KNOTION
the smart way forward

EA Deliverable

Rwanda Development Board

JRLOS Full Technology Solution Architecture

Version 1.0

RDB RWANDA DEVELOPMENT BOARD

Document History

Ver.	Date	Author	Reviewed by	Date	Validated by	Date

Version Control

Date	Version	Object of revisions

Related Documents

Document name	Document version

Approval

The signatories hereof, being duly authorised thereto, by their signatures hereto authorise the execution of the work detailed herein, or confirm their acceptance of the contents hereof and authorise the implementation/adoption thereof, as the case may be, for and on behalf of the parties represented by them.

Knotion: Director Mr Marius SNEL	Date
Rwanda Development Board: Project Owner Mrs Rita KAMANZI	Date

Table of Contents

List of Figures

1. Technology Architecture Definition

The purpose of this section is to provide a solution view of the Technology Architecture for the IECMS project. The specific objective of this Solution Architecture was to go out on a Request For Proposal (RFP) for the provisioning of a system complying to this developed architecture definition.

This whole section must be read in conjunction with and reference to the Technology Segment Architecture for the JRLOS domain performed in the previous Phase 2 of this project.

Various levels of detail and abstraction are supported in this report:

- The most important aspects from a Solution perspective are provided in architectural diagrams.
- Each of the objects referenced in the architectural diagrams are explained and requirements and/or constraints are specified where applicable. (This layer of detail may be reflected in this specific version of this document or omitted in summarised type documents).
- The full spectrum of technology components, which were investigated in the technology architecture, is provided in the Usage catalogue with various classifications and categorisations of usage, importance and standardisation. The description of all these components is provided in a separate Technology Reference Model (TRM) deliverable from a previous phase.
- The diagrams only address aspects viewed as important from a pre-RFP perspective regarding the acquisition or development of the required solution. More detailed design and/or architecture can only be done once a post-RFP decision has been taken on a specific system.
- Various other lower level technology components (e.g. anti-virus, firewalls, server/network/RDBMS monitoring software, etc.) are assumed to be in place (or put into place) as part of the normal efficient day-to-day operation of the infrastructure (e.g. ITIL).
- The eventual as-built Solution Architecture may include various other components, from the usage catalogue, in the diagrams, to support various other viewpoints of the acquired system.

Various options are still open on this level of definition to still accommodate different potential solutions, whether re-used, bought as COTS or developed from scratch (in-house or out-sourced) or any combination of these. The 'ideal solution' can be deduced from the preference sequences defined for each component where the options are left open.

1.1. Solution Overview

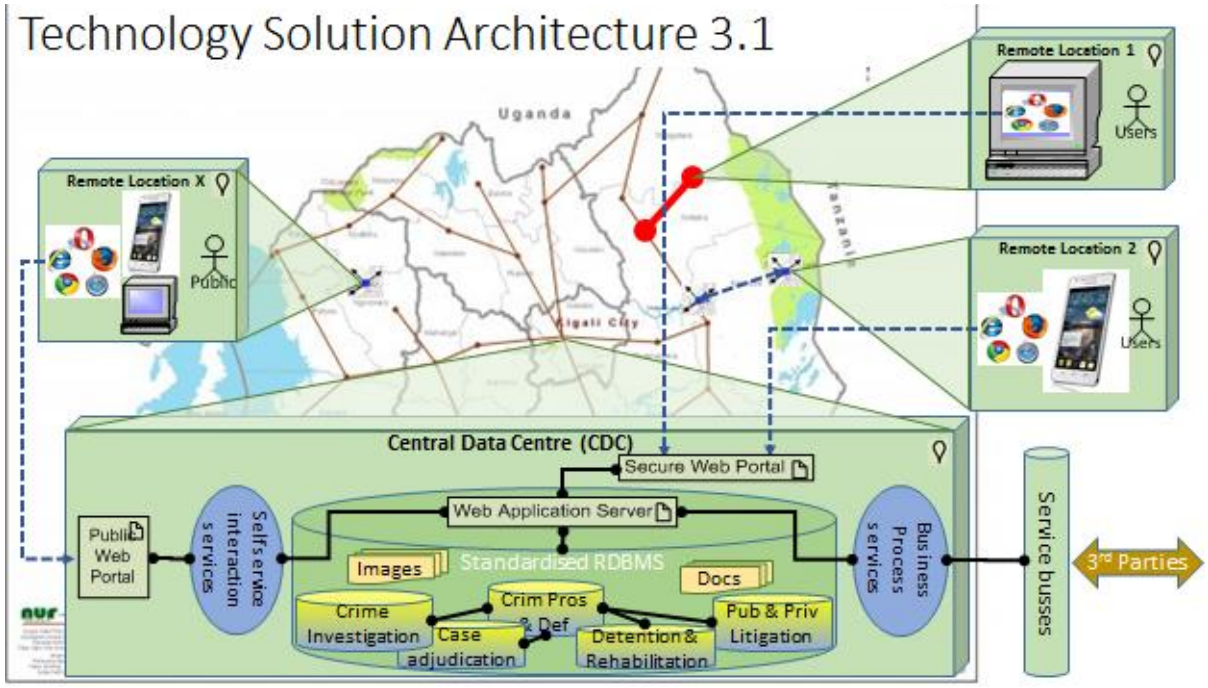
The following picture depicts a more pictorial high-level executive management overview of the proposed Technology Solution Architecture. (The equivalent of an 'artist's impression' in normal architectural terms).

The optimal architecture can be summarised on a high level as follows:

- Remote sites need to be WAN connected to the National Backbone or optionally via VPN on cell phone infrastructure
- The data, images and documents need to be hosted in preferably one (or many data integrated) central database(s), hosted in one standardised RDBMS on virtual clustered servers at a Centralised Data Centre
- Information access is provided to any device, anytime, anywhere via simple web browsers
- Various levels of business services are provided to various types of external parties via SOA-based web services
- The biggest challenge in establishing this architecture is indicated pictorially with the red connector; to get the outlaying remote areas connected to the NBB and/or the CDC

Various alternative options are proposed on more detailed levels in the rest of the architecture.

Figure 1: Solution Overview



1.2. Server Architecture

This section describes the server architecture for the various types of servers in more detail. It also provides a service interaction overview with the end-user workstation.

1.2.1. Server Architecture Diagram

The associated diagram depicts the generic technology patterns for the different server nodes (hosting platforms) in context of the exposed services, particularly to the end-user workstations.

Emphasis is placed on different server types and configurations due to the uncertainty at the time of compiling the architecture whether and exactly which components of the server architecture will need to be supplied by the successful vendor as part of the overall solution provisioning and which components will be the responsibility of the GoR under separate contracts.

The detailed server requirements will need to be addressed in the detail design of the solution.

All servers will be suitably protected, including (but not limited to) the following mechanisms:

