review Status	Varchar (Picklist)	Status of review (e.g. To be planned, scheduled, completed, cancelled etc)
Due Date	Date	Date by which the review is due
Planned Date	Timestamp	Planned review date and time
Actual Date	Date	Actual review date
Delay Reason	Varchar (Picklist)	Reason why review is late or has been cancelled
Review Outcome	Varchar (Picklist)	Outcome of the review
Comments	Varchar	Comments regarding this review
Responsible Professional User ID	Number	Professional responsible for requesting the review (Foreign key to user table)
Responsible Agency/Department ID	Number	Department/Agency of the professional responsible for the review (Foreign key to Organisation / Department table)
Service User ID	Number	Service User linked to this review (Foreign key to Service User)

7.13 Involved Professionals

Field Description	Expected Data Type	Comments
Involved ID	Number	Field which uniquely identifies this Involved Professional record
Involved Professional User ID	Number	Professional responsible for requesting the review (Foreign key to user table)
Involved Agency/Department ID	Number	Department/Agency of the professional responsible for the review (Foreign key to Organisation / Department table)
Role	Varchar (Picklist)	The capacity in which this professional is involved
KeyWorker Indicator	Varchar (Picklist)	Flag to indicate if this professional is the KeyWorker for this Service User
Start Date of Involvement	Date	Date when this professionals involvement with this Service User started
End Date of Involvement	Date	Date when this professionals involvement with this Service User ended
Service User ID	Number	Service User whom this involvement concerns (Foreign key to Service User)

7.14 Organisations / Departments

Field Description	Expected Data Type	Comments
Department ID	Number	Field which uniquely identifies this organisation / department
Department Name	Varchar	Name of the organisation / department
Department Type	Varchar	Type of Organisation / Department (e.g. Social Services, Health PCT, Police etc)
Department Abbreviation	Varchar	Abbreviated organisation / department name
Department Status	Varchar (Picklist)	Indicates if the organisation / department is currently used and if they have access to Protocol
Department Address	Varchar	Address
Department Telephone	Varchar	Telephone Number
Department Email	Varchar	Email address
Parent Department ID	Number	Field which indicates the responsible organisation / department for this organisation / department

7.15 Warnings

Field Description	Expected Data Type	Comments
Warning ID	Number	Field which uniquely identifies this warning
Warning Type	Varchar (Picklist)	Type of warning (e.g. potentially violent person, contact CPU etc)
Warning Date	Date	Date warning was added
End Date	Date	Date warning was cancelled
Comments	Varchar	Comments relating to this warning
Responsible Professional	Number	Professional who added the warning (Foreign key to user table)
Service User ID	Number	Service User linked to this warning (Foreign key to Service User)

7.16 Offender Register

Field Description	Expected Data Type	Comments
Offender Register ID	Number	Field which uniquely identifies this offence registration
Offence Type	Varchar (Picklist)	Type of offence
Registration Status	Varchar (Picklist)	Status of this registration
Registration Date	Date	Date added to register
Imprisonment Date	Date	Date of imprisonment
Release Date	Date	Date of release
Comments	Varchar	Comments relating to this offender registration
Service User ID	Number	Service User to whom this registration applies (Foreign key to Service User)

Appendix A – Swift GovTalk XML Message Example

Example GovTalk message to create a person in Swift:

```
<?xml version="1.0" encoding="UTF-8"?>
<GovTalkMessage xmlns="http://www.govtalk.gov.uk/CM/envelope"
xmlns:dsig="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.govtalk.gov.uk/CM/envelope envelope-v2.0.xsd
 http://www.w3.org/2000/09/xmldsig# xmldsig-core-schema.xsd">
    <EnvelopeVersion>2.0</EnvelopeVersion>
    <Header>
        <MessageDetails>
            <Class>Sxapi</Class>
            <Qualifier>request</Qualifier>
            <CorrelationID>2</CorrelationID>
        </MessageDetails>
        <SenderDetails>
            <IDAuthentication>
                <SenderID>userfd</SenderID>
                <Authentication>
                    <Method>clear</Method>
                    <Value>dXNlcmZk</Value>
                </Authentication>
            </IDAuthentication>
        </SenderDetails>
    </Header>
    <GovTalkDetails/>
    <Body>
        <Sxapi xmlns="http://www.aniteps.com/schemas/swift/sxapi"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.aniteps.com/schemas/swift/sxapi sxapi-10.xsd"
SchemaVersion="1.0">
            <InsertPersonDetails>
               <PersonOrFamily>person</PersonOrFamily>
               <CitizenNameForename>JOE</CitizenNameForename>
               <CitizenNameSurname>BLOGGS</CitizenNameSurname>
               <Initials>T</Initials>
               <TitleCode>MR</TitleCode>
               <CitizenNameTitle>Mister</CitizenNameTitle>
               <CitizenSex>1</CitizenSex>
               <MaritalStatusCode>M</MaritalStatusCode>
               <EthnicOriginCode>01</EthnicOriginCode>
               <EthnicSelfDefined>no</EthnicSelfDefined>
               <BirthDate>1948-06-11</BirthDate>
               <SpeechImpairment>no</SpeechImpairment>
               <HearingImpairment>no</HearingImpairment>
               <VisualImpairment>no</VisualImpairment>
               <InterpreterRequired>yes</InterpreterRequired>
               <AdvocateRequired>yes</AdvocateRequired>
               <AdvocateSupplied>yes</AdvocateSupplied>
               <SSClientIndicator>yes</SSClientIndicator>
               <WorkerIndicator>no</WorkerIndicator>
               <ResourceIndicator>no</ResourceIndicator>
               <PlaceOfBirth>Buckingham</PlaceOfBirth>
               <NationalityCode>BRIT</NationalityCode>
               <RefugeeStatusCode>AM</RefugeeStatusCode>
               <HomeOfficeNumber>123456AA</HomeOfficeNumber>
               <HealthID>654321ZZ</HealthID>
               <PropertyAccess>Enter via back door</PropertyAccess>
               <EmailAddress>abc@1234</EmailAddress>
               <LivesAlone>yes</LivesAlone>
               <PayChildInFCP>no</PayChildInFCP>
               <SPClientIndicator>no</SPClientIndicator>
               <SPLandlordIndicator>no</SPLandlordIndicator>
               <SPResourceIndicator>no</SPResourceIndicator>
            </InsertPersonDetails>
        </Sxapi>
    </Body>
</GovTalkMessage>
```

