E-vote 2011

SSA-U Appendix 3

Customer Technical Platform

Project: E-vote 2011

CONTENT

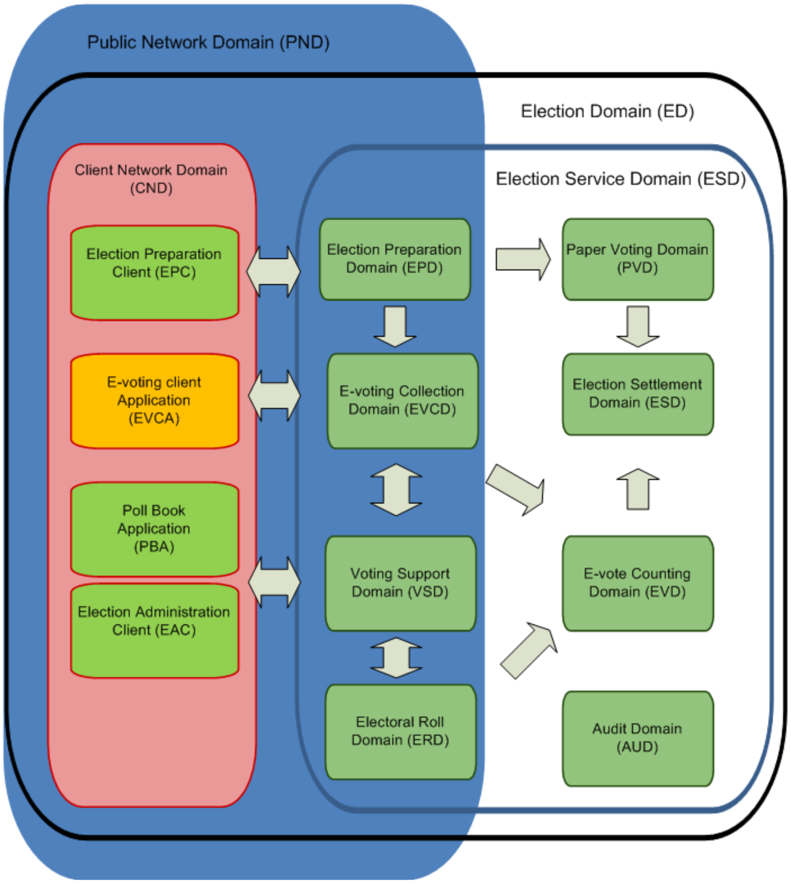
1. Overview of Technical Platform

This document describes the technical platform, in relation to security domains, infrastructure and the required software stack and hardware technology required for operating the platform. The conditions for running the system, with the necessary redundancies, performance and fall-back solution is also described. The system is designed around extensive use of open source software and the use of open source development tools. This document is therefore describing the tools and libraries that will be used.

The document initially describes how the system is designed around the required security domains. The goal is to reduce residual risk to the voting system to the lowest level possible. The domains create the basis for the IT infrastructure design, and how hardware and software solutions are implemented.

2. Security Domains

To meet the established security objectives, the platform is designed around a set of domains as described in the System Requirements Specification. The domain model has been expanded slightly as displayed below.


Public Network Domain (PND)
Election Domain (ED)
Client Network Domain (CND)
Election Service Domain (ESD)
Election Preparation Client (EPC)
E-voting client Application (EVCA)
Poll Book Application (PBA)
Election Administration Client (EAC)
Election Preparation Domain (EPD)
E-voting Collection Domain (EVCD)
Voting Support Domain (VSD)
Electoral Roll Domain (ERD)
Paper Voting Domain (PVD)
Election Settlement Domain (ESD)
E-vote Counting Domain (EVD)
Audit Domain (AUD)

Election Domain (ED) covers all Election and Voting services, with the exception of casting and counting paper vote ballots, as described in the system requirements.

Public Network Domain (PND) is the communications infrastructure not under the control of the Election service operators and clients, as described in the system requirements. This includes the Internet.

Election Service Domain (ESD) is the Government controlled infrastructure, as described in the system requirements.

The following are domains within the ESD:

- **Election Preparation Domain (EPD)** contains all the services required to prepare an election, both for e-voting and p-voting, as described in the system requirements. The Contractor treats this as a "static" domain, meaning that all configuration items will be exported and signed *prior* to the election. No information generated within this domain should be altered during or after the election.

- **Voting Support Domain (VSD)** handles the management of the election. This includes updates to the electoral roll, key management, generation of one-time tokens for e-voting, changes to user access, and other configuration that must be handled throughout the entire election life cycle.

- The **e-Voting Collection Domain (EVCD)** contains the IT infrastructure related to e-voting as described in the systems requirement. The physical separation from EPD and ESD is established as part of this domain configuration.

- The **Electoral Roll Domain (ERD)** contains the IT infrastructure that hosts the Electoral Roll. Access to read and update the Electoral Roll is handled within the domain.

- The **Paper Voting Domain (PVD)** includes the infrastructure to perform p-vote scanning and counting.

- The **Election Settlement Domain (ESD)** contains the IT infrastructure to host all of the election settlement processes, as described in the system requirements. This process is the merging of e- and p-votes, distribution of seats, and computation of results. This domain is air gapped.

- The **E-vote Counting Domain (EVD)** has the IT infrastructure to cleans, mix and count e-votes as required for the contractor solution. Mixing and counting is air gapped, while cleansing is connected to the ERD to remove ineligible votes.

- The **Audit Domain (AUD)** gathers all auditable information, ensures integrity of the auditable data and secures log information in an immutable format. This domain is air gapped.