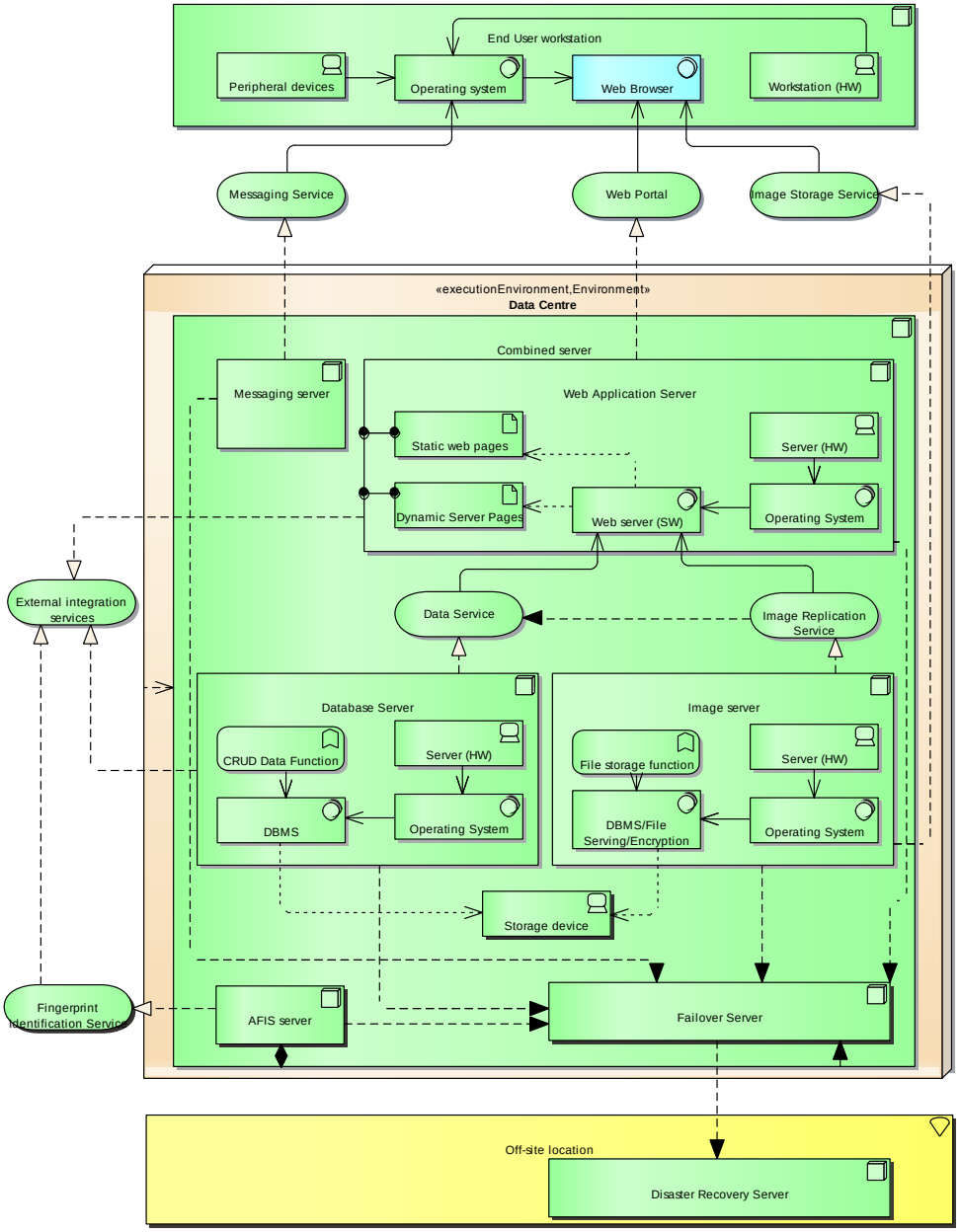
- Operating system and network (logical) access control and management
- Anti-virus, fire-wall and spyware protection

- Patch management
- Configuration management
- Protection against system administrator misuse and ensure segregation of duty
- Audit logging
- Protection against environmental risks such as fire, water and theft
- Physical access control
- A comprehensive set of system management tools (e.g. remote management software; monitoring software – to monitor the state of the workstation)
- Software discovery tools – to monitor and discover unwanted software installed on the workstations

The server management should provide for load balancing, fault tolerance, scalability, capacity management, backup/recovery, redundancy and disaster recovery; and a logical access control policy should be established, documented, and reviewed based on business and security requirements. The policy will serve as a minimum baseline to adhere to.

(All these preceding aspects are assumed as normal day-to-day operational best practice of the infrastructure and platform technology and are thus not described in great detail anywhere in this (acquired Application supporting) Technology Solution Architecture document.)

Figure 2: Server Architecture Diagram



1.2.1.1. Data Centre (Environment)

Although Data Centre infrastructure is normally not strictly seen as part of Enterprise or Technology Architecture, this component is added due its importance in this situation.

A ministerial instruction has been issued which forces all government departments to utilise the National Back Bone fibre optic network and host their solutions at the National Data Centre at Telecom House in Kigali.

Due to the specific security requirements for Judicial information and the recent changes in ownership of the NDC, an exception may be allowed in this case and a separate Data Centre may be required on the MINIJUS campus.

1.2.1.1.1. Combined server (Node)

Dependent on data volumes, performance, connectivity bandwidth and various other factors the different depicted logical servers can be combined into a single physical or virtual server.

Dependent on the same factors, multiple instances of the various logical servers may also be required, independent on whether they are physically distributed, even in various regions, or not. Replication and/or synchronisation services should then be provided between the different server instances.

Requirement: Standardisation - Want - *GoR/V*

TR11: Valid for all servers: No specific manufacturer is preferred.

1.2.1.1.1.1 Database Server (Node)

The Database server node, containing all the technology components to realise the required services.

1.2.1.1.1.1.1. Server (HW) (Device)

If required a separate physical/virtual/clustered Database server can be used, but for the anticipated amount of data and the number of users, the same physical/virtual/clustered server as the Web application server can most probably be used.

Requirement: Provisioning - Need - *V*

TR02: Server HW capability to host the RDBMS of choice will be supplied.

1.2.1.1.1.1.2. DBMS (SystemSoftware)

Dependent on data volumes, performance, connectivity bandwidth and various other design factors, multiple database schemas and/or instances on a single or even multiple servers can be deployed.

The preference is to have a single database schema with fully defined role-based user security, with each user having its own log-in to the database, instead of one generic application user defined.

Requirement: Functional - Need - *V*

TR04: The solution will employ a RDBMS for Master Data Management.

Rationale:

Even though this may be an assumed requirement, no older (e.g. file system/ISAM type) or newer (e.g. big data/ HANA/ NON-SQL type) data storage schemes will be allowed for the Master Data Management.

A balance point should be achieved on the current stable technologies without exerting undue pressure on limited suitably skilled resources with bleeding or trailing edge technologies.

Requirement: Standardisation - Want - *V*

TR05: The RDBMS of choice is the latest stable versions of Oracle or Microsoft SQL Server

Rationale:

Current investment in skills and licenses for MSS and it should cater for estimated data volumes and number of users.

Oracle would be preferred due to superior functionality, performance and leadership position - Refer to Gartner Magic quadrant for RDBMS. Some skill and license investment in other GOR departments.

Selecting other RDBMS vendors will lower the economies of scale regarding available skills to a critical point.

Requirement: Functional - Want - *V*

TR06: If multiple RDBMS schemas, instances or installations are deployed, data integration will be performed on the database level.

Rationale:

Proper RDBMS design must be utilised to ensure all data requirements

and principles are being complied to. Database links, views and automated processes and triggers should be used to ensure data integrity, security, currency, etc.

Requirement: Functional - Want - *V*

TR07: Full role-based user security will be implemented in the RDBMS

Rationale:

To ensure that the database is a 'self-sustained universe' to ensure that any user (normal, support, admin, super-user, developer, etc.) using any other 3rd party toolsets are subject to the same security rules than when accessing the database through the application.

Additional security can be implemented in the application for whatever other purposes.

Requirement: Functional - Want - *V*

TR08: All business logic will be implemented in the RDBMS as far as practically possible.

Rationale:

To ensure that the database is a 'self-sustained universe' to ensure that any user (normal, support, admin, super user, developer, etc.) using any other 3rd party toolsets are subject to the same business rules (e.g. data type restrictions, validations, foreign key constraint, CRUD constraints, business rules, allowable values, initial values, etc.) than when accessing the database through the application.

Additional (non-core) validations, integrity constraints, format prescriptions, etc. can be implemented in the application for whatever other purposes.

1.2.1.1.1.1.3. CRUD Data Function (InfrastructureFunction)

The full data Create, Read, Update, Delete (CRUD) functions, which realises the associated service.

1.2.1.1.1.2 Web Application Server (Node)

The server node containing all the required components to realise the required web based services, whether static, dynamic or web services.

1.2.1.1.1.2.1. Operating System (SystemSoftware)

All logical server Operating Systems will be hardened to strengthen the security posture and minimise the vulnerability landscape of these systems (e.g. remove unnecessary services and applications).

Host hardening includes (but are not limited to):

- Removing unwanted services
- Installing system patches, updates and fixes for the OS and applications
- Changing default passwords for system accounts
- Changing default communication ports (if possible)
- Replacing insecure or vulnerable services
- Removing sensitive information from system and application messages and banners
- Enabling system-wide auditing
- Monitoring file and directory access of sensitive information

(These aspects are also seen as day-to-day operational best practice and are not elaborated on in much more detail elsewhere.)

Requirement: Standardisation - Preference - *GoR/V*

TR12: Valid for all servers: Microsoft Windows Server is currently most prevalent, but any Open Source operating systems can be proposed.

Rationale:

Most current baseline servers are running Windows Server from Microsoft and most competency investment is already done on this technology.

Even though not formalised at the time of compiling this architecture, actions have been identified and launched to investigate the usage of Open Source operating systems instead.

Dependent on the services provided by the server (e.g. RDBMS), these system software component choices (e.g. Oracle) may influence the choice of OS (e.g. Unix/Linux).

1.2.1.1.1.2.2. Web server (SW) (SystemSoftware)

The Web server software serving the static or dynamic web pages, e.g. IIS, Tomcat, JBoss, Websphere.

No specific standardisation is required here, but some of the current systems use JBoss and the choice will mainly depend on functionality, performance, number of users, load balancing, underlying OS, etc. requirements and design.

