ATTACHMENT 1 to EXHIBIT A: REQUIREMENTS
"Supplier's Supplemental Word Document to Attachment A"

4. Requirements – Statement of Needs

The Canton Group has completed Attachment A – providing detailed information related to the Operational and Technical Performance Standards identified in the RFP for a solution that meets the needs of Commonwealth Stakeholders for a stable, accurate, secure and highly functional Virginia statewide voter registration and election management system (SVRS), where performance meets workload demands without degradation.

4.1 Operational Standards

4.1.1 Multi-jurisdictional

SVRS is a multi-jurisdictional Voter Registration and Election Management System that supports independent conduct of election business by one hundred thirty-three (133) localities under central administration of ELECT. A workflow example that touches multiple jurisdictions is found with automated Duplicate voter record management.

1. SVRS receives a new voter registration from Department of Motor Vehicles, the incoming registration is for a voter who lives in City of Richmond.
2. During the registration process, SVRS links a single high confidence duplicate match for the voter's prior registration in the City of Virginia Beach.
3. The voter registration is updated to their Richmond address with the same voter ID and automatically assigned their precinct within City of Richmond.
4. The system generates a Voter Notice Card for the voter to let them know via mail they are now a voter in Richmond.
5. The City of Virginia Beach receives a no-action system notification of the voter's move out of the locality.
6. The system queues a notification for Virginia Beach to inform the voter of their move out of the locality.
7. The Voter can view the current status of their registration on ELECT's Online Citizen Portal (OCP).

4.1.2 Adaptable

Our system is adaptable to future statutory, regulatory, policy, or technology changes in several ways:

- This is accomplished via System Configurations (Business Rules, Data Validation, Correspondences, Templates, Code Types), External Integrations (API-based), or Customization. This allows the system to be prepared to be compliant with future changes in the Election Law.
- Technical changes: SVRS Reactive Architecture allows the system to be responsive, scalable and resilient. A container-based deployment model uses orchestrators to monitor services allowing the system to adapt to changes in the number of users to provide stable performance behavior all the time.

The SVRS Microservices architecture consists on independent deployable components (Microservices) that implements functional domains with immutable end points. This allows us to replace the implementation of a Microservice with another that provides newer or better technology without disturbing the other Microservices of the system. New functionality is implemented as a Microservice, so it can be maintained and deployed without altering the surrounding Microservices.

4.1.3 Auditing

SVRS provides a robust tracking of changes, events, and errors. The logs provide the context of who, when, what values before/after, and along with reason for change.

Some examples of logs provided in SVRS:

- Voter changes
- User Access logs (e.g., searches where confidential voters were included)
- Data extracts and criteria used to extract
- Geographical boundary audit logs
- System access logs
- User-Friendly, Actionable Error Logs at the Application Level
- Bulk Change Logs (list maintenance)
- System automated changes (e.g., Ballot Suspend and Reissue, High-Confidence Duplicate match)
- Logs at service levels
- Monitoring and application.
- Infrastructure logs (monitors services to ensure 24/7 accessibility)
- Audit logs of reports generated via batch jobs include information such as who, when the report was scheduled, report name, description, report start, end date, status of the report, etc.

4.1.4 Configurable

Team Canton understands that laws changes affect pre-established work flows. SVRS's built-in work-flow engine can change execution path based on received input values. Administrator can change the input values to adopt the path of workflow to meet the change in law.

The configurable options within SVRS includes (not an exhaustive list) – business rules, data validation rules, correspondence templates, document templates for auto read, and standard codes.

If a new law "Allow early registration starting at 16 years" is approved in the future, a SVRS user can:

- Add a new Pending Status Reason code of "Underage" for the entire state
- Modify the Early Registrant Minimum Age configuration to "16"
- Setup a new correspondence template called Underage Voter Registration Card (U-VRC)
- Set system Rules to trigger the U-VRC correspondence for new Underage voters' registrations
- With the SVRS setup in place, ELECT and Localities can focus on voter education and not on running design sessions and testing.

4.1.5 Data Synchronization

SVRS provides a “single source of truth” on voters, ballots, and elections for all stakeholders of Commonwealth of Virginia. SVRS is a centralized voter registration and there is one record per one voter (1-1).

SVRS’s Partner Data Integration Services supports real-time and batch ingestion of data from multiple sources including Department of Motor Vehicles (DMV), Virginia State Police (VSP), Bureau of Vital Statistics (BVS), and Federal Courts. SVRS supports plug-able interface to receive real-time voter data from validated and authorized data sources such as DMV and NVRA agencies. SVRS implements Master Data Management using Machine Learning and Data Science based ****** Service Agent to match and merge duplicates of voters, felons, deceased, non-citizens, and adjudicated incapacitated while preserving the data lineage.

4.1.6 Legal Requirements

The Team Canton leadership team has over 15 years of experience in implementation and support of voter registration and election management systems that are compliant with federal, state, and local statutory requirements. Our team has worked with California, Colorado, Mississippi, Maryland, Wyoming, Washington, Ohio, New York, New Jersey, Rhode Island, Idaho, and Oregon on such system. SVRS is designed to adopt to the changing election laws and regulation at federal, state, and local level.

Our experience in IT project implementation has shown that state system requirements are often drafted to meet the federal, state, and local statutory requirements. Team Canton considers ELECT’s Centralized Voter Registration System requirements as the baseline to meet federal and Commonwealth of Virginia statutory requirements. Team Canton’s business team helps ELECT to draft acceptance criteria for the requirement to aligns with the laws. The acceptance criteria are then used to test and ensure the delivered SVRS meets both legal and system requirements.

Additionally, SVRS supports locality level configuration to meet local statutory requirements. Examples of local legal requirements envisioned includes languages supported, poll worker contact, and election for local districts and offices.

4.1.7 Usability

SVRS provides a high-usability user interface by balancing the requirement of experienced users of system to perform data entry tasks efficiently and the need of new users to understand and learn the system easily. SVRS’s user interface is built by following user interface style guides and standards that address factors such as Task efficiency, Ease of learning, Ease of remembering, Understandability, and Subjective satisfaction.

Here are some highlights of usability features incorporated in SVRS –

- A modern and reactive screen layout that adjust automatically to different screen sizes and reduces the fatigue on the user.
- Use of quick search on top navigation bar to look up voter and ballots from anywhere in the system.
- No need for SQL queries, SVRS provides powerful search features that allows for nearly endless combination of search results across locality or state, depending on user privilege.

- Use of dashboards to notify of outstanding tasks or tasks past due and ability to navigate to that task and process them.
- Use of keyboard short cuts for Next, Previous, Open, Close, Accept, Reject, Save, Search, and so on.
- Use of standardized memorable screen layouts across the system.
- Use of breadcrumbs, wizard styles, and so on to hand-hold the user through less intuitive multi-user/multi-step tasks.
- Use of intelligent record locks to allow one or more users to collaborate on batch tasks.

4.1.8 Longevity

SVRS runs on FedRamp certified Cloud. A cloud-based deployment allows continual upgrades, server patches, keeping up-to-date with changing security needs and recommendations. Versioning by microservice is important to ensure SVRS is always using the latest version. Ramping-up server performance as needed, security upgrades, server upgrades, system functional upgrades to adhere to laws, compatibility for external integrations, compatibility with third party tools utilized in SVRS. The SVRS team performs .NET Core upgrades to keep the technology stack relevant. As a result, the product can evolve functionally and technically, ensuring the SVRS tools and components relevancy in the upcoming decades.

4.1.9 Reporting:

SVRS Election Analytics offers ELECT and Localities the best in class, near-real time, ease of access to any voter, election, ballot, candidate, petition, and other information within the solution. SVRS Election Analytics includes Standard Reports, Ad-hoc reports, and Advanced Analytics. SVRS provides drag and drop capability to building reports and sharing it within team. Access to data can be secured at the column and row level including Personal Identifiable Information. Consumption of analytics services does not impact the performance and operation of other parts solution. SVRS is built on an industry leading underlying technology, ******, that is already familiar to ELECT.