The **Client Network Domain (CND)**, has been expanded to include the following

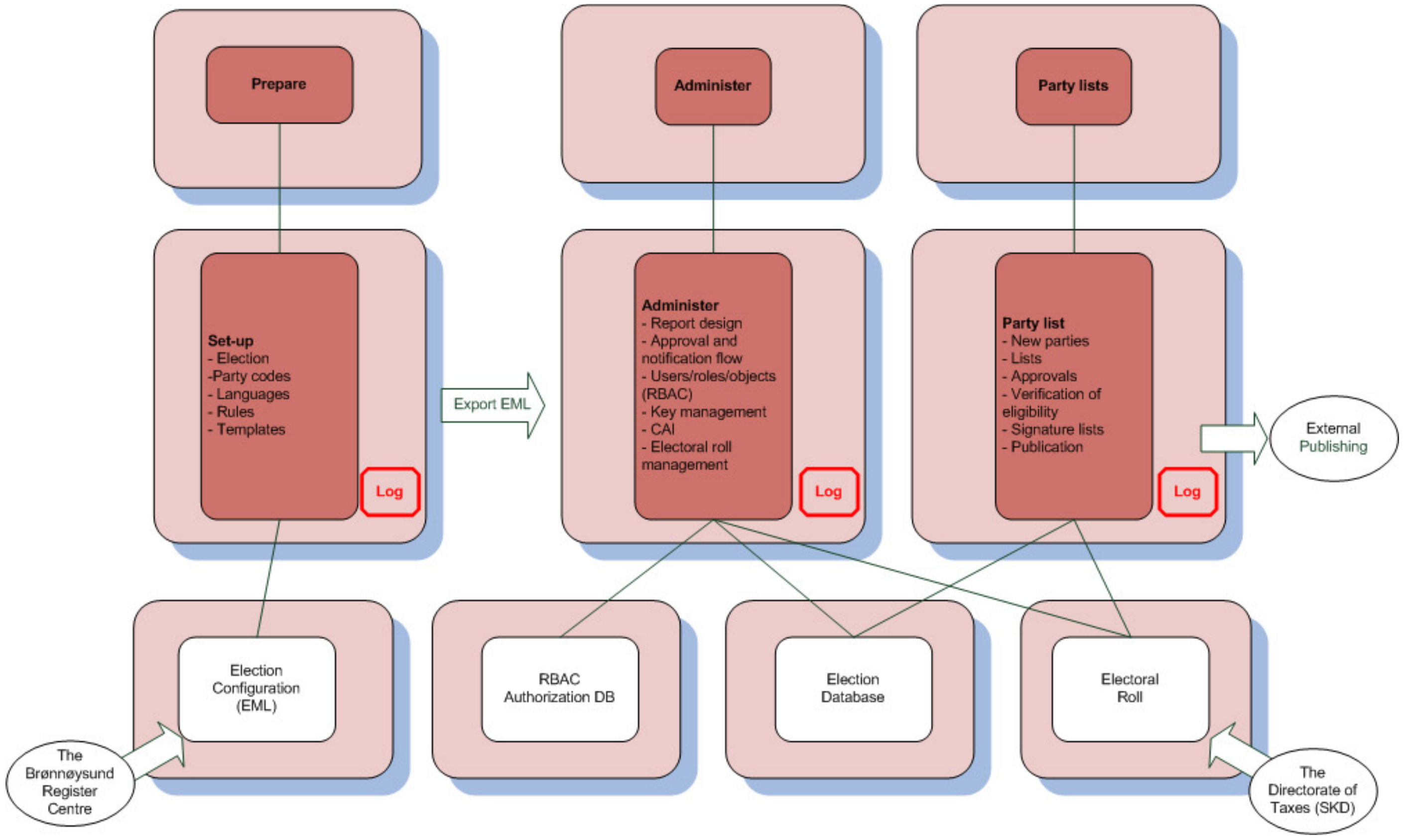
- **Election Preparation Client (EPC**) that provides functionality to set up the election. The client contains functionality both for central and local election administrators. This client is used to define and create the "static" information for the election. This information should not be altered or updated during or after the election.
- **E-Voting Client Application (EVCA)** is the client provided to the voter, either over the internet or at a polling station. This is where the voter makes her choices.
- The **Election Administration Client (EPC)** contains functions for election preparation. However, the set-up functionality has been divided between "static" and "dynamic", where this client domain contains functionality that must be available also during and after the election.
- **Poll book Application (PBA**) contains the application used to verify voter eligibility and mark-off votes in the Electoral Roll.

General considerations: Air gapping and isolation of resources

The guiding principle for the design is to reduce access and air gap as much of the solution as possible. For practical purposes, many of the system components will require online access. If online accessibility is necessary, the focus is to contain this functionality within a government controlled environment (secured network) to any extent possible.

3. Infrastructure Design

3.1. Election setup and Administration



3.1.1. Prepare (election)

The functionality to prepare an election is performed within its own system environment. This includes the user interaction to set up the election with the required information elements such as type of election, parties and party codes (imported from the Brønnøysund Register Centre), and rules applicable to the creation of election results. In addition, it provides functionality to create templates for voter cards, ballots and other material. Other information that may be deemed as “static” to the election is also created through this setup process.

This system will be accessed through an application that is made available to central election officials and local election officials. Role based access and authentication applies. The system can be made available in the PNB, but should be placed on a government control network infrastructure. All transactions are logged by immutable logs.

The output from this process is signed EML files (supplemented with an XML file for information not applicable for EML). These files will not be changed during the election and will in addition be used for auditing purposes.

3.1.2. Administer (election)

For the purpose of managing the “dynamic” configuration elements of the Election System, it is necessary to provide a system that is available to election officials and administrators throughout the entire election life cycle. This system provides the ability to design and run reports, change approval and work flow (restricted), and manage certain aspects of the Role Based Access, such as altering user permissions. In addition, it provides the functionality to manage (create) keys necessary for the various interactions requiring encryption. It also provides functionality to configure integration with authentication systems (CAI), and the management of Electoral Roll (updates).