1.2.1.1.1.2.3. Static web pages (Artifact)

e.g. HTML, CSS, JS, etc. artifacts

1.2.1.1.1.2.4. Dynamic Server Pages (Artifact)

e.g. .JSP, .ASP, .ASPX, .PHP, etc. artifacts

1.2.1.1.1.3 Image server (Node)

Server handling all images in various different formats, e.g. .docx, .xls, .tif, .pdf, .jpg, .mp3/4, .avi, .mpeg, .wmv, etc.

1.2.1.1.1.3.1. DBMS/File Serving/Encryption (SystemSoftware)

This component can be in the form of (in order of preference):

1. RDBMS
2. object storage
3. file serving
4. own published encryption/compression algorithms
5. proprietary storage mechanisms

Requirement: Performance - Preference - *V*

TR03: Even though no specific technique for storing of

images/documents is prescribed, storage as blob data types in the RDBMS is preferred.

1.2.1.1.1.3.2. File storage function (InfrastructureFunction)

The full image/file Save, Read, Update, Delete functions, which realises the associated service.

1.2.1.1.1.4 Storage device (Device)

No specific storage device concept is prescribed, as it is dependent on the detail design of the image handling function/service, as well as the overall redundancy requirements of the failover server, disaster recovery server and normal backup component designs.

The preference between some of the components is as follows:

1. Direct-attached storage (DAS) - also utilising SSD - especially for RDBMS storage components. (Where multiple physical disks are also required for separated storage of data and indexes)
2. Solid-state drives (SSD) - especially for RDBMS storage components. (Where multiple physical disks are also required for separated storage of data and indexes)
3. Hybrid Redundant Array of Independent Disks (RAID) (utilising SSD) - for required level of redundancy required in conjunction with failover server and disaster recovery server designs
4. Traditional RAID
5. Network-attached storage (NAS) - should images be stored as objects or files, outside the RDBMS
6. Storage area network (SAN) - dependent on the failover server, disaster recovery server and normal backup component designs

(Most of these are described in the TRM deliverable).

1.2.1.1.1.5 Messaging server (Node)

This server provides functionality for messaging and notifications, typically e-mail, Short Messaging Service (SMS) or any other protocol type which become relevant.

1.2.1.1.1.6 Failover Server (Node)

This server is not depicted in significant detail but needs to cater for all required back-up and failover for all required functions of other servers.

Requirement: Performance - Want - *GoR/V*

TR09: 99% availability of infrastructure will be maintained.

1.2.1.1.1.7 AFIS server (Node)

This server provides functionality for the identification and verification of persons through their fingerprints. Its detail composition will typically be the same as all other servers, with very specific additional components for the handling and storing of fingerprint images and their analysis.

Even though depicted here in concept as part of this solution architecture, it may be handled as a separate solution, due to its specialised nature and possible back-record conversion requirements.

1.2.1.1.1.8 Data Service (InfrastructureService)

A service to provide for the full CRUD (Create, Read, Update, Delete) functionality of structured enterprise Master Data.

If data needs to be replicated between distributed servers, due to whatever performance or functional requirements, this service also needs to cater for this service.

Should the preferred design concept, of storing all images in the database, be implemented, this data service also needs to cater for the CRUD handling of the blob data type.

1.2.1.1.1.9 Image Storage Service (InfrastructureService)

A storage service to handle the storage of all potential file types of all different encodings. e.g. biometric images, photographs, office automation documents, scanned images, TIFF, PDF, audio/video files (different codecs), XML/XMI/JSON/Binary data files, ASCII, Comma Separated Values (CSV), etc.

1.2.1.1.1.10 *Image Replication Service (InfrastructureService)*

A service to replicate images from distributed image servers to a central server, if required due to performance or functional design requirements.

1.2.1.1.1.11 *Messaging Service (InfrastructureService)*

Simple Notification/Messaging Service: Could include pushing directly to mobile devices, SMS text message, e-mail, simple messaging queues or to any HTTP end-point.

1.2.1.1.1.12 *Fingerprint Identification Service (InfrastructureService)*

An Identification and Verification service provided by the AFIS server to identify or verify fingerprints and link it to a specific person.

The Identification service may be consumed as part of crime scene investigation, criminal record checks, etc. where only a fingerprint image is available.

The Verification service may be consumed for the verification of a person's identity where another form of identification is already available (for example in prison, property searches, criminal record clearance certificates, etc.)

In the future this service may also be consumed externally for example for passport applications, driver's license applications, job candidate clearance, etc.

1.2.1.1.1.13 *External integration services (InfrastructureService)*

Services provided for integration to external services and systems, typically to systems running in other Government (GoR) departments or to external 3rd parties (not really relevant in this specific situation).

No such integration requirements could be identified, except a "nice to have" interface to the NIDA (National Identification) system.

Typical protocols:

1. Usage of secure SOAP, WSDL, XML, RESTful web services, etc. (as provider or consumer)

2. Direct secure exposure of RDBMS objects through named pipes, TCP/IP, SQL*Net, etc. (for exposure or consumption of CRUD functions)

1.3. Client server architecture

This Technology Solution Architecture definition can be used for certain components of the solution where specific Infrastructure and/or platform constraints or functional requirements necessitate such an architecture.

Examples:

1. mobile applications
2. back-office client applications
3. development/support workbenches
4. back-scanning workstations

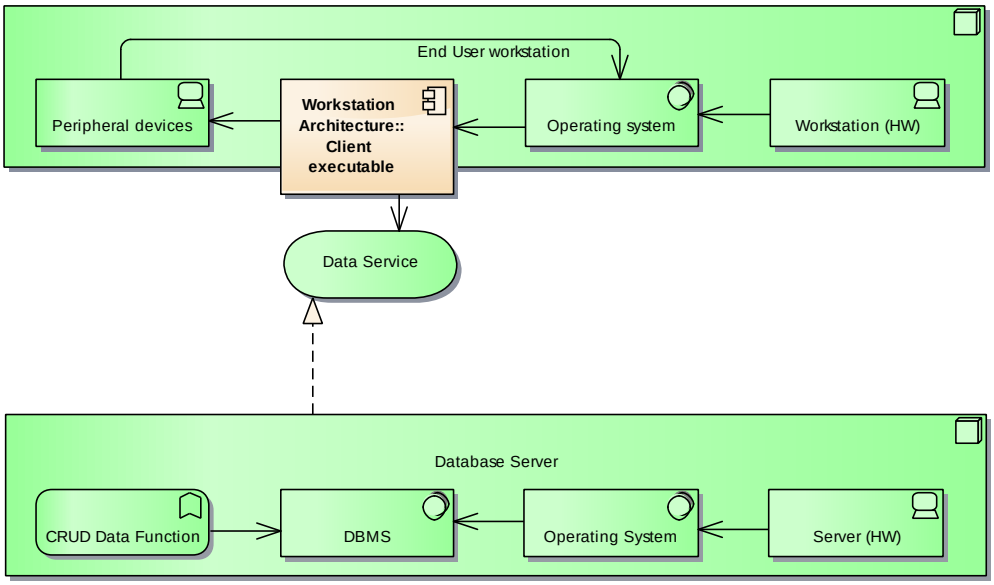
Restriction:

The full client executable and its components must be automatically updated when new versions become available.

1.3.1. Client Server Architecture Diagram

This diagram depicts this client server architecture on a high level.

Figure 3: Client Server Architecture Diagram



1.4. Workstation Architecture

This section describes the workstation architecture for various types of workstations in more detail.