4.1.10 Performance

SVRS architecture and implementation based on microservices deployed in Docker containers and managed by Kubernetes cluster(s) is highly scalable. Kubernetes will scale the containers based on their capacity (memory, processors) and how they are handling their load. The container manager monitors the containers and when it notices that one of them is under performing it automatically spins out a new container to share the load of the slow container. When the load is reduced and the containers are not used at all, Kubernetes disposes the extra created containers. This elastic behavior of our architecture (application and infrastructure) assures that the SLAs required by the client will be met. The SLAs will be defined in conjunction with the client based on the user expectations. The metrics will be determined during performance tests in the actual production environment or a production like environment for performance testing. We will fine tune the initial number of Microservices replicas to achieve an optimal performant system that meets from day one the SLAs. The scalability of the system can also be done manually using the Azure Kubernetes configuration and monitoring views, but we encourage the use of the automated scalability capacity offered by Kubernetes and our Microservice architecture. We have been scalable to meet performance to support 1500 concurrent users,

Official list extracts of 1 million records every 10 minutes, 100 hundred county transactions every one minute, etc. We will help identify similar metrics for this project.

4.1.11 Workflow Management

SVRS's workflow engine supports orchestration of work-flow path using configurable components such as Business Rules Engine, Data Entry Assist, Custom Notices, and Standard Values. Here is an example for workflow within SVRS: -

- Voter Registration Team/Data Entry Assist either manually or automatically process the voter registration data from paper, online, or DMV queues.
- List Maintenance Services queues duplicate matches, felon, and deceased matches. High confidence matches are automatically merged, moved, or canceled as per business rule. Potential matches are available for Voter Registration Team to review and work them.
- Election Management Services based on voter eligibility rules adds, updates, or cancels voter on the upcoming election's workspace.
- Ballot Services automatically issues, suspends, or reissues ballots as per business rules. The Election Team can print the ballots ready to be mailed out.

4.2 Performance Standards

4.2.1 Business Continuity / Disaster Recovery

SVRS solution to be available 99.99% uptime, our strategy started at design time, development and during implementation following a Reactive Architecture: Fault Tolerant, Responsive and Elastic using Messaging for inter service communication. Our solution implements a replica of the Production environment in a Recovery Site physically located in a different Azure Geographical Region. We will implement Azure Recovery Site (ARS) to replicate Virtual Machines, ****** to back up the replicated Data Base and Azure Backup Services to replicate our Data Repository, resulting in a RPO of 60 minutes or configurable to the client needs.

4.2.2 Database of Record

SVRS is a centralized repository at the state-level with a unique voter id; however, each locality has maintenance control of its subset of voter data that is logically separated from other localities who do not have jurisdictional control over the record. This allows elections to be conducted at a statewide or locality-wide level. Due to this approach, the most up-to-date data is available instantaneously, while maintaining control of the data modifications.

4.2.3 Data Compliance

SVRS incorporates number of following important design features that supports compliance with data tracking and reporting requirements for the Commonwealth –

- Data tracking and control entry error – SVRS already tracks data fields necessary for the:
- Americans With Disabilities Act (ADA)
- Election Assistance Commission (EAC)
- Election Administration and Voting Survey (EAVS)
- Geographic Information System (GIS)
- National Voter Registration Act (NVRA)
- Military and Overseas Voter Empowerment (MOVE) Act
- Voters Information Project (VIP)
- Uniformed and Overseas Civilians Voting Act (UOCAVA)

In addition, entry of bad or inconsistence data is controlled with use of standard values, validation rules, and predefined data transformations.

- Point-in-time reporting: Most of these reports are generating as ‘As-of’ a date so from the get go, SVRS support tracking and reporting as of a point in time. SVRS also supports generation of reports that compares data changes in a period.
- Readiness Check and reconciliation: In most cases the above-mentioned reports are generated at the ELECT level thus is important to ensure localities have completed the task necessary before reports are extracted. Once the reports are generated, Locality should be able to reconcile the counts with what ELECT has. SVRS supports both readiness check and reconciliation.

As the data tracking and reporting requirements change, Team Canton works with ELECT to make the system changes to remain compliant with data tracking and reporting requirements.

Team Canton's IT Security Policies and Guidelines for system and organization was prepared by independent IT Security contractors to comply with appropriate standards and best practices as published by National Institute of Standards and Technology (NIST) and International Organization for Standardization (ISO) including -

- FIPS 199 Standards for Security Categorization of Federal Information and Information Systems
- NIST 800-18 Guide for Developing Security Plans for Federal Information Systems
- NIST 800-53 Recommended Security Controls for Federal Information Systems and Organizations
- NIST Privacy Impact Assessment Plan
- NIST SP 800-34 Contingency Planning Guide for Federal Information Systems
- ISO 27001 Information Security Management System (ISMS)

Team Canton follows the Commonwealth Information Technology Resource Management (ITRM) recommendations for Cloud based services, by hosting SVRS in the Public Fed Ramp Certified Azure Government Cloud.

4.2.4 Data Reporting / Warehouse

SVRS provides the best match and the superior choice for ELECT with our Elections Data Analytics Platform. Our analytics platform is built on ****** Datawarehouse and ****** for reporting thus ELECT and Locality staff can continue to use ******. Team Canton publishes data models that election staff can understand and use easily. Staff can drag and drop the necessary fields, without worry of data relationships and SQL queries, to produce beautiful reports and dashboards. SVRS supports multitenancy and security is applied at both row and column level thus each locality can keep their data private and accessible to their own staff and ELECT. In addition, ELECT and locality can securely connect any other business intelligence solution such as PowerBI or ******, Looker to our warehouse if they choose to. Truly a game changer and unlocks the power of your data.

4.2.5 Product Roadmap

SVRS utilizes industry-standard technology, such as Microsoft development stack. Team Canton actively avoids the use of open-source components, choosing instead reputable, commercial operations with existing support models.

SVRS prefers relationships with companies that have existing government presence to ensure SVRS's long-term product viability and continued improvements to the product's roadmap.

An example of how SVRS utilizes widely used technology is with the integration with ArcGIS. This includes an established partnership with the company ESRI. The main components in SVRS's technology stack are supported by Microsoft's development stack (SQL Server, Azure DevOps, Azure Kubernetes). Smaller components, such as the scheduler Hang fire and messaging service RabbitMQ are both widely used and established in the marketplace. Team Canton evaluates component technology providers based on the product roadmap for the companies that provide the tools utilized in SVRS.

This includes .NET (at current or one prior version), major technical components are sourced from DevOps GIT repository, Microsoft Pipelines for continuous deployment, containers, dockers, container orchestrators, relationship db, Microsoft SQL server. Security resources such as Firewalls, architecture is service-oriented microservices which allow flexibility to future change, and add components without altering existing components. Because the system is hosted on Fed Ramp certified Cloud environments hosted by major companies, allows SVRS to scale to resource loads as needed (such as two weeks prior to an election, etc.). Job scheduler, system monitoring logs and health of the system.

4.2.6 Cloud Ready Exec Order-19

The system is intended for Azure Fed Ramp-certified Cloud environments; however, it can be supported on hosted environments or a hybrid version where the database is locally hosted and services are deployed in the cloud. Cloud-hosted environments allow SVRS to scale to resource loads at critical periods or over the course of time.

An example of this can be seen two (2) weeks prior to a major election when a sudden spike of users (temporary data entry users, election workers) begins logging into the system.

Scaling the load between services depending on different time periods (e.g., Election service impact, Petition verification, Voter Registration drives). Spin up additional resources (e.g., microservices), scale down again to maintain costs. Team Canton monitors SVRS to determine the health of the system at any point in time.

Determining the ideal configuration services to be maintained in an ideal state is monitored automatically and supervised by a sysadmin. The SVRS team provides technical consultation for the effective scaling of the components of the system. Services are separated to maintain a better control of server resources. Performance test-based determination is key to determine the initial configuration of the system and predict how to scale the system appropriately. All this procedures and practices helps manage the cost of the cloud services.