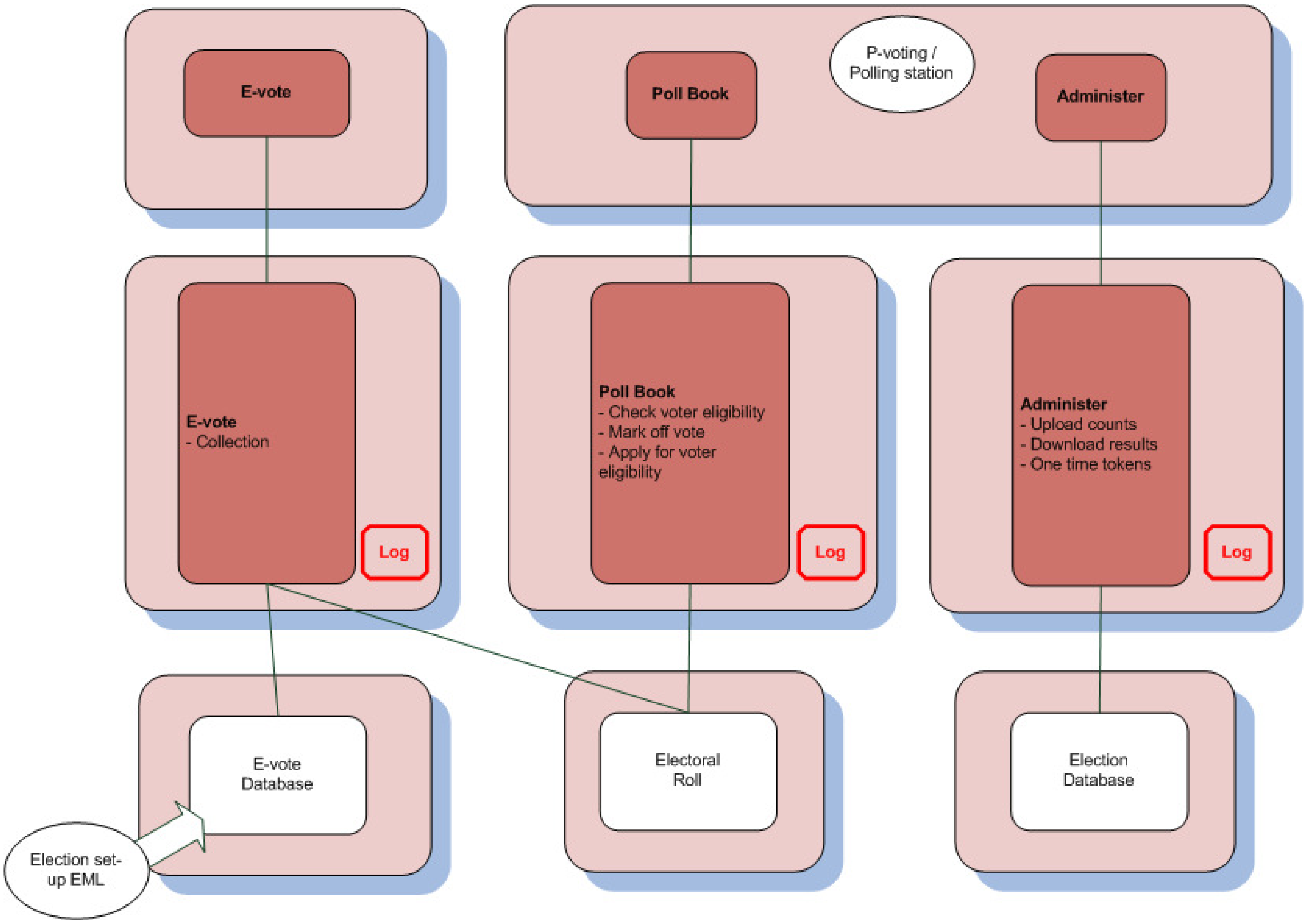
A database, Election Database, is used to store the data elements for this configuration. This database will also have an import of the “static” data from the “Prepare” phase. Role based access and authentication applies, and the applications within this system must only be available within a government controlled network. All transactions are logged by immutable logs.

3.1.3. Party lists (submit and process)

The party list system covers the functionality for submitting and approving party lists. This includes the functionality to register a new party with the associated signature lists, the submittal of candidate lists for existing parties, verification of signatures (new parties), and verification of candidate eligibility in the Electoral Roll. Further, it provides the ability for party officials and election officials to approve or reject submissions with cause. The system provides the final party lists in formats suitable for publishing and the creation of ballots.

Party lists and approval information is stored in the Election Database. It is possible to extract data in EML format for signing and future auditing purposes. All transactions are logged by immutable logs. This application must be widely accessible to the parties and candidates. The application is therefore provided as an Internet application. Role based access and authentication through CAI applies. *Altinn* is an option as front-end to this system.

3.2. Voting systems



3.2.1. E-vote (on the Internet or at Polling Station)

The E-voting system provides the application necessary for the voter to vote electronically through a web interface, and the functionality to collect the votes and store these in a database. The E-vote system is pre-configured with the EML created in the prepare phase.

The E-voting system has integration with CAI for voter authentication. For voting at polling stations, it provides the ability for voters to utilize one-time authentication. The E-vote system has real-time integration with the Electoral Roll system for verification of voter eligibility and mark-off.

All transactions are logged to immutable logs. Votes are signed and encrypted, and exported out of the e-vote database for air gapped counting.