1.4.1. Workstation Architecture Diagram

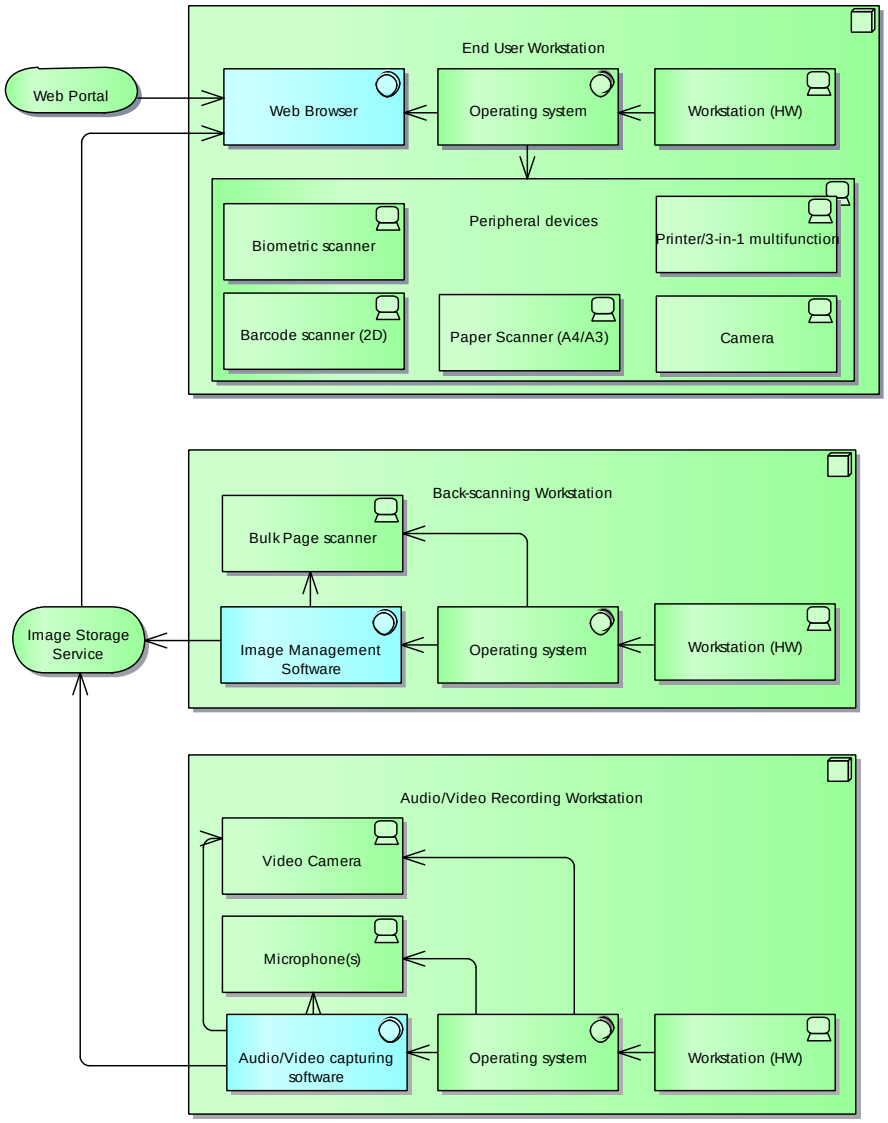
The associated diagram depicts the generic technology patterns for the client workstation nodes. Even though different types of workstations are indicated separately, they may also be combined on one or more workstation(s) in different combinations and permutations.

All workstations will be suitably protected, including (but not limited to) the following mechanisms:

- Operating system and network (logical) access control and management
- Anti-virus, fire-wall and spyware protection
- Patch management
- Configuration management
- Protection against system administrator misuse and ensure segregation of duty
- Audit logging
- Protection against environmental risks such as fire, water and theft
- Physical access control
- A comprehensive set of system management tools (e.g. remote management software; monitoring software – to monitor the state of the workstation)
- Software discovery tools – to monitor and discover unwanted software installed on the workstations

(These aspects are assumed as normal day-to-day operational best practice of the infrastructure and platform technology and are thus not described in great detail anywhere in this (acquired Application supporting) Technology Solution Architect

Figure 4: Workstation Architecture Diagram



1.4.1.1. Back-scanning Workstation (Node)

This node is specifically for back-scanning of all the historic documentation, if required.

1.4.1.1.1. Bulk Page scanner (Device)

This scanner provides for the batch loading of documents which is then scanned sequentially and automatically and directly associated with other data elements.

A4 to A3 size should be sufficient.

No specific requirements for bigger sizes could be identified.

No specific standardisation requirements could be identified.

1.4.1.1.2. Image Management Software (Software)

Software to manage images/files. The specific functional requirements will depend on the underlying design concept employed out of the four basic options.

1.4.1.2. Audio/Video Recording Workstation (Node)

This node is specifically for the audio and/or video recording of events, whether for historic record purposes or due to geographic constraints.

1.4.1.2.1. Microphone(s) (Device)

Any standard microphone.

1.4.1.2.2. Video Camera (Device)

Any IP based video camera.

1.4.1.2.3. Audio/Video capturing software (Software)

Software to handle the capturing and editing of digital audio and/or video data. No specific digital formats or codecs are prescribed from a standardisation perspective.

1.5. Network Architecture

This section describes the network architecture for the various segments of the conceptual network in more detail.

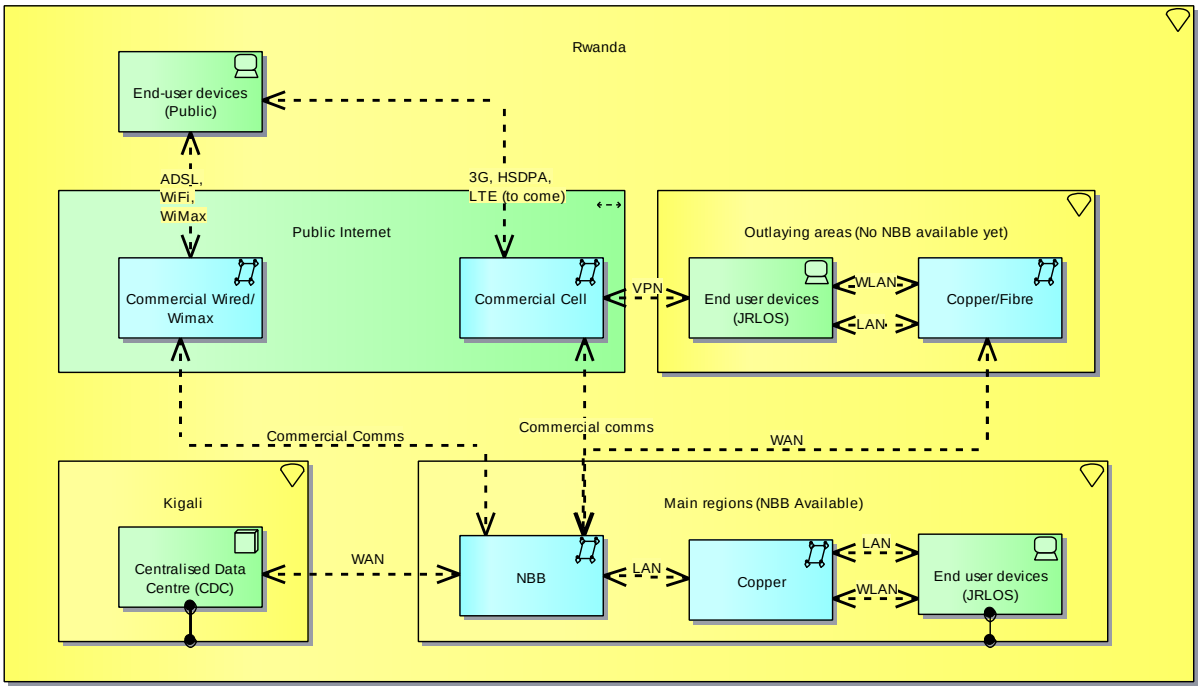
1.5.1. Network Architecture diagram

This diagram depicts the conceptual network architecture which should be in place to implement an application solution on. The GoR will be responsible to ensure this.

The detail design of the network is available, but is not relevant here.

This diagram endeavours to rather depict the different infrastructure components with the relevant platform components and protocols running on them in the different locations. It also depicts the two main network topologies available for consumption by the applications.

Figure 5: Network Architecture diagram



1.6. Application Servicing

This section describes the Service architecture between the underlying technology and the utilising application in more detail.