Appendix B – Pixel SOAP Server WSDL example

Example WSDL describing connection to the Pixel SOAP Server:

```
<?xml version='1.0' encoding='UTF-8' ?>
 <!--  Generated  02/10/04  by  Microsoft  SOAP  Toolkit  WSDL  File  Generator,  Version
1.02.813.0 -->
<definitions  name ='CallScnPool'   targetNamespace = 'http://tempuri.org/wsdl/'
         xmlns:wsdlns='http://tempuri.org/wsdl/'
         xmlns:typens='http://tempuri.org/type'
         xmlns:soap='http://schemas.xmlsoap.org/wsdl/soap/'
         xmlns:xsd='http://www.w3.org/2001/XMLSchema'
         xmlns:stk='http://schemas.microsoft.com/soap-toolkit/wsdl-extension'
         xmlns='http://schemas.xmlsoap.org/wsdl/'>
  <types>
    <schema targetNamespace='http://tempuri.org/type'
      xmlns='http://www.w3.org/2001/XMLSchema'
      xmlns:SOAP-ENC='http://schemas.xmlsoap.org/soap/encoding/'
      xmlns:wsdl='http://schemas.xmlsoap.org/wsdl/'
      elementFormDefault='qualified'>
      <complexType  name ='ArrayOfstring'>
        <complexContent>
          <restriction base='SOAP-ENC:Array'>
            <attribute ref='SOAP-ENC:arrayType' wsdl:arrayType='string[]'/>
          </restriction>
        </complexContent>
      </complexType>
    </schema>
  </types>
  <message name='ScnFuncWrap.CallScnPool'>
    <part name='logicalApplication' type='xsd:string'/>
    <part name='logicalFunction' type='xsd:string'/>
    <part name='parameters' type='typens:ArrayOfstring'/>
  </message>
  <message name='ScnFuncWrap.CallScnPoolResponse'>
    <part name='Result' type='xsd:string'/>
    <part name='parameters' type='typens:ArrayOfstring'/>
  </message>
  <portType name='ScnFuncWrapSoapPort'>
    <operation  name='CallScnPool'  parameterOrder='logicalApplication  logicalFunction
parameters'>
      <input message='wsdlns:ScnFuncWrap.CallScnPool' />
      <output message='wsdlns:ScnFuncWrap.CallScnPoolResponse' />
    </operation>
  </portType>
  <binding name='ScnFuncWrapSoapBinding' type='wsdlns:ScnFuncWrapSoapPort' >
    <stk:binding preferredEncoding='UTF-8'/>
    <soap:binding style='rpc' transport='http://schemas.xmlsoap.org/soap/http' />
    <operation name='CallScnPool' >
      <soap:operation    soapAction='http://tempuri.org/action/ScnFuncWrap.CallScnPool'
/>
      <input>
        <soap:body use='encoded' namespace='http://tempuri.org/message/'
                 encodingStyle='http://schemas.xmlsoap.org/soap/encoding/' />
      </input>
      <output>
        <soap:body use='encoded' namespace='http://tempuri.org/message/'
                 encodingStyle='http://schemas.xmlsoap.org/soap/encoding/' />
      </output>
    </operation>
  </binding>
  <service name='CallScnPool' >
    <port name='ScnFuncWrapSoapPort' binding='wsdlns:ScnFuncWrapSoapBinding' >
      <soap:address location='http://ED/QDXScreenRejuvenator/CallScnPool.wsdl' />
    </port>
  </service>
</definitions>
```