3.2.2. Poll Book (Electoral Roll)

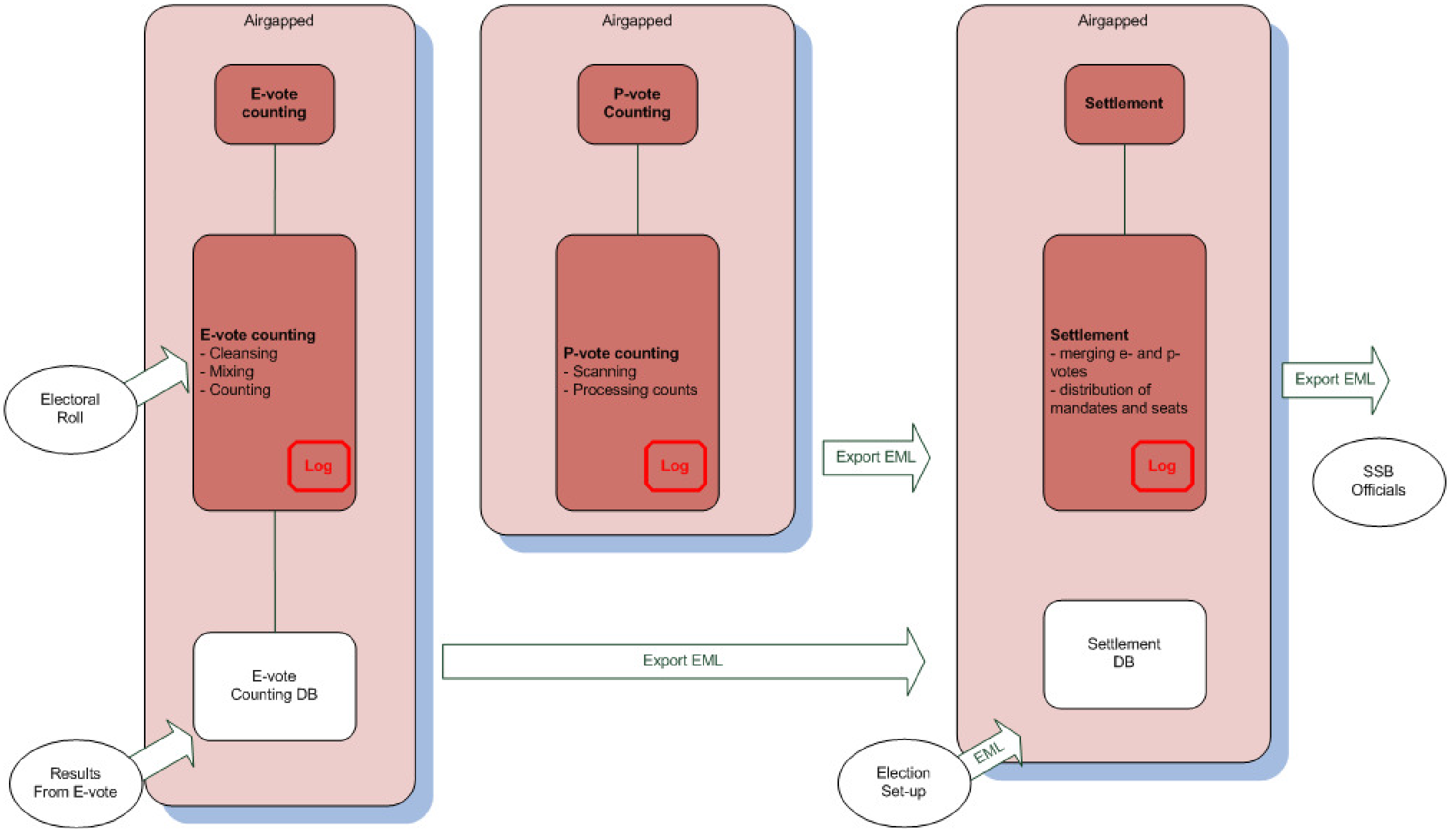
For the purpose of supporting electronic Electoral Roll at the polling stations, a Poll Book application provided. This application allows election officials the ability to verify voter eligibility in the Central Roll. In addition, it allows for vote mark-off.

The Poll Book application will provide the functionality necessary to request voter added to the electoral roll, ability to update information on voter (restricted), and the ability to mark-off votes submitted in cover envelope. The Poll Book application also provides the functionality for the fall-back solution described in section 4.1.3 of this Appendix.

3.2.3. Administer (P-voting)

To allow Election officials the ability to interact with the central Election System, an Administration application is provided. This is a system for uploading P-vote counts, download results for their voter area, and issue one-time tokens for voters that wish to utilize an E-voting station at the Polling Station. Authentication and Role Based access applies. Transactions are logged to immutable logs.

3.3. Counting and Settlement



It is necessary to provide secure, air gapped, system environments for the counting and settlement phases.

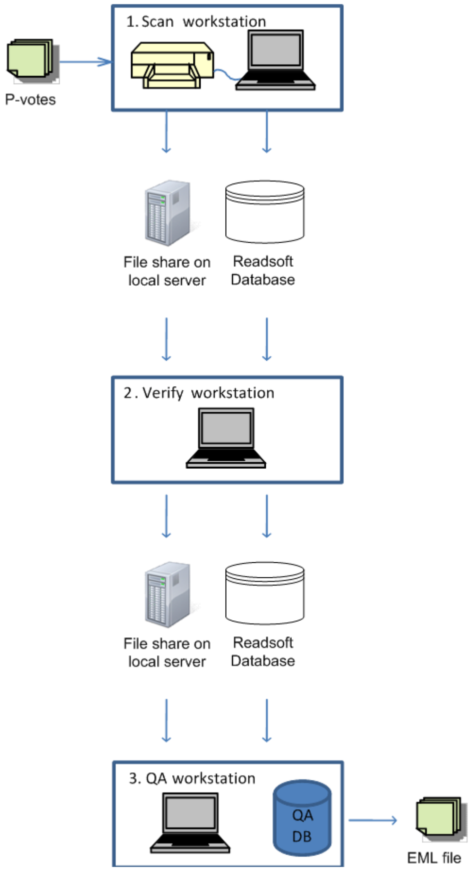
3.3.1. E-vote counting

Counting of E-votes is performed in an isolated central system. Encrypted and signed votes from the E-voting system are imported, and Electoral Roll data is loaded to verify e-voter eligibility and to support the principle of pVotes overriding eVotes. This task is performed by selected Election officials and administrators, through a separate authentication process requiring several security tokens. The final results are generated as a signed EML file.

3.3.2. P-vote counting

Paper based votes may be tallied manually and the result registered (and signed) in the election administrative system. Paper based votes may also be scanned at the scanning centers, and signed EML files are produced at each location. These files are uploaded by the local election officials through the administration interface. The scanning environment is air gapped and access to the system is handled through a separate authentication process.

Scanning process detailed



Paper votes are categorized according to p-voting categories and location. The paper ballots are scanned using a document scanner with Automatic Document Feeder (ADF). The scanner produces TIFF-images in 2 colors with 300dpi resolution. OCR is performed on each image with *Readsoft* software, and interpreted data is stored in a database. Images are stored on a file server. Scans are performed in batches, and batches can be discarded in the event of errors.

Verification is performed by the operator by side-by-side comparison of image files and interpreted data. The operator confirms or rejects values according to the rules defined by the EC. If the election allows the voter to enter candidates on the ballot, the system validates the candidate names against a list of valid candidates.

Quality assurance is performed by importing counts into a QA database. Counts and recounts are compared against each other; these can be manual- or scan counts. The QA system produces an EML file that has to be identical to the EML file produced prior to QA.