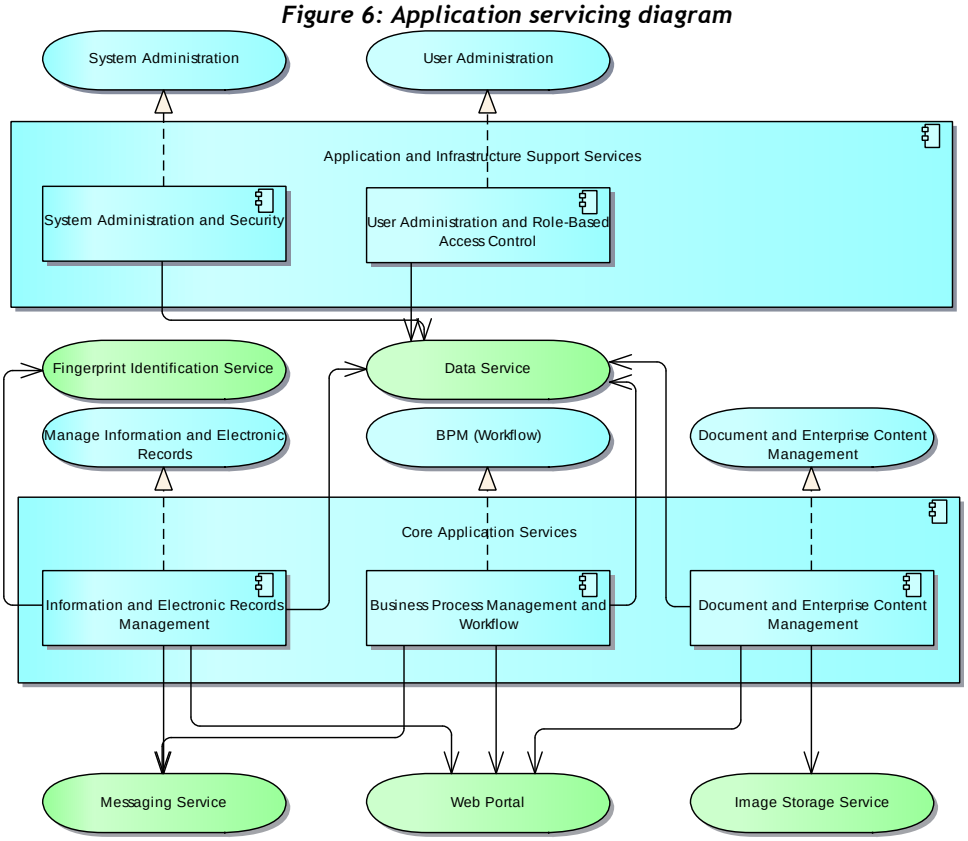
1.6.1. Application servicing diagram

The associated diagram depicts the consumption of the core technology service layer by the Core Application Services components and the Application and Infrastructure Support Services components.

The details of how the core technology services are realised are provided in the rest of the technology diagrams.

The details of how the indicated core application services (realised by these application components) are consumed, are provided in the rest of the application diagrams.

Figure 6: Application servicing diagram



1.7. Integration servicing

This section describes the consumption of the external integration services in more detail.

1.7.1. Integration servicing diagram

The associated diagram depicts the consumption of the core technology service layer by external Services components.

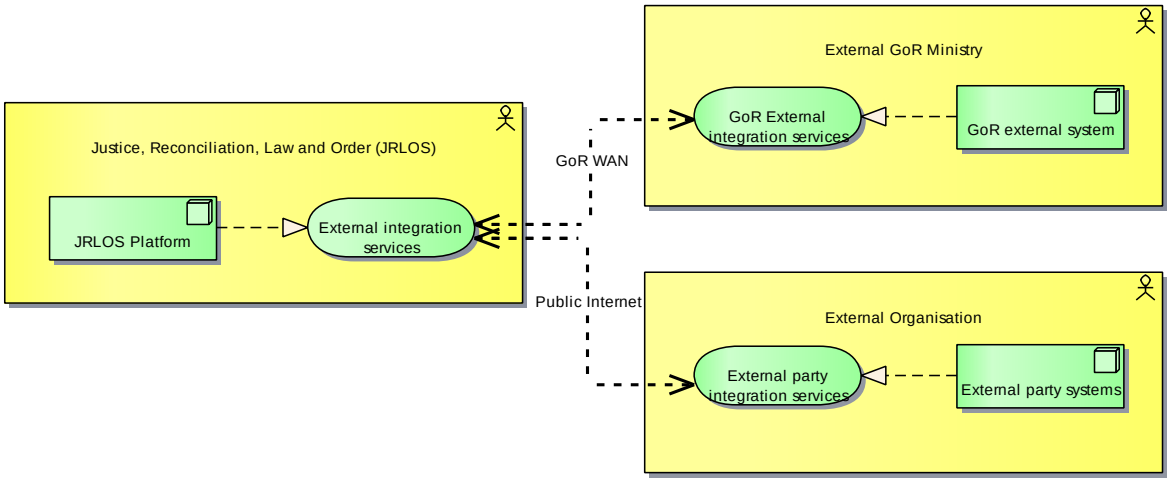
External in this context means ‘external to JRLOS’, which can be other GoR ministries or any other external entities such as private companies, public institutions, etc.

The details of how the core technology services are realised (or consumed) are provided in the rest of the technology diagrams.

The details of how these services are consumed (or realised) externally are only provided on a very high level for illustrative purposes, as the detail is totally dependent on the 3rd party’s detail design.

(Just as a reminder: These services include Web Services as well as secure open data services).

Figure 7: Integration servicing diagram



1.8. Platform Usage Catalogue

This section describes the Platform usage catalogue in more detail.

1.8.1. Usage Catalogue diagram

The rest of this Technology Architecture definition focuses on the most important Technology aspects from the support of an Application Solution perspective. To ensure the full picture is kept in mind, the associated diagram depicts the full spectrum of specific technology components, both from usage in the existing baseline as well as for the proposed solution.

Only a snapshot is presented here pictorially; the full Technology Usage Catalogue was developed in the form of a spreadsheet with about 350 detailed components. It is provided as a separate artifact, with Document ID: Technology Usage Catalogue vx_y.xlsx (where x_y is the specific version number).

This is a dynamic document, which will still evolve over time as transitional progress is made from one baseline to the next. Expansion to more applications, services and other institutions in subsequent phases may also influence the selections. Changes will be depicted with the version number suffix.

The spreadsheet contains the full Technical Reference Model (TRM) structure definition as delivered during Phase 1 of the project, with a matrix mapping to indicate which of

these technology components are relevant somewhere in the in-scope business domain. Both the Baseline Architecture and the Target/Solution Architecture are mapped in the matrix. The full description of each Domain, sub-Domain and Component can be referenced in the full TRM document.

The usage of some components will only be determined by the detail as-built design of the eventual solution.

Some other aspects are viewed as day-to-day operations of the Technology infrastructure and were thus not defined in further detail in this Technology Solution Architecture, which focuses on the acquisition of a specific (Application) Solution through a tender process.

Candidates for standardisation are also indicated in the Catalogue and will also influence the document changes as specific standardisation decisions are made.

Figure 8: Usage Catalogue diagram

Technology artifacts			Baseline Investigation					Target Suggestion	
			Investigated			Not investigated			
Technology Domain	Technology Sub Domain	Technology Component	None found	Documented	Not documented	Non relevant	Low priority	Required	Standard
Application and Directory Service	Application Service	Application Server (software)		X				X	
	Application Service Enablement	Load Balancing	X						
		Site Manager	X						
	Directory Service	Management Directory	X						
		Meta Directory	X						
	Search Engine	Data Search Engine	X						
		Internet Search Engine			X			X	
		Intranet Search Engine	X						
	Transaction Monitor	Transaction Monitor Tool	X						
	Web Service	Portal		X				X	
		Web Application Service		X				X	
		Web service - Internet		X				X	
		Web service - Intranet		X				X	
		Web Service - Proxy	X						

1.9. Standardisation

Although a Standardisation Information Base (SIB) is quite important for a Technology Solution Architecture, only a very few informal inputs could be obtained regarding the standardisation of Solution Building Blocks.

Most of these are also only preferences and does not exclude any solution if it does not comply to the set standard.

The following formal feedback was received on a specific request for the standardisation methodology:

"As agreed we are pursuing standardization across the entire government (we are not just focused particularly on justice sector) and we intend to have RDB/ICT and MINISTRY OF ICT jointly handling the decision making as far as the SBB standardization is concerned.

*And we should basically be able to standardize everything apart from <<**manufacturers of equipments**>> as this would be against our procurement law which prohibits the mention of vendor name in any procurement document.*

The review of the processes and standards should be in my view handled by the Architecture Review Board.

*And after consultation we thought that <<**preferences should be all treated as standards**>> instead of separating them to avoid confusion."*

A formal Architecture Review Board (ARB) does not yet exist at the time of compiling this architecture.

Candidates for Standardisation are thus indicated in the separate Usage Catalogue spreadsheet. This Catalogue can be used to indicate the specific SBB's once a formal standardisation decision has been made through the standardisation process.

Due to the importance, some standardisation requirements have been proposed throughout the Architecture definition. Unfortunately it is very difficult to standardise without using some form of product name, which directly translates to a vendor name.

As a result RDB/JRLOS will have to assess whether these proposals contravene the said procurement law if these requirements would be included in the formal RFP going out to the market and then delete or rephrase these requirements.

1.10. Security Standards

This section consists of two main aspects.

- The first aspect is a "Draft Report for Government Security Architecture for Republic of Rwanda", which was developed from July 2013 to December 2013. Reference should be made to this document for assessment of the required security controls for the full architecture. At the time of compiling this Solution Architecture, said document has not been formally approved yet.
- The second aspect is to identify and list the international information security standards that should apply to JRLOS systems. It specifies uniform use of specific

technologies, parameters or procedures and spans across all four the architectural domains.