4.2.7 Security

Security compliance is a part of Team Canton's design methodology. The SVRS Security model transcends all layers of the application: Frontend, Application, Information and Infrastructure tiers.

***Redacted**

4.2.8 Integration

SVRS's Partner Data Integration Services supports real-time and batch ingestion of data from multiple sources including Electronic Registration Information Center (ERIC), Virginia Department of Motor Vehicles (DMV), Virginia Department of Health (VDH), Virginia State Police (VSP), National Change of Address (NCOA), and other internal and external agencies.

Batch Data Integration service uses a scalable design for import of high volumes of data with fast load times from new interfaces. This service includes components to support capture of metadata, movement of data between servers or networks, data transformation, and scheduling.

Real-time Data Integration Service is a decoupled model of integration defines an Integration Service (Service Bus) - ISSB for SVRS to consume external agency web services. This microservice is fully autonomous in processing all the steps needed to integrate SVRS with external agency web service. It implements data transformations, protocol transformation, messaging, routing design patterns or Enterprise Integration Architecture (EAI), as well as industry common application design patters such as Singleton and Factory Method.

4.2.9 Configuration Capability

Team Canton built SVRS to be operated as a Software as a Service (SaaS) model. The intent behind this is to incorporate configurability widely-used throughout the system. ELECT and locality can maintain configuration locally or inherit through hierarchy. The configurable options within SVRS includes (not an exhaustive list).

• Business Rules Engine: Supports authoring of data validation and business rules with minimal code changes.

• Workflow Engine: Supports orchestration of work-flow path using configurable components such as Business Rules Engine, Data Entry Assist, Custom Notices, and Standard Values.

• Data Entry Assist: Intelligent assistant to complete data entry from paper documents. Supports setting up templates for new documents to automatically read data from.

• Custom Notices – Supports setup of custom templates to print or extract notices. Prints with jurisdiction specific details. Previous versions of the documents can be preserved.

• User Interface: Supports configuration at jurisdiction level with specific logos, dashboards, office locations with hours, contacts, and so on.

• Standard Values: Supports creation of standard values at both state and local level. Local level values are automatically mapped to default state values on search, view, and reports.

4.2.10 Modern Architecture

SVRS is architected using Service Oriented SOA principles and the Reactive Architecture paradigm. This modern architecture is based on responsiveness, scalability and resilience so that the different components (Microservices) continue functioning when one of them is not performing well or is compromised. Each Microservice exposes its end points as a RESTful API.

Our DevOps process is designed to automatically build and deploy the application on premises or the cloud using Docker containers, one for each Microservice, Kubernetes and Istio. Kubernetes orchestrates the containers and also provides for automatic scalability and fault tolerance of the Microservices in SVRS.

SVRS client is browser-based user interface that communicates with the SVRS service-oriented backend via REST API.

4.3 ELECT Requirements

4.3.1 Voter Registration

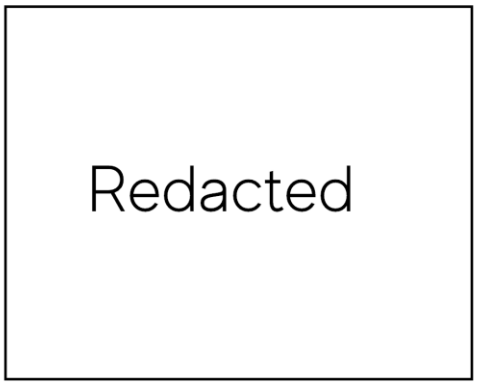
The system ViBE (Voters, Ballots, Elections) will be referred to as SVRS (Statewide Voter Registration System) in this document for the purpose of clarity in our response to the ELECT Requirements. The following sections describe how SVRS meets the requirements set out by the State of Virginia for Voter Registration.

4.3.1.1 VR-1.1: Functionality to Manage Voter Registration

SVRS is a fully-functional elections platform, and contains all elements necessary to conduct a successful election end-to-end. This includes new and existing voter list maintenance (**VR-1.1.1**) (**VR-1.1.3**) (**VR-1.1.13.1**) (**VR-1.1.13**), geographical boundary maintenance, office, referendum, contest, position, petitions, public portal, voting locations, workers, ballot management, election day portal, reporting. SVRS is compliant with NIST specifications (**VR-1.2.1.3**).

In addition to the in-built features, the SVRS platform is designed with integration in mind—external government agencies, reporting bureaus, customer service tools, messaging servers, single-sign on, FTP or any other modern integration needed for the State of Virginia to be successful.

Records who fail to meet HAVA (**VR-1.1.3.1**) requirements are clearly marked on the Additional Information tab.



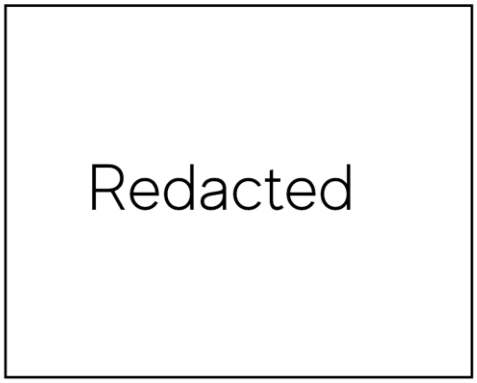
HAVA ID and Validation Fields in Voter

The infrastructure exists, a five-day waiting period to delay the addition of a new voter record into an election workspace to delay ballot issuance is possible, but not currently in SVRS. In SVRS, the voter is validated in real-time (**VR-1.2.1.6**) against Dept. of Health, Dept. of Corrections, DMV and SSA agencies in the present implementation (**VR-1.1.3.3**).

The system supports same-day registration (**VR-1.1.3.4**). For an election, this can be done either by making the Close of Registration for an election the same day as the election, or a process similar to the current implementation by the State of California can be implemented for Virginia, where the ballot envelope must pass through a validation step for voters who register between the close and election day. For first-time voters, the HAVA ID required flag means the voter must vote in person (**VR-1.1.3.2**).

SVRS, by default, contains four primary voter statuses: Active, Inactive (**VR-1.1.13.4**), Pending, and Canceled (**VR-1.1.13.2**). Each status can contain reasons customized by the State of Virginia. Voters can pass from Canceled to Active status assuming they meet the requirements of an active voter record (**VR-1.1.13.3**). Inactive or Canceled records' Reason code can reflect the reason a registration was denied (**VR-1.1.13.6**). Any validations that prevent a voter from becoming Active are presented to the user on the screen (**VR-1.1.9**).

SVRS is a “top-down” system, meaning that all jurisdictions exist together in the same eco-system, but are logically separated from one another. This means that one locality has access to data from their own jurisdiction only. However, voter records can transfer to a different jurisdiction of ownership (**VR-1.1.13.5**). This can be done via the registration process linking an applicant to an existing record anywhere in the state—or by one jurisdiction pushing the voter into another jurisdiction for validation.

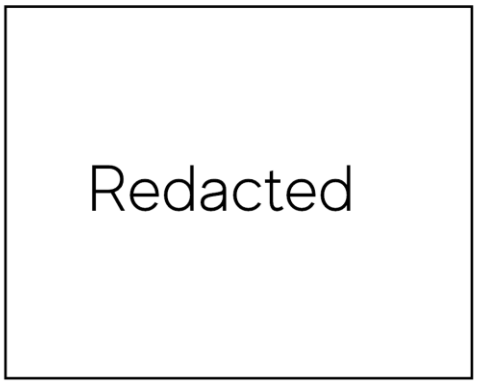


Political Party field – Voter Record details

SVRS tracks qualified, qualifying, and unqualified parties in System Configuration. These can be associated to a voter record (**VR-1.1.11**). “Other” political parties are added as freeform text. These “Other” parties are added into the system as unqualified parties.