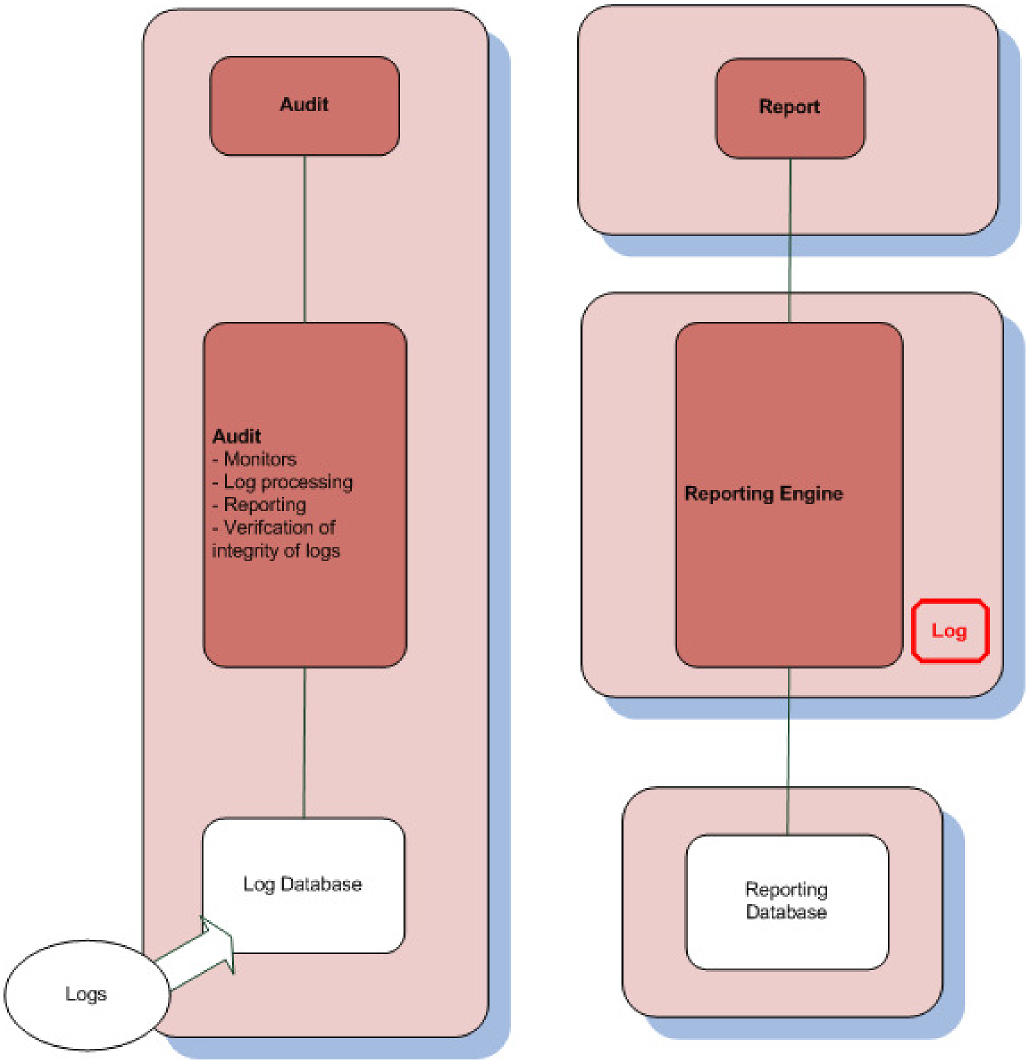
The quality assured EML file is digitally signed, encrypted, and uploaded to the central Election System. The digital signing is performed by one or two representatives from the EC.

3.3.3. Settlement

The final election settlement is a separate air gapped environment only accessible to chosen Election officials and administrators through an authentication process requiring several security tokens. The settlement performs the final merging of E-votes and P-votes, distribution of mandates and seats according the rules imported from the setup EML. The result is a signed EML file containing the final results that is to be published.

For details on the secure and reliable merging of eVotes and pVotes please refer elaboration of requirement F4.2.1 in Appendix 2A.

3.4. Auditing and Reporting



3.4.1. Auditing

For the purpose of analyzing the election setup and results for accuracy, a separate auditing system will be made available. This system has a dual purpose. It provides the ability to set up monitors for real-time tracking of events such as system failures. Secondly, it provides the scanning and analysis of immutable log data and the functionality to search and report on this.

3.4.2. Reporting

A separate system for reporting is provided. A reporting database is used to isolate this functionality from the other systems for performance and aggregation purposes. Information relevant to report on will be continuously fed into this system. This can be election data, results, but also system events and log data that requires more extensive reporting functionality. Reports are created through the Election Administration system, but this system will provide the ability to execute and retrieve reports. Role based access and authentication applies. All report execution is logged to immutable logs.

4. Infrastructure considerations

4.1.1. Hosting

The IT infrastructure is intended to be operated in a secure controlled environment. Security domains requiring network connection must be placed in separate VLAN's and firewall protected. Redundant application servers must utilize load balancer hardware.

Air gapped hardware must be operated in an environment that is accessible to the administrators who perform the counting and settlement tasks necessary to complete the election. Air gapped equipment will not require continuous operation and can be turned off until needed. However, it is paramount that this equipment is protected and that only authorized election and administration personnel are allowed access. As it is necessary to frequently and quickly transport election data *manually* to the air gapped systems, this location should be in connection with the data center.

4.1.2. Redundancy

The following systems have redundancies (fail-over and load balancing):

System	**Reason**
Party list application	Large user group, high visibility to parties
Election Administration applications	Overall management and required for support during the election
E-voting	Large user group, time constraints.
Electoral Roll	Availability for voter checks and vote mark-off. Large volumes.
Role Based access	Security
Log writers	Data volumes, risk for large backlog during down time.
E-vote counting (cleansing, mixing, counting)	Deciphering computation needs
Settlement	Only spare hardware in case primary settlement server would fail to operate

Election set-up is not defined as redundant because the user group is small and timing constraints are not a primary factor. It is assumed that hardware can be repaired within a reasonable time frame. P-voting systems (Scanning) are distributed throughout the counting centers and the local operator will have the ability to restart systems, or receive support to resolve issues.

In addition, reporting is not redundant. The reporting database has copies of data from other databases, and it is presumed that reporting can handle some downtime.

4.1.3. Fall-back solution (GR1.5)

The requirement GR1.5 states that *"In the event of a loss of communication, the Electoral Roll shall still be available locally on client PCs in polling stations. The local copy of the Electoral Roll shall automatically synchronize with the central master copy on restoration of communication. The user in the polling station shall be notified of the loss of and restoration of communications."*

As briefly discussed in Appendix 2A in the chapter 1.1 - Overview of Proposed Functional Solution, there may be changes to the regulations that do not require that voters that are not voting in their own voting district, are handled as "foreign" voters.. This raises some challenges, or more precisely, predicts the way a fall-back solution must be designed for the polling stations.