These definitions are not strictly part of a Technology Solution Architecture, but part of day-to-day secure operational running of technology infrastructure. Dependent on the complexity of the technology and applications used in other GoR ministries, some of these standards and definitions may already be in operation and should be taken cognisance of.

Both these aspects shall aim to:

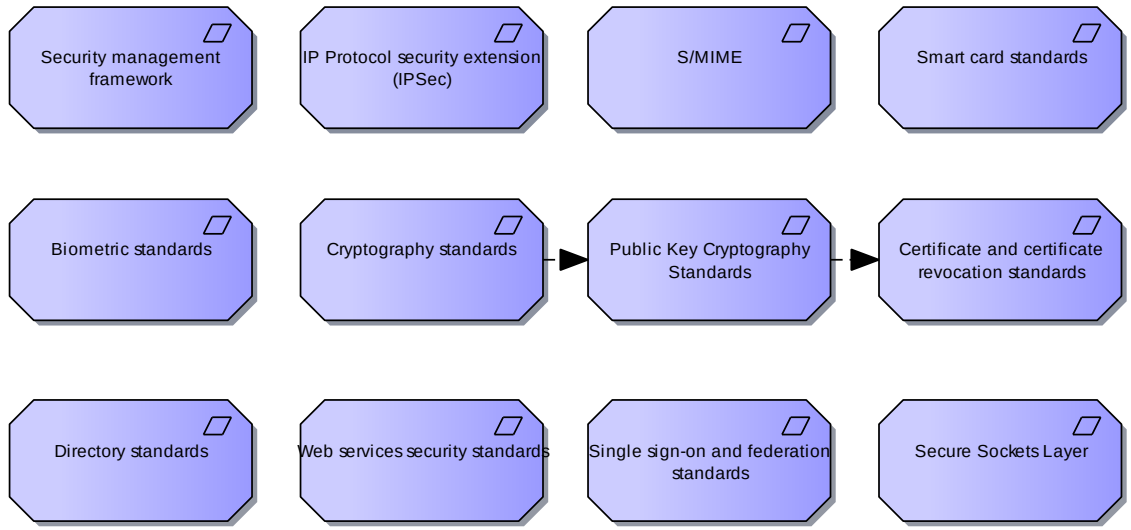
- Define key performance objectives that shall be achieved to meet security objectives, and which can be measured
- Define technical standards that will be complied with when implementing security technology controls
- Form the basis for metrics, measurement and audit purposes

The primary GoR report will override the generic international standards as applicable and must be approved, adopted and maintained as foundational security policy to guide and lead future Enterprise Architecture initiatives, system development, operation and procurement.

1.10.1. Security Standards and Requirements diagram

The associated diagram provides a high-level overview of the second addressed aspect to indicate different international security standards which should be applicable to ensure secure information and systems.

Figure 9: Security Standards and Requirements diagram



1.10.1.1. Biometric standards (Requirement)

Biometric standards, standards for biometric readers/devices and Biometric Data Interchange standards

Consider the following ISO/IEC biometric standards:

- ISO/IEC 19794: Information technology - Biometric data interchange formats
- ISO/IEC 19784: Information technology - Biometric application programming interface (BioAPI)
- SO/IEC 19785: Information technology - Common Biometric Exchange Formats Framework.
- OASIS XCBF 1.1 - Secure XML encoding for exchanging biometric data

Requirement: Security - Want - *V*

Biometric readers/devices must conform to the associated standards:

Specific standards:

- Optical scanner with at least 500 DPI resolution/sensor
- Sensing area of at least 0.95 X 0.5 cm or more, or similar oval size
- USB Interface
- Must have 128-bit symmetric algorithms or equivalent – non-

repeatable key

- Must have a Live Finger Detection (LFD) feature
- Support capture, enrol, identify and verify
- Support ISO/IEC 19794-4 for finger image data
- Support ISO/IEC 19794-2 for finger minutiae data
- Support ISO/IEC 19794-3 for finger pattern spectral data
- Support ISO/IEC 19794-8 for finger pattern skeletal data

Rationale:

Conforming to the above standards will ensure a high degree of interoperability with biometric solutions and products.

BioAPI provides a Human Authentication - Application Program Interface (HA-API) compatibility mode to support existing HA-API implementations. HA-API is a commonly implemented standard describing a generic API designed to integrate biometrics into applications requiring identification. It supports the enrolment sampling, processing and verification of biometrics.

1.10.1.2. Certificate and certificate revocation standards (Requirement)

- Public Key Certificates must be based on the X.509v3 standard (ITU-T)
- Certificate revocation list (CRL) (X.509v2 CRL / RFC3280)
- Online Certificate Status Protocol (OCSP) / RFC 2560

Requirement: Security - Want - *GoR/V*

The associated standards should be utilised for PKI certificates.

Rationale:

The standards are widely-used industry standards for certificates and certificate revocation.

Despite the widespread acceptance of this standard, care must be taken when dealing with vendors. Proprietary extensions to

certificates could inhibit interoperability and shall be avoided.

1.10.1.3. Cryptography standards (Requirement)

RSA-2048 (ISO/IEC 18033-2) - Public key cryptographic standard

AES (FIPS 197) - Symmetric key cryptographic standard

SHA-256, SHA-384, SHA-512 (FIPS pub 180-2) and RIPEMD-160 (ISO/IEC) - Message digest standards

DSA, RSA (ISO/IEC14888) (+SHA) and ECDSA - Digital signature standards

Requirement: Security - Want - *GoR/V*

Cryptography must be based on open standards.

Rationale:

The associated cryptographic standards have received wide acceptability and can be found in most products. Only full strength cryptography shall be used. For example browsers are often supplied with weakened versions such as 40 bit DES, RC2 and RC4. Only browsers with full strength keys shall be used for transactions.

1.10.1.4. Public Key Cryptography Standards (Requirement)

- PKCS#1 / RFC3447 - 2.1 - RSA Cryptography Standard
- PKCS#3 - 1.4 - Diffie-Hellman Key Agreement Standard (key exchange)
- PKCS#5 / RFC 2898 - 2.0 - Password-based Encryption Standard
- PKCS#7 / RFC 2315 - 1.5 - Cryptographic Message Syntax Standard - used to sign and/or encrypt messages under a PKI
- PKCS#10 / RFC 2986 - 1.7 - Format of messages sent to a Certification Authority to request certification of a public key.
- PKCS#11 - 2.20 - Cryptographic Token Interface (cryptoki) - An API defining a generic interface to cryptographic tokens

Requirement: Security - Want - *GoR/V*

The associated standards should be utilised for PKI implementation.

Rationale:

Although devised and published by RSA laboratories, and therefore not actual industry standards despite the name, these standards are widely used and have in recent years begun to move into 'standards track' processes with one or more of the standards organisations (e.g. IETF PKIX working group).

1.10.1.5. Security management framework (Requirement)

Introduction

The UK Government's DTI published a document in 1992 with the title 'Code of Practice for Information Security Management'.

This was subsequently upgraded by the British Standards Institute (BSI) as 'BS 7799-1 - Code of Practice for Information Security' in 1995.

BSI enhanced this document, and also published a second part: BS7799-2, which was a specification for security management.

In 2000 ISO adopted BS 7799-1 and renamed it to ISO/IEC 17799:2000.

In 2005 ISO also adopted BS7799-2, which became ISO/IEC 27001:2005.

ISO/IEC 17799 was also re-published in 2005 and was renamed to ISO/IEC 27002 in July 2007.

Also in 2005 BSI published BS7799-3. This is 'Guidelines for information security risk management' which are likely to evolve into an ISO standard (possibly ISO 27005).[1]

The standard ("Code of Practice for information security management") establishes controls, guidelines and general principles for implementing and maintaining information security management in an organisation and to develop information security standards and practices.

The ISO/IEC 27002 standard is a framework containing 11 security clauses collectively containing a total of 39 main security categories. The standard also contains an introductory clause introducing risk assessment and treatment. Each clause contains a

control objective and a number of controls with implementation guidance to achieve the stated control objective.

In summary, existing documents are named as:

- ISO/IEC 27001:2005 ("Information technology-security techniques-code of practice for information security management")
- ISO/IEC 27002:2007 ("Code of Practice for information security management")

---

[1] http://www.27000.org/

Requirement: Security - Want - *GoR/V*

The ISO/IEC 27002 standard framework should be utilised as far as applicable.