In SVRS, only an authorized end user views or updates confidential voter registrations (**VR-1.1.4**) received at the county. These protected voters have their information (residence address, phone number and email address, IDs) masked by default for even privileged users.

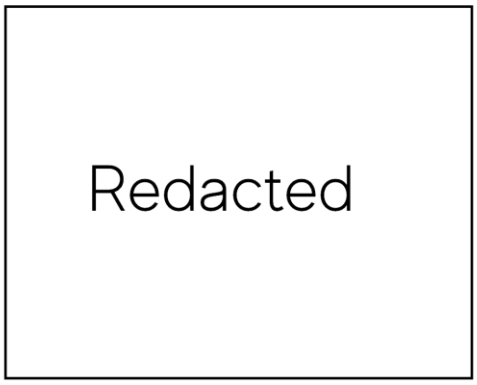
On the voter’s record, it includes an indication of the confidential status along with application date and legal basis. Voter is designated by default as an Absentee (Vote-by-Mail) voter and are required to include a mailing address that is different than residential address.



SVRS Confidential TRUE – Flag, Type, and Date

Access to Confidential Voters in SVRS is managed by user roles. Each role is created with a set of permissions a system administrator can add this to a role, all users assigned this role will have all associated permissions.

Confidential Voter access permissions are under the Voter service section. This permission is labeled “Voter Records (Confidential)” and it can contain any of the following Create, Read, Update, and/or Delete. Read Only privileges allow users to view, but not Update the records. This masks the record for this user everywhere in the system, including other system services utilizing voter service data.



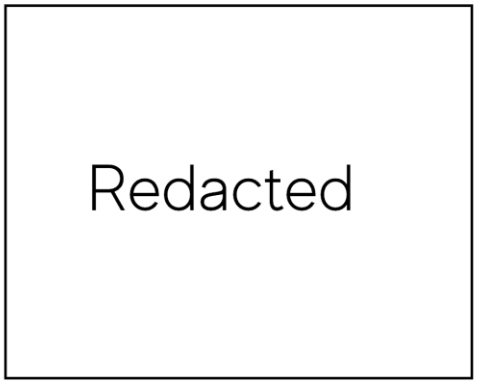
User Administrator “Confidential” Voters

Early Registrants automatically move to Active status on their 18th birthday (**VR-1.1.10.1**) assuming all other validations for an Active voter are met. Pending, Underage voters who are 18 on or before an election day, and who are otherwise eligible, are added to an election workspace.

Minimum age of early applicants is a system configuration item—California has theirs configured to 16, and Virginia would define this per their current laws (**VR-1.1.10**).

SVRS Participation History allows users to add participation for a voter record, define the type of Ballot (ex: “FWAB”), and whether or not it was Counted (**VR-1.1.7**).

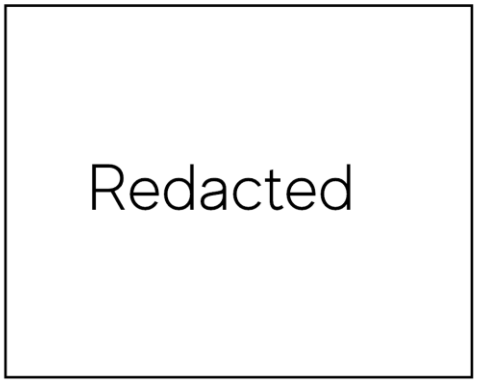
County election staff can create custom scan templates to include any form the county election staff receives, including Federal Postcard Application (**VR-1.1.6**)



Voter record Election's tab

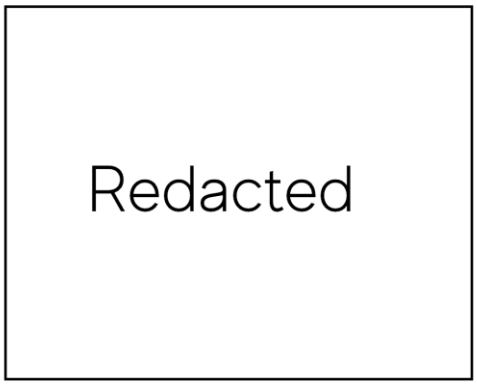
On the voter record's Elections tab, the user can define the types of ballots the voter should receive. In California they have Permanent Vote by Mail, Election Specific Vote by Mail, and UOCAVA (**VR-1.1.5**). For Virginia, the absentee application process would have to be implemented.

For UOCAVA voters, simply indicate the voter is a UOCAVA, define the UOCAVA type, and the Delivery Preference. Mailing Address becomes required for voters flagged as a UOCAVA.



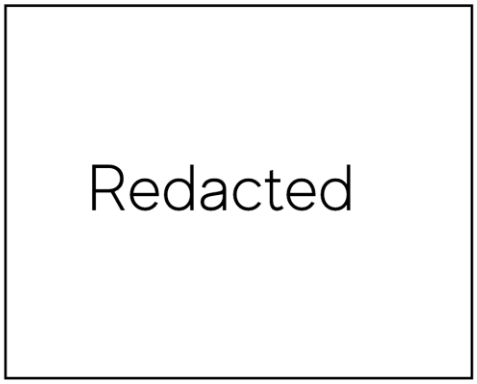
Voter record Election tab - VBM Ballot Section with UOCAVA and Temp (Election Specific) Address

On this same tab, the user can define one or multiple voter Assistance metrics for voters who have a disability or impairment that requires accommodation on the ballot or at the poll site (**VR-1.1.8**) (**VR-1.1.12**).



Voter record Election tab details for Voter Assistance

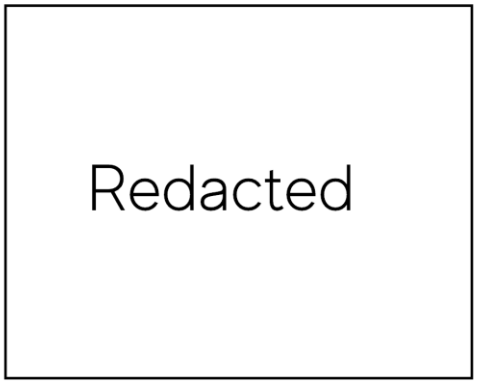
SVRS offers a large number of ways to locate the record (**VR-1.1.2**). On the perennial header at the top of the screen, users can perform a quick search for a voter record based on name, Driver License ID, Voter ID (or scanned barcode of voter id (**VR-1.2.1.6**)), and Ballot ID.



Quick Search with No Records Found prompt

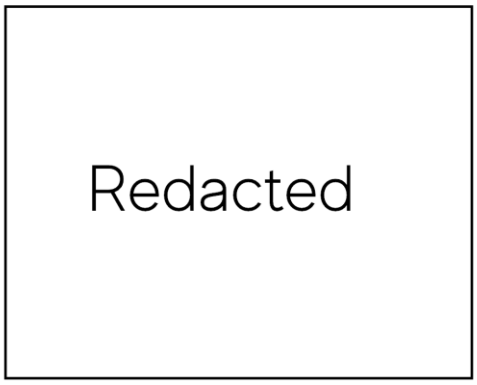
If the record is not found this way, the user can correct a typo and try again, create a new voter record, or go to the Advanced Search.

As with most SVRS modules, the Voter module starts with a search screen. This is the Advanced Search. Every voter data field is available to search on or add as a column. Privileged users can additionally extract results to a CSV comma separated file, the metadata of what and who extracted data is stored in the system.



SVRS Advanced Search

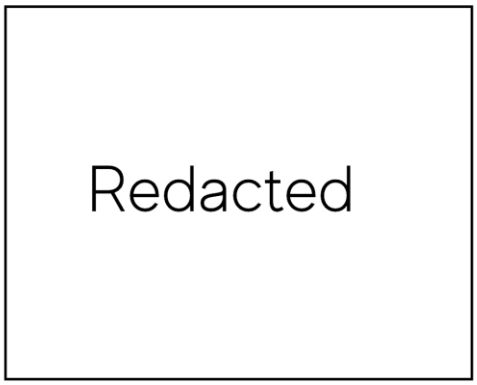
The “Result Columns” search tab, offers the user a drag and drop immediate reordering and adding of voter record data fields. This search not only helps locate specific voters, but acts as a kind of workable list for unique scenarios where creating hundreds of standardized reports would not be beneficial.