The fall-back solution is to cover the event that a polling station loses its connection to the outside world (or more precisely looses the connection to the on-line central electoral roll). In this case the intention is that votes can still be casted on the polling station.

However, when voters are no longer belonging to a specific voting district they can vote at any polling station within the municipality. A voter can already having casted his vote on another polling station when he present himself at a new polling station. If the previous vote was casted while the new polling station had lost its connection with the central electoral roll, the mark-off of the voter in the central electoral roll will not be visible in the local copy of the electoral roll. So even if there is a local copy of the electoral role, you cannot be sure if the voter has already casted a vote that has been accepted and stored in a ballot box or not.

We see no other solution to this challenge than a fall-back solution which involves the usage of votes in special cover at the polling station that has lost its communication towards the central electoral roll. Thus, all voters that presents at the polling station during the period that a polling station have lost the connection with the central electoral roll, will need to be registered and cast their votes in special covers. These votes will then have to be evaluated and accepted or rejected when the polling station restores the connection to the electoral roll with the same type of procedure as for other votes in special covers.

Thus, our solution will support the following functionality to handle the fall-back solution:

- There will be a local copy of the electoral roll at each polling station.
- The poll-book application will have local functionality (which is available in the event of lost communication with the central system) to register ballots in special cover. A new type of event for

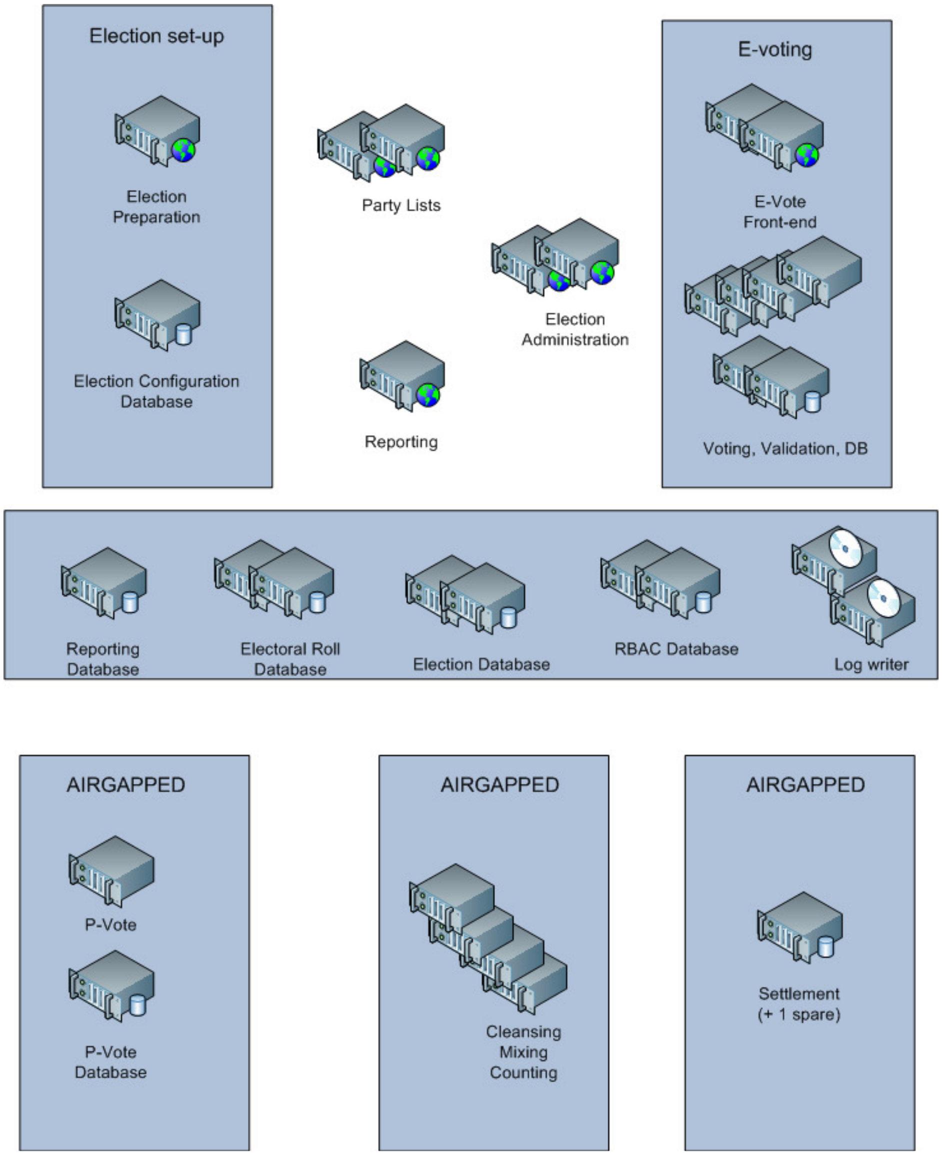
ballots in special cover is introduced to indicate that the vote has been casted in a period with no communication with the central electoral roll.

- When the communication is restored, the local register of ballots in special cover is automatically restored with the corresponding register in the central administrative system. In addition the local copy of the electoral roll is synchronized with the mark offs that has been done in the central electoral roll during the period.
- eVoting at the polling station when communication with the central electoral roll has been lost, will not be allowed.

5. Hardware Platform

5.1. Overview

The overview includes the server hardware required. Client PC hardware is not included with the exception of Scanning hardware. A number of client PC's for the use of Election Administration, eVoting, and Poll Book must be anticipated in addition. These PC's will require smart card reader.