1.10.1.6. Single sign-on and federation standards (Requirement)

Some of the applicable standards in this domain are:

- Kerberos (Massachusetts Institute of Technology - MIT) is a popular mechanism for applications to entirely externalise authentication. Users sign into the Kerberos server, and are issued a ticket, which their client software presents to servers that they attempt to access. Kerberos allows individuals communicating over an insecure network to prove their identity to one another in a secure manner. Its designers aimed primarily at a client-server model, and it provides mutual authentication, i.e. both the user and the server verify each other's identity.
- Web-services federation is a new approach, also for web applications, which uses standards-based protocols to enable one application to assert the identity of a user to another, thereby avoiding the need for redundant authentication. Standards to support federation include SAML and WS-Federation.
- JOSSO or Java Open Single Sign-On, is an open source J2EE-based SSO infrastructure aimed to provide a solution for centralised platform neutral user authentication. It uses web services for asserting user identity, allowing the integration of non-Java applications (i.e. PHP, Microsoft ASP, etc.) to the Single Sign-On Service using the SOAP over HTTP protocol.

Requirement: Security - Preference - *GoR/V*

The associated standards should be utilised as far as applicable.

1.10.1.7. Smart card standards (Requirement)

- ISO/IEC 7810 Identification Cards: Physical Characteristics, Resistance to chemicals
- ISO/IEC 7811 Describing the recording technique on identification cards
- ISO/IEC 7812 Integrated Circuit(s) cards with contacts
- ISO/IEC 7816-1 Physical characteristics
- ISO/IEC 7816-2 for Dimensions and locations of the contacts
- ISO/IEC 7816-3 for Electronic signals and transmission protocols;
- ISO/IEC 7816-4 for Inter-industry commands for interchange
- ISO/IEC 7816-5 for Numbering system and registration procedure for application identifiers
- ISO/IEC 7816-6 for Inter-industry data elements
- ISO 877 - Daylight Exposure Stress Method
- Card manufacturing compliant with ISO 9001:2001
- Security standards:
- Evaluation Assurance Level 4+ (EAL4+) of the Common Criteria
- NIST cryptographic module specification (FIPS publication 140-2, level 2)
- Encryption and authentication, and digital signature standards:

♦ 3-DES

♦ RSA

♦ SHA

♦ AES or ECC PKCS#1-RSA (version 1.5)

♦ PKCS #11 or PC/SC for integration of smart cards and host/reader-side applications

♦ FIPS 186-3

- Match on smartcard functionality with integrated fingerprint scanner compliant to the MINEX II certification standard and ISO 9794-2: 2005 and ISO 7501/ICAO MRTD standard

Use ISO 14443A and Mifare Smart Card standards for contactless smart cards.

Requirement: Security - Want - *V*

Comply to the listed Smart Card standards as far applicable

Rationale:

- The command set defined by the ISO 7816 standards are wholly included or in part by most smart cards on the market
- PKCS #11 is a widely-accepted standard for integrating smart cards to applications supported by many vendors

ISO 14443A standards for contactless smart cards define the characteristics and communication protocols between contactless cards and card reader. These standards are still in development. The Mifare architecture is the *de facto* global interface standard for contactless and is based on ISO 14443A. Contactless cards under this standard use RF power and frequency protocols and cover read/write distances up to 10 cms of the reader.

1.10.1.8. Web services security standards (Requirement)

There are a variety of specifications associated with web services. These specifications are in varying degrees of maturity and are maintained or supported by various standards bodies and entities, e.g. W3C (World Wide Web Consortium) and OASIS. Specifications may complement, overlap, and compete with each other. These include:

- **WS-Security**[1]: WS-Security (Web Services Security) or WSS is a communications protocol providing a means for applying security attach signature and encryption headers to SOAP messages. In addition, it describes how to attach security tokens, including binary security tokens such as X.509 certificates and Kerberos tickets, to messages. WS-Security build on standards such as XML Signature and XML Encryption.
- **XML Signature**[2]: XML Signature (also called XMLDsig, XML-DSig, XML-Sig) is a W3C recommendation that defines an XML syntax for digital signatures. Functionally, it

has much in common with PKCS#7 (Public Key Cryptography Standards) but is more extensible and geared towards signing XML documents. It is used by others.

- XML Encryption[3] is a W3C specification that defines how to encrypt the content of an XML element.
- **Security Assertion Markup Language (SAML):** SAML[4] is an OASIS XML standard for exchanging authentication and authorisation data between security domains, that is, between an identity provider (a producer of assertions) and a service provider (a consumer of assertions). The single most important problem that SAML is trying to solve is the Web Browser Single Sign-On (SSO) issue. SAML has become the definitive standard underlying many web SSO solutions in the enterprise identity management problem space.
- **XML Key Management (XKMS)**[5] specifies protocols for distributing and registering public keys, suitable for use in conjunction with the W3C Recommendations for XML Signature (XML-Sig) and XML Encryption (XML-Enc).
- **eXtensible Access Control Markup Language (XACML)**[6] may be used to describe authorisation policies. It is a declarative access control policy language implemented in XML and a processing model, describing how to interpret the policies.

Other standards include **WS-SecureConversation** (WSSC), **WS-SecurityPolicy**, **WS-Trust** and **WS-Federation**.

---

[1] http://en.wikipedia.org/wiki/WS-Security

[2] http://en.wikipedia.org/wiki/XML_Signature

[3] http://en.wikipedia.org/wiki/XML_Encryption

[4] http://en.wikipedia.org/wiki/SAML

[5] http://www.w3.org/TR/xkms2/

[6] http://en.wikipedia.org/wiki/XACML

Requirement: Security - Want - *GoR/V*

Web Sevices should comply to the associated standards as far as applicable.

1.11. Recommendations

This section contains various recommendations on various aspects of this Technology Solution Architecture.

1.11.1. Approach

The specific objective of this Solution Architecture was the acquisition of a specific (application) solution to cater for the business and functional requirements of the JRLOS Integrated Electronic Case Management System. This technology section thus documented the Technology (Infrastructure and Platform) Architecture Definition required to support/enable such a (application) solution.

From analysis of the **business model** of the JRLOS it can be derived that it is primarily a Cost and Operating business model domain.

It follows that it doesn't require core differentiating or competitive competencies. The goal would therefore be to meet market standards with a focus on effectiveness and efficiency.

From a Technology perspective this will have the effect that Standard Best Practice should be employed, with a focus on 'Out of the box' functionality, infrastructure and platforms with as much standardisation across business units as possible. Differentiation (for competitiveness) will add no value to the business model.

From a Cost perspective this will have the effect that the order of preference would be to:

- Re-use existing
- Buy 'off the shelf'
- Build custom

for all the technology layers and components.

The existing network, servers, end-user devices, system software, platform software, data centres, etc. (as examples) should therefore first be re-used as far as possible in any (application) RFP response.

These aspects are addressed, where applicable, throughout the architecture definition.

Once a specific (application) system and/or approach has been decided upon through the first RFP procurement process, various other (Technology) Solution Architectures may be spawned, based on the specific decision. Examples could be:

- the acquisition of additional server capacity for the acquired system
- the acquisition of additional user workstations for the RNP
- the acquisition of mobile end-user devices for the RNP or the Judiciary
- the development and building of a JRLOS-owned specific Centralised Data Centre
- the strategic establishment of a Government-wide secure cloud computing infrastructure
- security upgrading of the existing National Data Centre at Telecom house in Kigali to cater for the specific JRLOS security requirements
- the upgrading and/or expansion of the NBB
- the installation of (device based) VPN security on the then newly implemented 4G/LTE cell network infrastructure
- the establishment of a Government wide Call Centre
- the acquisition of a full blown AFIS system
- the initiation of a Back Record Conversion project for manual fingerprint records or for archived case documentation

Each specific Solution Architecture will be followed by a detail design, development, customisation, implementation lifecycle, which will then be concluded with a new (As-Is) Solution Architecture phase to document the new ‘as-built’ architecture for that specific Solution Architecture.

A number of these spawned Solution Architecture phases can also be combined into a single subsequent (As-Is) Solution Architecture, dependent on the interdependency between them.

1.11.2. Re-Use opportunities

In the previous phase (Phase 2) all the existing baseline applications were investigated and documented.

Most of these applications and their specific technology infrastructure and platforms are not really re-usable in the proposed solution architecture.

The one exception to this statement is the Omni suite of products:

The OmniDocs and OmniFlow components of this suite is already licensed for between 2000 - 3000 GoR users and mainly used for simple workflows, such as mail routing, leave applications, transport applications, etc. Various expansions in functionality are continually being developed. At this stage it is implemented and managed by RDB. A specific Case Management functional expansion is also available, based on the same technology platform.

The basic Technology Architecture is depicted in the following figure.

This existing Technology infrastructure, platform and Licenses may be utilised in a proposed (application) solution.