SVRS Advanced Search – Modify Result columns and order tab

4.3.1.2 VR-1.2: Registration Data Entry and Processing

SVRS allows system users to scan single paper registration and bulk paper registrations (**VR-1.2.1.1**), or attach saved files individually or add to a bulk queue (**VR-1.2.1.10.1**). The system groups scanned/attached documents into work queues of like documents. These have their Received Date, Source (**VR-1.2.1**), and Method—as well as other details that are used to generate NVRA statistical reporting (**VR-1.2.1.18**).



SVRS Bulk Scan Prompt to collect metadata

County election staff can create custom scan templates to include any form (**VR-1.2.1.16**) the county election staff receives from Voter Response forms, Driver License, DMV forms, Federal Registration Forms, returned responses, and more to create the workable queues of like documents (**VR-1.2.1.15**). Additional metadata data points such as form language can be easily added. Preferred Language is a voter record data element, as is a voter registration field.

Because this data is stored in SVRS, users can utilize the SVRS Data Analytics tools to produce any number of statistical reports, or formal standardized reports can be created as defined by the State of Virginia (**VR-1.2.1.17**).

When performing bulk scans, SVRS generates a thumbnail preview for users to quickly verify if an entire batch is rotated or upside-down to correct it while the user has the physical file close at hand instead of during processing.

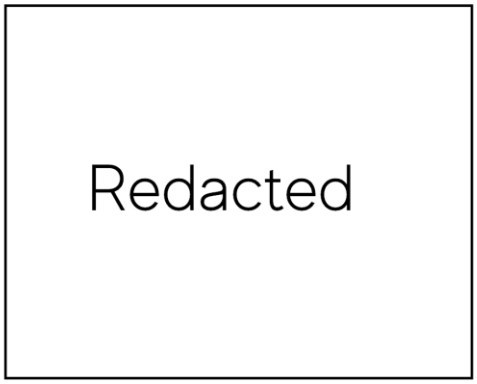


Figure 1 SVRS Bulk Scanned Document Queue

SVRS only locks the individual document in the moment it is processed by a user. This prevents accidental double-entry, but also scales to allow multiple users to collaborate in a single queue to work on pending documents without locking up an entire “scanned batch” with a single user

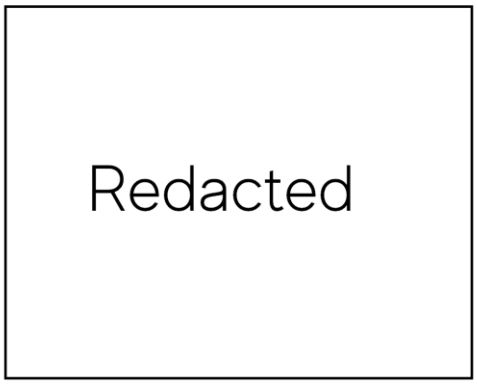


Figure 2 SVRS Unprocessed Scanned Document Queue Details

When processing queues of scanned documents, users can sort to individually select a record or use SVRS’s Process Next feature which follows FIFO (First In, First Out). FIFO selects an unprocessed record based on the oldest received date of the file and ensures older documents are processed first. Users can skip over applicants who have pending information, and leave them unprocessed if that is desired to avoid record creation; however, SVRS includes a Pending status (**VR-1.2.1.11**) for the purpose of managing these applicants and producing correspondences to resolve any missing information (**VR-1.2.1.14**) (**VR-1.2.1.14.1**).

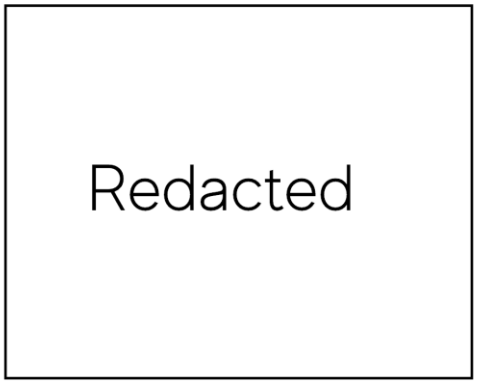


Figure 3 SVRS Voter Registration New Scanned Document Processing Screen

Data Entry for Paper Forms

The SVRS data entry screen layout is side-by-side to ease data entry by the user.

Common data entry functions such as hotkeys (“Ctrl+S” for Save) and key navigation (Tab and Shift+Tab to transverse between data entry fields) are present in SVRS to reduce mouse-dependent navigation. Hotkeys for buttons are expressed in the tooltip, to support users still learning SVRS.

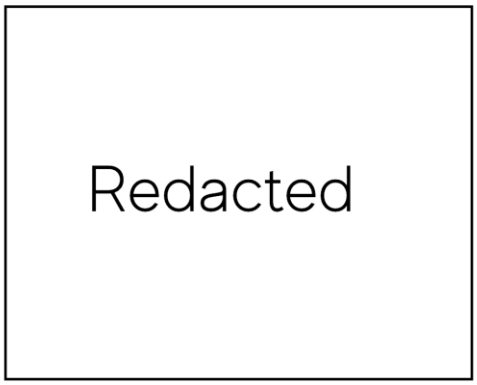


Figure 4 Button Tooltip showing Hotkey

While SVRS allows users to defined signature capture areas, it may be necessary for users to re-crop a signature for a better capture due to extra small or extra-large signatures.

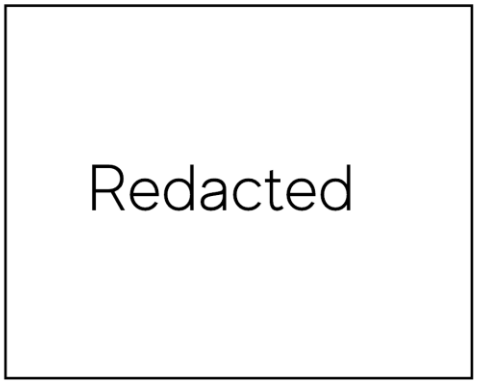
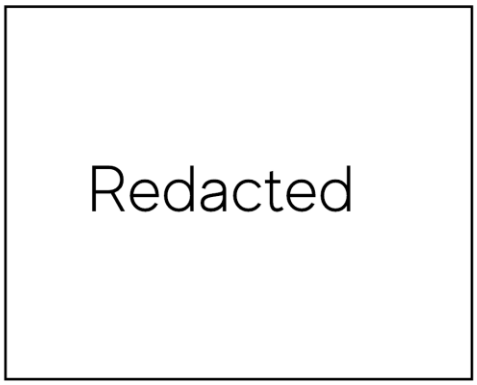


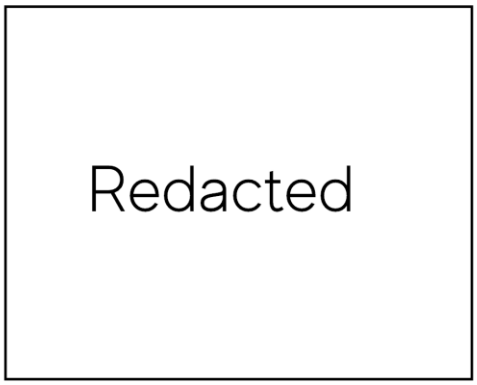
Figure 5 Show a re-crop of a Signature from Preselect

The user can additionally quickly re-scan, re-attached, or rotate the document if necessary.



caption

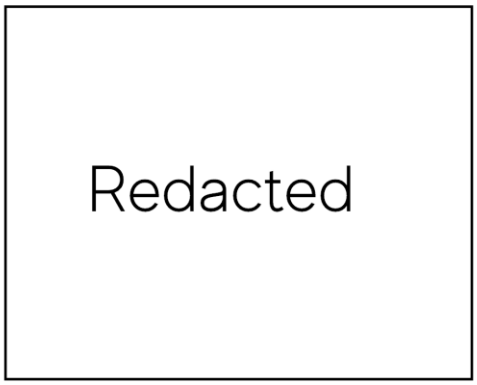
The user enters the data side-by-side to ease data entry. Future roadmap allows for character recognition software for automated data entry.



caption

The user enters in applicant's personal identification information needed which searches for potential duplicate records across the state.