5.2. Hardware inventory

5.2.1. Component suggestions

Server type	Suggested hardware	Usage
Mid-size	2 CPU Quad Core	Application servers
Large	4 CPU Quad Core	Database servers and servers requiring extensive computation power
Storage system	Storage Area Network (SAN)	Data storage
Small	1 CPU dual core	Scanning servers (for scanning centers)
PC	Personal computer	Applications (web, desktop, scanning)
Scanner	ISIS / Cofax scanner	Scanning of ballots

5.2.2. Hardware list -- Full Election Rollout

Hardware	Number of items	Suggested hardware	Operating system	Fail-over	Scaling comments
Election preparation	1	Mid-size	Red Hat Enterprise Linux		
Election configuration database	1	Mid-size	Red Hat Enterprise Linux		
E-vote front end	2	Mid-size	Red Hat Enterprise Linux	Yes	Load-balance
E-vote back end	4	Large	Red Hat Enterprise Linux	Yes	Scaled for encryption
E-vote DB	2	Mid-size	Red Hat Enterprise Linux	Yes	
Party list application servers	2	Mid-size	Red Hat Enterprise Linux	Yes	High availability – visible for many users
Election Administration application server	2	Mid-size	Red Hat Enterprise Linux	Yes	
Reporting application server	1	Mid-size	Red Hat Enterprise Linux		
Reporting DB	1	Large	Red Hat Enterprise Linux		
Electoral Roll DB	2	Large	Red Hat Enterprise	Yes	Load-balanced for

			Linux		large amount of transactions during voting
Election data DB	2	Mid-size	Red Hat Enterprise Linux	Yes	
RBAC DB	2	Mid-size	Red Hat Enterprise Linux	Yes	
Log writers	2	Mid-size	Red Hat Enterprise Linux	Yes	
Cleansing, mixing and counting	4	Large	Red Hat Enterprise Linux	Yes	Scaled for deciphering of large amount of votes
Mixing and counting DB	2	Mid-size	Red Hat Enterprise Linux	Yes	
Settlement	2	Mid-size	Red Hat Enterprise Linux	Spare	
Storage arrays	2	SAN storage		Yes	

5.2.3. Hardware list scanning -- Full Election Rollout

Hardware	**Number of items**	**Suggested hardware**	**Operating system**
Scanning work station	300-600	PC	Windows
Scanners	300	Scanner	
Scanning servers	50	Small	Windows

In the requirements (GR3.6), it is stated that the full rollout should cover 200 scanner centers and that each scanner center should have 3 scanners. From our experience, a large portion of the scanning installations would manage well with 1 scanner and a connected PC. Above we have therefore assumed that there are 50 large scanning installations that each requires 3 scanners connected to 1 PC each + 1 scanning server. The other 150 scanning centers are small and manage with 1 scanner and 1 connected PC.

The municipalities are expected to provide PC hardware and small servers for the scanning centers.

5.2.4. Hardware list -- 2011 pilot

Hardware for election administration and e-voting	**Number of items**	**Suggested hardware**
Election preparation	1	Mid-size
Election configuration database	1	Mid-size
E-vote front end	2	Mid-size
E-vote back end	4	Mid-size
Party list application servers	0	Combined with E-vote front end
Election Administration application server	2	Mid-size
Reporting application server	0	Combined with Election Administration application server
Reporting DB	0	Combined with Election Administration application server
Electoral Roll DB	0	Combined with Election administration data DB
Election administration data DB	2	Mid-size
RBAC DB	0	Combined with Election administration data DB
Log writers	1	Mid-size
Cleansing, mixing and counting	2	Mid-size
Mixing and counting DB	0	Combined with Cleansing, mixing and counting server for pilot
Settlement	1	Mid-size
Storage arrays	0	Servers have embedded storage with redundancy

Hardware for e-counting	**Number of items**	**Suggested hardware**
Scanning work station	11	PC
Scanners	5	Scanner
Scanning servers	3	Small

We suggest that the 5 pilot scanning installations are 2 small installations with 1 PC and 1 scanner and 3 large installations with 3 PCs, 3 scanners and 1 scanning server each. The Contractor may opt for more small installations, if the municipalities chosen are small and have lesser performance requirements.

6. Software stack

6.1. Components

The software stack consists entirely of open source software, with the exception of the scanning solutions. User interfaces are developed with HTML, CSS, and JavaScript utilizing Java JSF web framework. In addition to Java JSF, a presentation framework will be used. Potential candidates are JBoss Seam, Spring, or Apache Wicket. Applications are written in Java utilizing open source libraries. PostgreSQL is chosen as the open source database.

6.2. Application Layering

For application layering, a standard three layer approach will be used:

- Application (presentation layer and logic layer as defined by the presentation framework)
- Middleware (secure web service transaction engine)
- Persistence (and database)

The middleware component manages transactions between the application layer and the persistence layer. It will perform the necessary logic necessary to: validate the transaction, authorize the transaction, transformations, and trigger relevant events (log generation, workflow initiation).

The middleware software will be implemented with Mule, JBoss ESB, or internally developed with open source libraries. Transactions will be clearly defined secure web services in regards to function and boundaries. The Persistence layer will consist of the open source database (PostgreSQL) and most likely Hibernate open source.

6.3. Implementation of the domain model

The domain model will be implemented with focus on isolation of resources both logically and physically. For instance, the Electoral Roll domain will contain a database and middleware/persistence that only supports the function of Electoral roll. We perceive that this will run in its own operating environment, where only software related to Electoral Roll will be permitted to run.

This will create clear boundaries and simplify the task of validating and auditing the system. Transactions in and out of the domain will be clearly defined web services that can be controlled in regards to what external entities are communicated with and the information transmitted. Some domains, such as Settlement, will be physically isolated and disconnected from the network. However, the same application layering principles outlined will also be used in this environment.

It may be applicable to use desktop applications in some circumstances; this is mainly relevant for applications that must be available off-line (Poll Book).

6.4. Framework data validation

Data validation will be performed in the front-end and in the back-end. In the front-end we will utilize the functionality for this in the presentation framework. In the back-end we will most likely use Hibernate Validator for this purpose.

6.5. Support for internationalization (i18n)

Java Resource Bundles will be used for internationalization. This will also be supported by the chosen presentation layer. The Bundles are populated from the Election setup database.

6.6. Server software

All software in this section is open source, with license described, unless other is specified.

Component	**Product**	**Version**	**Vendor**	**License**
Web server	Apache HTTP server	2.2	Apache Software Foundation	Apache License 2.0
Application Server	GlassFish	3.0	Sun Microsystems	CDDL, GPL
Database	PostgresSQL	8.4	Postgres Global Development Group	BSD
Middleware (*)	JBoss ESB	4.7	Red Hat	LPGL
Middleware (*)	Mule	2.2	Mulesoft	CPAL
Persistence	Hibernate	3.3	Red Hat	LPGL
Data Validation	Hibernate Validator	3.3	Red Hat	LPGL
Reporting	JasperReports	3.0	JasperSoft	GPL
Reporting	iReport	3.6	JasperSoft	GPL
Logging	slogger		Scytl	E-voting core system
Audit	E-voting log viewer	4	Scytl	E-voting core system
E-voting	E-voting Core System	4	Scytl	E-voting core system

(*) These are candidates, and one of the software solutions will be selected. The middleware component may as an alternative be developed internally, to minimize the code foot print.