The OmniSuite Technology Architecture fits closely to the proposed Technology Solution Architecture Definition, but there are also numerous gaps.

Three basic options are therefore available for exploration:

1. Customise the Master Data requirements into the Omni platform.

2. Integrate 3rd party COTS Master Data functionality with the Omni Document Management and Workflow functionality.

3. Utilise the Omni Case Management functional extension and customize any additional Master Data requirements.

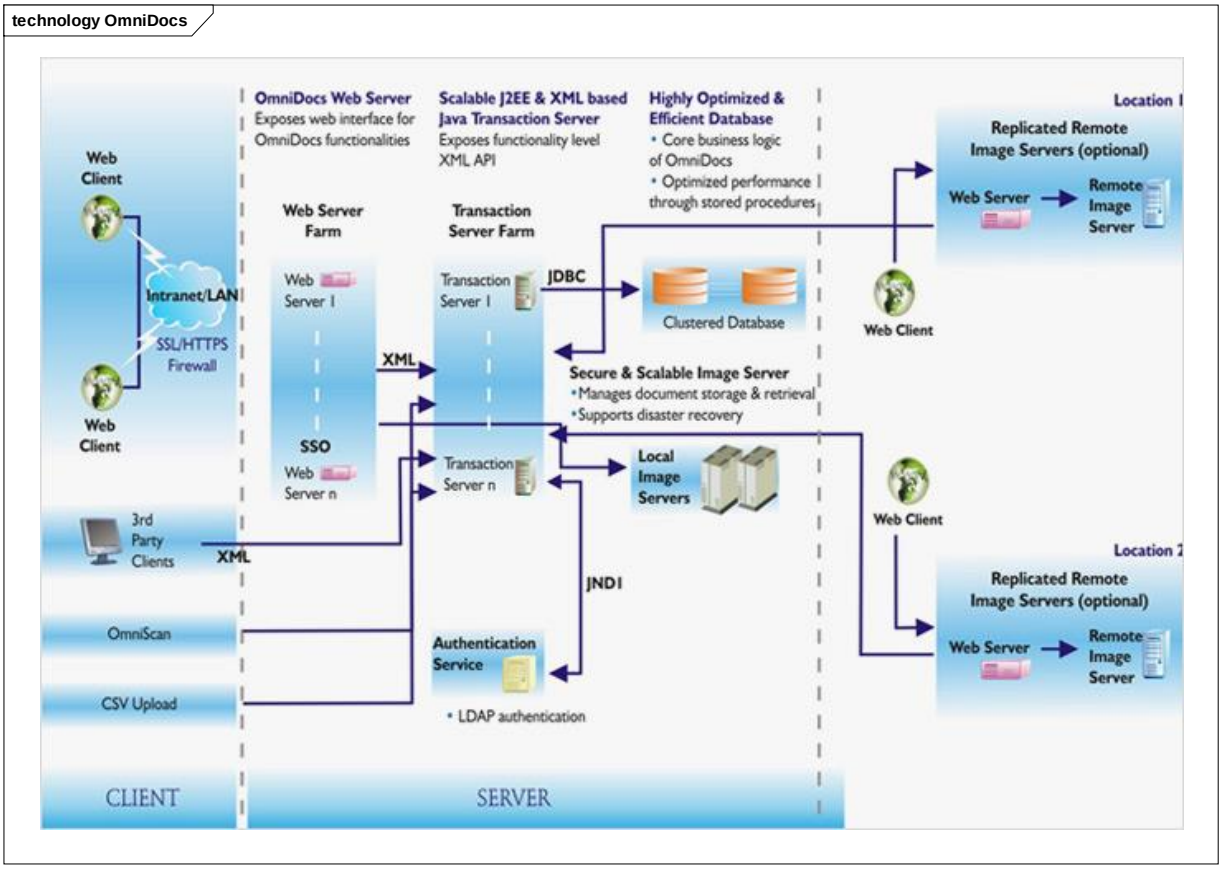
(From a customisation perspective, the PrisonWatch application's data models may be used as a base for the customisation of the Master data requirements for the prison management functionality.)

For more information on this existing Baseline Architecture, refer to www.newgensoft.com or the RDB Technology Architects.

1.11.2.1. Re-Use opportunity

This diagram depicts the basic Technology Architecture as extracted from the OmniDocs documentation.

Figure 10: Re-Use opportunity



1.11.3. Usage profiles

To aid in the calculation of detailed infrastructure requirements and licensing for a proposed solution, it is recommended that the following guidelines are followed:

Number of users:

- The total number of eventual registered users will not exceed 3000
- During the phased implementation approach this total will only be reached in the final stages of implementation
- The Kigali region will contribute about 800 users of this total user base

- About another 1200 users will be from the regions which are already linked to the NBB
- The remaining 1000 users will be from regions which are not currently linked to the NBB and may have to use the commercial cell infrastructure if costing prohibits expanded linking to the NBB

User concurrency:

No existing system is available to accurately determine user concurrency parameters. The user concurrency ratio is guesstimated even as low as:

- 40% of total registered users for regularly logged-in users
- 20% of total registered users for active transacting users

Transaction profiles:

No existing system is available to accurately determine user transaction profile parameters.

On average, about:

- 50 new cases per month will be registered at the start of the value chain
- 40 of these will continue through to actual court cases

Exceptional cases can have multiple cubic meters of paper evidence associated to the case. Not all will necessarily be required to be scanned into the system.

On average, a case can be taken to comprise of about:

- 100 physical paper pages to be scanned and referenced over its entire lifecycle
- 100 additional normalised database records, spread over various other entities, which will be managed over its entire lifecycle

1.11.4. Ideal solution

As could be seen throughout this Technology Solution Architecture Definition, various options are still allowed on almost all the technology components. The main reason for this is due to the specific objective of this Solution Architecture, namely to go out on

an RFP to acquire a (application) solution to cater for the business and functional requirements of JRLOS.

- Too strict a definition would disqualify a number of potential solutions
- Too loose a definition would invite unsuitable solutions

An attempt was made to get the balance right between these two extremes for this specific objective.

The RFP also does not include any infrastructure or platform components to be provided as it is primarily the GoR's responsibility to provide the underlying infrastructure and platform for such a solution, with guidance from the potential providers in terms of which specific technology components will be required to effectively run their proposed solution.

The acquisition and implementation of these underlying technology components for the successful (application) solution, will most probably be the subject of a successive Solution Architecture and corresponding acquisition/procurement process through a subsequent RFP.

For most of these options (even for the recommended approach) a preference rating was provided, which should provide the ideal infrastructure and platform to be provided for the potential optimal solution. A specific rationale for these preferences is also provided throughout the document.

The detailed architecture definitions will take precedence in case of contradictions with the following. For internal RDB guidance, with due consideration of the usage profiles, the 'ideal solution' can be summarised on a high level as follows:

- Remote sites need to be WAN connected to the National Backbone (or optionally via VPN on cell phone infrastructure) with at least consistent 1.5 Mbps bandwidths. Users with high document/image usage profiles, should have at least 5 Mbps bandwidths
- All the data, images and documents will be hosted centrally in one schema in one central Oracle database instance, hosted on Unix virtual clustered servers at the National Data Centre
- Information access is provided to any device, anytime, anywhere as long as it is connected via a wide range of simple web browsers

- The web server, messaging server, and database server run on the same virtual clustered servers (Even initially one physical server if possible.) This is to ensure ease of support and optimised performance
- Storage is provided on Direct Attached Storage (DAS) with separate physical disks for indexes, master data, meta data and images in a hybrid RAID array with as much usage of SDD as possible
- The AFIS solution is managed as a separate Solution Architecture and acquisition process, with integration to this required solution as required
- Any (internal) integration between different systems is done with database level integration of data, as far as possible
- The database capabilities are used as far as possible to ensure that the database is a 'self-sustained universe' to ensure that any user (operational or support) has the same privileges whether he accesses the database through the provided application or through any 3rd party toolset. This implies full referential integrity, constraints, auditing, business rules and full user/role based security implementation utilising the RDBMS capabilities
- From a programming framework perspective an AJAX type approach with RESTful services in HTML5 is recommended. Java is slightly lighter on resources than .NET, but dependent on the final service choice, any one of the two is acceptable. The Adobe (Air or Forms) and Flash (Forms) frameworks should be avoided as far as possible due to the limited network bandwidths and support complexities. Java applets should also be avoided due to (mobile) device portability and network performance reasons
- Various levels of business services are provided to various types of external parties via SOA-based web services

The biggest challenge in establishing this architecture is connecting the outlaying remote areas to the NBB and/or the NDC with sufficient bandwidths.