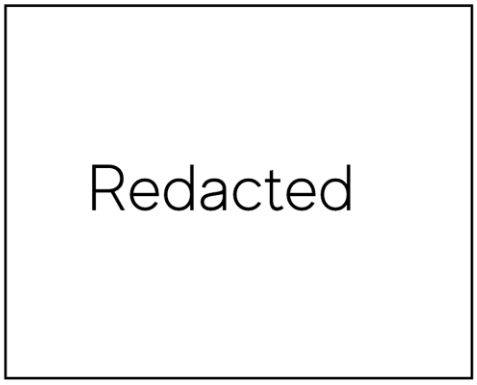
SVRS populates and modifies the voter Status and Reason as data is inputted about the applicant. If they are underage if their residence address cannot be auto-precinct, a first-time registrant, or linked to an existing record all affect the status/reason suggested by SVRS; however, the system user can modify the fields with what they deem the true Voter Status and Reason (**VR-1.2.1.1.1**) at their discretion as well as the signature date by the applicant. The signature date / effective date is used to validate that the registration form or list maintenance being processed is not older than the current record, the user is alerted and can decide to "Ignore" the form (**VR-1.2.1.9**).



Status and Reason during Add/Update

Duplicate Record Prevention Updates to Existing Records

SVRS uses high-confidence matching metrics defined by Los Angeles County and used by the State of California to ensure no duplicate voters exist across their respective 6 million and 22 million count of voter records (**VR-1.2.1.8**). This efficient process verifies if an applicant has already applied or is registered within the entire state (**VR-1.2.1.12**).



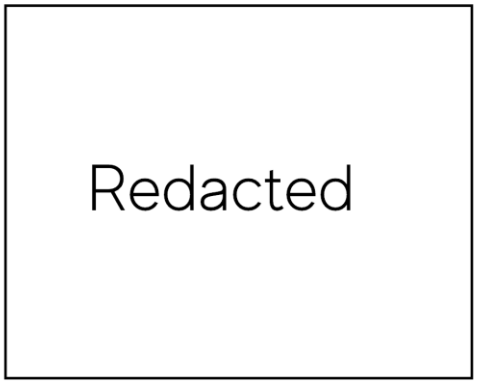
Existing Record Found prompt

If the applicant is NOT considered a high-confidence match with existing voter records, the applicant is automatically treated as a new voter record.

Lower-confidence record matches are linked and post-processed in the List Maintenance service by elections staff this ensures any staff processing scanned queues do not accidentally link lower-confidence matches that may go undetected.

When an existing voter record matches the applicant, the user is alerted that an applicant already has a pre-existing record and SVRS links the record and populates data fields.

SVRS populates the Voter ID, Current Status, Current Reason, and Current signature is visible side-by-side for comparison with the applicant's newest voter signature.



Add Voter header where existing record exists

Current implementations of SVRS include real-time identity verification from any external agency, including DMV (**VR-1.2.1.2**). The VA DMV would need to be configured in SVRS. SVRS is designed to accommodate many API integrations, including real-time integration and validation with the DMV. This integration effort would include the interpretation of the "Voter Registration Opt-Out" (**VR-1.2.1.13**).

SVRS captures e-mail and "notice email" on the Voter Record for voters who wish to forgo printed documentation/notices. This is currently used in the State of California for electronic "opt in" by voters for their Voter Information Guide and Sample Ballots.

As detailed in VR-1.1: Manage Voter, the voter record can be directly edited. This does not overwrite existing NVRA statistics (**VR-1.2.1.18.1**). All edits to the voter record are captured on the History tab for the voter record (**VR-1.2.1.20**). All elements of the voter record are saved, making it possible to generate the necessary information for any statistical analysis or reporting, including EAVS (**VR-1.2.1.19**).

4.3.1.3 VR-1.3: Qualify Voter Registration

Minimum Required Fields are enforced by SVRS in order to save a voter record as Active (**VR-1.3.1**). These fields at present are Last Name, DOB, DL (**VR-1.2.1.5**) or SSN (**VR-1.2.1.4**), Language, Party, and valid

Residence address. Additional field validations or requirements, such as Middle Name, can be added as needed by Virginia (**VR-1.3.1.2**) (**VR-1.3.1.4**)

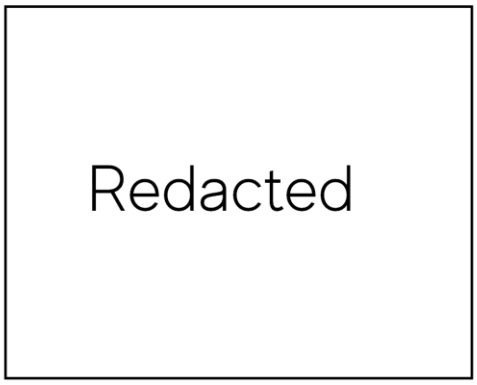
Integration with external agencies, such as Virginia Department of Corrections (VADOC) or the DMV, would have to be configured. Once configured, SVRS validates if the voter is eligible (**VR-1.3.3**), and if eligibility rights were restored when performing voter validations (**VR-1.3.1.1**). This is done currently with the State of California. Currently in SVRS, applicants who are registered felons, are placed in the Canceled status and are not considered voters.

The infrastructure exists to support this validation; however, the system does not currently enforce signature images (**VR-1.3.1.3**).

Minimum age of early applicants is a system configuration item—California has theirs set to 16, and Virginia would define this per their current laws. Early Registrants automatically move to Active status on their 18th birthday (**VR-1.3.2**) assuming all other validations for an Active voter are met. Pending, Underage voters who are 18 on or before an election day, and who are otherwise eligible, are added to an election workspace.

For registrations that are too incomplete to even move the voter into a Pending status to begin issuing correspondence too, the user can simply delete the image—akin to placing the paper form in an actual recycling bin. However, if there is enough information to contact the Voter, SVRS offers a Pending status to allow staff a record to work with in their attempt to rectify the missing data elements (**VR-1.3.4**).

As described above, jurisdictions can enter any form that is appropriate to accept into the county (**VR-1.3.5**). The only forms available in the Form drop-down when adding records are the ones defined in system configuration by the jurisdiction.



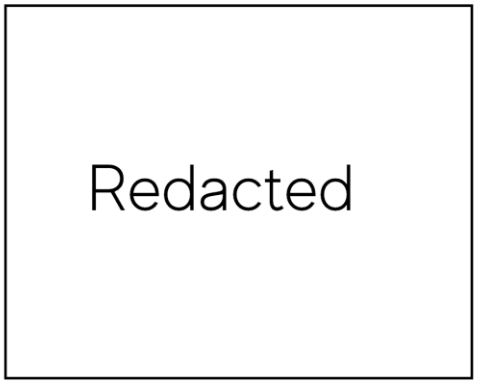
SVRS Bulk Scan Prompt to collect metadata

4.3.1.4 VR-1.4: Address Handling

SVRS accommodates both standard (**VR-1.4.5**) and non-standard types (**VR-1.4.6**) of Virginia residential addresses (**VR-1.4.1**); including flagging any restricted addresses (e.g., business addresses) that an applicant

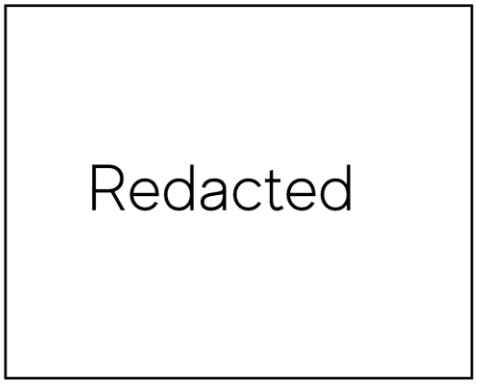
may not register at (**VR-1.4.8.1**) (**VR-1.4.12**) or if the standard address requires a mailing address (**VR-1.4.10**) (**VR-1.4.8**). (**VR-1.4.4**)

SVRS includes predictive type-ahead for standard addresses within the user's jurisdiction. Residence addresses for active voters are constantly associated to a latitude / longitude and/or precinct-portion (**VR-1.4.7**) (**VR-1.4.14**). Non-standard addresses either get a precinct-portion by pin-drop on a map, or by manual user selection (**VR-1.4.14.1**).