6.7. Scanning software

Component	**Product**	**Version**	**Vendor**
ICR/OCR software	ReadSoft Documents for forms	5.2	ReadSoft
Database	SQL Server	2008	Microsoft
Operating system	Windows server	2008	Microsoft

We have not priced the Windows operating system and SQL server in Appendix 7. We know that the municipalities have access to the KS Select D agreement for Microsoft software, and that the majority have purchased these products over this agreement. The municipalities may therefore utilize existing licenses, but some may need to purchase additional licenses. Due to the large user base, it is difficult to predict the additional licenses needed.

6.8. Client platform and software

Component	Product	Version	Vendor
PC client (*)	Windows	XP, Vista, 7	Microsoft
Web browser (**)	Internet Explorer	7, 8	Microsoft
Web browser (**)	Mozilla Firefox	3	Mozilla Foundation
Web browser (**)	Opera	9	Opera Software

(*) It is not a strict requirement that PC clients are Windows in other cases than for scanning.
(**) This is browser support for the Administrative system and pVoting support. The eVoting client will support a larger number of options as explained in SSA-U Appendix 2A, elaboration of requirement AU2.

6.9. Development tools for the presentation layer

All software in this section is open source, unless other is specified.

Tool	Product	Version	Vendor	License
Java Development Kit	Java JDK	1.6	Sun Microsystems	Closed, but generates general source code
Integrated Development Environment	Eclipse	3.5	Eclipse Foundation	Eclipse public license
Web Framework	JSF (part of Java JDK)	2.0	Sun Microsystems	Other: closed but generates general source code
Presentation Framework (*)	JBoss Seam	2.2	Red Hat	LPGL
Presentation Framework (*)	Spring	2.5	SpringSource	The Apache Software License, Version 2.0

Tool	Product	Version	Vendor	License
Presentation Framework (*)	Apache Wicket	1.4	Apache Software Foundation	The Apache Software License, Version 2.0
Performance test	JMeter	2.3.4	Apache Software Foundation	The Apache Software License, Version 2.0

(*) These are candidates and one for the frameworks will be selected.

6.10. Other tools and libraries

All software in this section is open source, unless other is specified.

Tool	Product	Version	Vendor	License
Cryptography	Bouncy Castle	1.43	Legion of the Bouncy Castle	Adapted MIT X11 License
Bug tracking	Bugzilla	3.0.5	Mozilla	Mozilla Public License
Test Management	Hudson Continuous Integration Server	1.329	Sun Microsystems	MIT license
Build Management	Maven	2.2.1	Apache Software Foundation	The Apache Software License, Version 2.0
Version Management	Subversion		Tigris.org	Subversion License
Performance test	Jmeter	2.3.4	Apache Software Foundation	The Apache Software License, Version 2.0
Profiling	Rational Purify Plus	7	IBM	Other: commercial, but used only for profiling purposes.
Code analysis	FindBugs	1.3.9	Sourceforge	GNU LGPL
Test coverage	Cobertura	1.9.3	Sourceforge	Apache Software License, Version 1.1, GNU General Public License, Version 2.0
Standards compliance	Checkstyle	5.0	Sourceforge	GNU LGPL

Functional tests	Jemmy	V2	Java.net	Common Development and Distribution License
Development kit for e-counting	MS Visual Studio		Microsoft	Other: commercial

7. Requirements for Open Source and Closed Source

7.1.1. Open Source (MC2)

The software stack is entirely open source, with the exception of the paper vote e-counting system.

User interfaces are developed with HTML, CSS, and JavaScript utilizing JSF web framework. Applications are written in Java utilizing open source libraries. PostgreSQL is chosen as the open source database. Linux Red Hat, which is a generally recognized open source license, is chosen at the operating system.

This gives the Customer the possibility to make the source code of both the administration system and the e-voting system available to the public and allow anyone to copy, modify, inspect, compile, debug and run the core systems for testing purposes.

7.1.2. Closed Source (MC5)

As described earlier in this appendix, the e-counting system is closed source. The paper vote e-counting system consists of third party ICR/OCR software (from Readsoft), of modules developed by the Contractor on top of this ICR/OCR software and of a database.

The modules developed by the Contractor comprise interpretation, verification, counting of votes and transfer of votes to the administrative election system. The environment for the e-counting system is Windows – Windows operating system, SQL server and MS Visual Studio for development.

As stated in the requirement MC2, and in answers to a question posted on the Q&A about the requirements for e-counting, we understand that the e-counting (scanning)-system might be closed source (paper audit trails will exist).

Despite of this, we have looked into some open source alternatives, as we understand the preference of open source over closed source. We have not been able to find an open source alternative that has, in our opinion, a well proven track record similar to the ICR/OCR software from Readsoft.

The paper vote e-counting is a crucial task in an election. Tuning ballots and setup rules for the scanning software is a difficult and time-consuming job. ErgoGroup have used a scanning solution based on Readsoft in every election since 1999. In contrast, the open source alternatives we have considered have less functionality than ReadSoft. This will create considerable risk in regards to more development work and complications to the scanning process.

7.1.3. Implementation environment (MC6)

The hardware and software requirements for the total election system are described and listed earlier in this appendix. A Hosting Partner would have to provide an infrastructure for the system, including firewalls, load balancers, potentially SSL accelerators, and more. The Contractor suggests an investment into a staging environment. As an additional benefit, some of the servers in the staging environment may be used for up-scaling the production environment during the most intensive part of the election/pre-election period.

An installation guide for the Hosting Partner will be provided with the software. The Contractor believes it is important that the Hosting Partner offers a managed service (not only co-location of the servers), and familiarize themselves with the system in cooperation with the Contractor to get a full understanding for the solution. As stated in the SSA-Maintenance agreement, the Contractor believes that well-functioning partnership between the Application Management provider and the Managed Hosting Provider must be a priority. Among other things, the Application Management Provider will need statistical data from the Hosting Provider to perform problem analysis and tune the Election System. In regards to Application Operations -- that might be performed by either the Hosting Provider or the Application Management Provider, we suggest the latter.