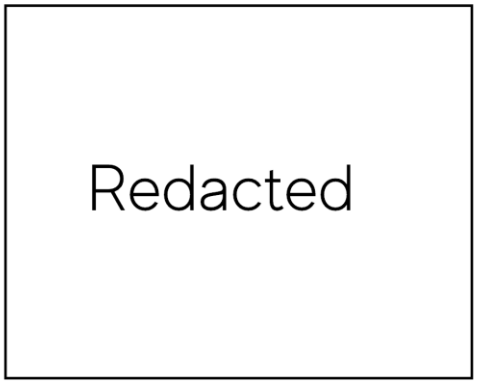
Residential Address – Quick Lookup

SVRS displays the applicant's indication of their residence on a map for the case in which they do not provide a street address.



Residential Address – Selected

After entering in the Record's key data, and residential address, the user would continue to fill out the remaining registration fields, including Mailing Address (Domestic or International) (**VR-1.4.2**) (**VR-1.4.2.1**) (**VR-1.4.3**), Email, Phone Number, Political Party Preference, Prior Registration Information, and declaration of US citizenship.



International Mailing Address for UOCAVA Voter

The distinction for domestic out of state or international address is based on if the state or country is different from the jurisdiction where the voter is registered and is determined by SVRS (**VR-1.4.9**).

SVRS allows for users to flag certain addresses and address ranges, and can be modified to add flags specific to the business needs in the State of Virginia (**VR-1.4.11**) (**VR-1.4.13**).

4.3.1.5 VR-1.5: Close of Registration

There is no need for data entry “black-outs” in SVRS. SVRS is designed to allow voter records to be continuously updated (**VR-1.5.1**) while elections are in flight. Paper or online registrations, updates via the Public Portal or other agencies can remain 24/7/365.

The Close of Registration for an election is defined within an Election workspace. The Election workspace continues to receive updates from the primary Voter record for changes effective on or before the Close of Registration. For example, paper registrations processed after the close would be dated effective prior to the close and therefore would update the election workspace (**VR-1.5.5**). This can be for multiple, overlapping, or dual elections—wherever the voter is eligible (**VR-1.5.2**).

The ability to define UOCAVA voters and election-specific addresses is managed on the voter record directly (**VR-1.5.6**).

The SVRS infrastructure logically separates the voter service from the election service. Only eligible voters exist inside of election workspace(s), and the official list of eligible voters are generated from the election service (**VR-1.5.3**). Not only does this help with enforcing the close of registration rules for an election, but it creates an election time capsule of geographic boundaries and eligible voters. Useful for historical reporting/analysis needs.

SVRS, by default, contains four primary voter statuses: Active, Inactive, Pending, and Canceled. Active voters within the election boundaries receive generated ballots and appear as eligible voters within an election workspace (**VR-1.5.4**). (Underage voters in the Pending status who will be 18 by the election and are otherwise eligible, are also brought into the election workspace).

SVRS provides a robust in-built Notification system that allows users to create or modify Notifications without the need to contact Developers (**VR-1.5.7**). Any automated triggers to generate notices/correspondences for a voter would have to defined in the system.

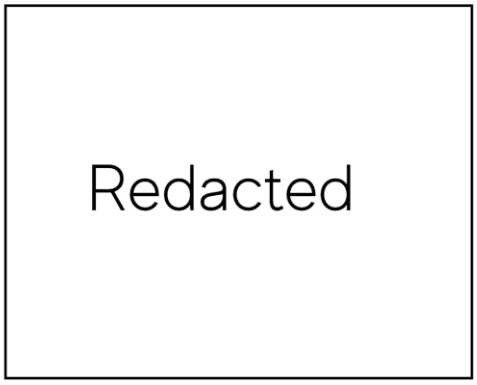
4.3.1.6 VR-1.6: Voter Registration Communications Management

SVRS provides a robust in-built Notification system that allows users to create or modify Notifications without the need to contact Developers (**VR-1.6.1**) (**VR-1.6.1.2**) (**VR-1.6.1.3**) (**VR-1.6.1.3.2**). This includes the ability to define extract data fields in JSON, .CSV, and TAB delimited formats for the purpose of sending to print vendors as well as extracting the fields, including Status Reason (**VR-1.6.1.3.1**) to create and upload custom Mail Merge formats directly into the system.

All custom Notices can be managed by system admins to apply new versions of their formats or extract fields. Only notices that become tied to automated generation workflows, such as the Voter Notification Card / Voter Registration Acknowledgement (**VR-1.6.1.1**), which are protected and cannot be deleted—but can be otherwise managed.

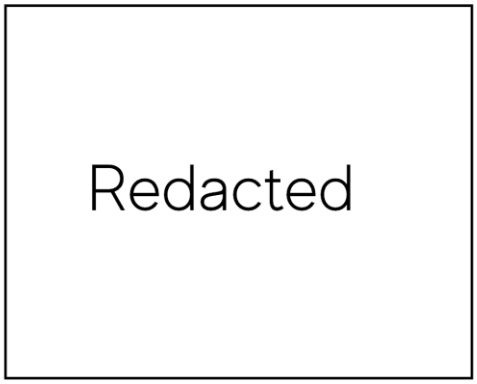
Users can define Notices to include “Business Days to Respond” action if the voter fails to respond (ex: Modify Voter Status to “Inactive” with the Reason “Failure to Respond to Notice”).

When a user processes a scanned record for the voter and the record’s effective date is updated, the countdown trigger is satisfied without having to manually work a queue of pending responses.



System Configuration “Voter Notice” – Add New Notice

All correspondences sent to a voter exist in the voter record on the Notice tab.

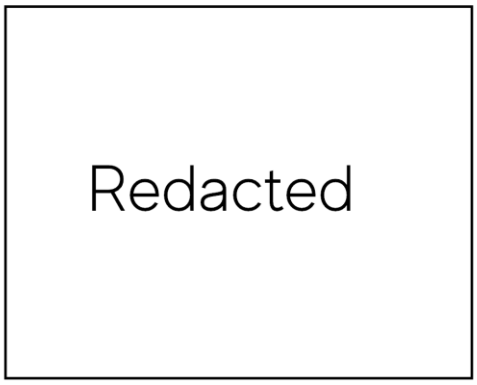


Voter Record – Notice tab, Notice History section

For Virginia, other custom workflows, such as flagging records with non-readable signatures and generating notices would require some level of configuration utilizing pre-existing Notice services native to SVRS.

4.3.1.7 VR-1.7: Correspondence and Document Management & Reports

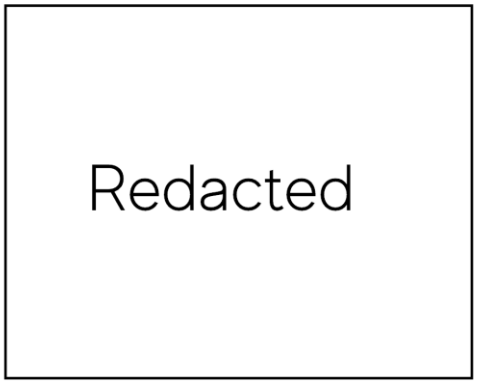
Once a record has been created in SVRS, the user can define additional details that may not be present on a Voter Registration form. The following section details the specific



Voter Registration Details tab

Any additional images (**VR-1.2.1.10**) can be uploaded (**VR-1.7.4**) or scanned (**VR-1.7.2**) to the Images tab on the voter record (**VR-1.7.1**), including proof of identification or citizenship. If the most-current signature is not appropriate for the voter's record, then user can go in and crop a new signature from any associated documentation on the Images tab of the voter's record. This includes electronic signatures imported from the DMV or other sources. (**VR-1.7.6**)

For e-mail attachment uploads, currently SVRS allows PDF and common image types up to 300 DPI. Other extensions would need to be defined.

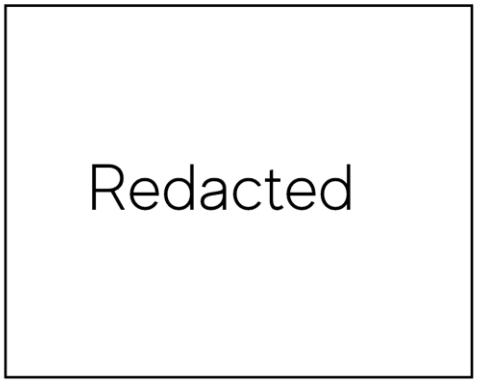


Voter Registration Image tab – Signature Crop button highlighted

A digitized image receipt of a DMV or Online Portal registration can be created and added to the Image tab (**VR-1.7.3**), this is not a present feature, but all of the data is there to do so.

SVRS captures e-mail and "notice email" on the Voter Record for voters who wish to forgo printed documentation/notices. This is currently used in the State of California for electronic "opt in" by voters for their Voter Information Guide and Sample Ballots. This can be expanded to be used for Voter Correspondence by E-mail (**VR-1.7.4.1**) and Text (**VR-1.7.5**). Integration with Virginia's existing Messaging services would have to be configured.

SVRS allows system users to scan single paper registration and bulk paper registrations, or attach saved files individually or add to a bulk queue. When the user is adding a new image, the system collects metadata information such as form type, received date, source, method, date scanned, user scanned, and user processed (**VR-1.7.7**)



SVRS "Add Image" prompt

The users can create any template form to capture signatures or different types of forms to associate to scanned or uploaded documents (**VR-1.7.8**). Voter Registration updates to the voter record via this process are recorded in the records transaction audit log.

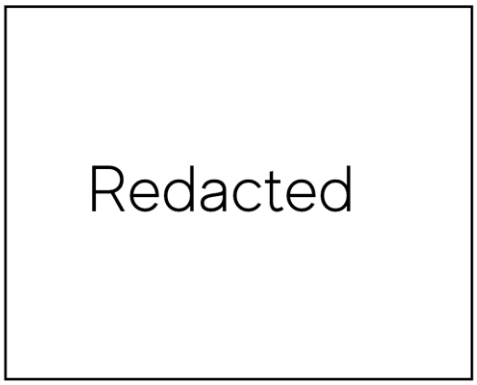
SVRS includes a robust data analytics tool, robust voter search (as described in more detail in Section VR-1.2 “Registration Data Entry and Processing”) (**VR-1.7.9.2**) as well as standardized reports in the Reports module (**VR-1.7.9.1**). These reports pull from the data stored and managed in the voter service. (**VR-1.7.9**).

Currently, SVRS masks Confidential Voter data, including SSN, Address, DOB, or any other PII by default. Only privileged users can unmask this data. Additional masking requirements can be added, such as SSN for all voters—or specific Public Facing reports can be created (**VR-1.7.9.3**).

In California, SVRS includes a process called Public Voter Record Data Request, which is a special report that collects information about the requestors, masks PII, and includes salted data. This could be a worthwhile solution that could be adapted for the State of Virginia.

When mailing a voter correspondence, the system selects the correct address based on address hierarchy, which for elections may include an election-specific address (such as a winter residence in Florida); For reporting purposes, this is the “Mail To” address (**VR-1.7.9.4**) which utilizes this hierarchy.

Similar to the description of “Transfer Voter” (**VR-1.7.10**), SVRS includes the ability to “Push Voter” to another jurisdiction. This appears in this receiving jurisdiction’s List Maintenance module; if rejected (**VR-1.7.10.1**), the voter remains in the original jurisdiction. In its California implementation, the state currently requires “proof” of the move in the form of a scanned or uploaded document, any process deviation from the current implementation can be modified to fit the needs of Virginia.



SVRS Push Voter Requests

4.3.2 Election Administration

The following section describes SVRS functionality related to managing Elections, Offices, Contests, Candidates, Referendums, and Petitions. The labels present in SVRS (e.g., “Races” instead of “Contests” and so forth) are reflective of its development with the State of California jurisdictions, and can be modified to suit the vernacular of elections in the State of Virginia.

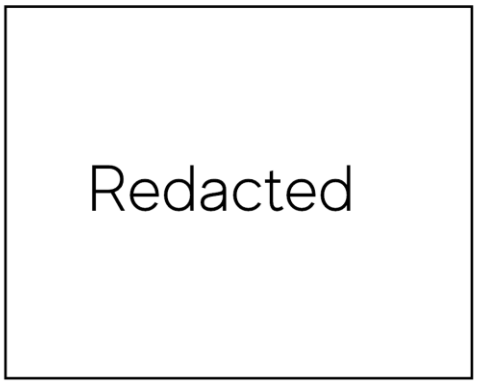
4.3.2.1 EA-2.1: Manage Elections

Unless a user has not been viewing a historic or active election, they will have

SVRS is designed to manage multiple overlapping election workspaces. For example, in California a jurisdiction like Los Angeles can conduct upwards of 50+ city, school and special district elections annually—and that's just one county in the entire state.

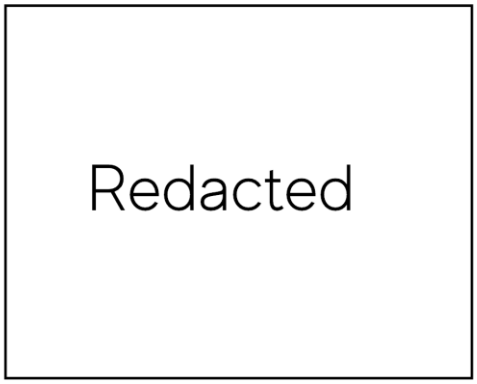
While SVRS prevents duplicate elections of the same Date/Type/Jurisdiction, it allows for overlapping and dual elections (**EA-2.1.4**). For example, a single jurisdiction can conduct a County Special Primary and City Special Primary both on June 8, 2021. Richmond and Fairfax can independently create a June 8, 2021 County Special Primary—and the State of Virginia, with all its jurisdictions, can host a statewide Gubernatorial Primary on June 8, 2021 as well—just to make things more exciting.

SVRS controls this with Election Workspaces. The current, default election for a specific user is displayed constantly on the SVRS perennial header; and includes historical elections. Clicking on the Current Default Election in any module in the system produces a simple prompt that displays key metrics like the current e-day countdown to election day and the user can jump directly into the election record to manage it (**EA-2.1.2**).



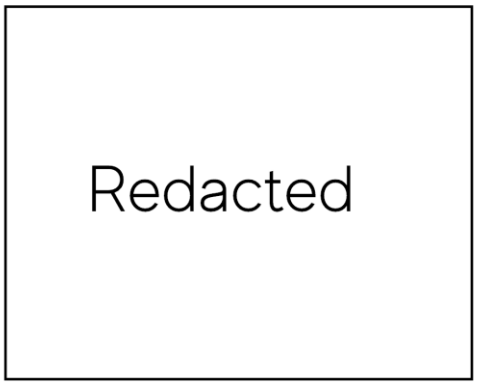
Default Election Details

Everything in elections is calendar-driven by key dates. These are displayed on the Default Election pop-up. The user can then switch the election, or jump to modules such as the election details, ballots, or location management.



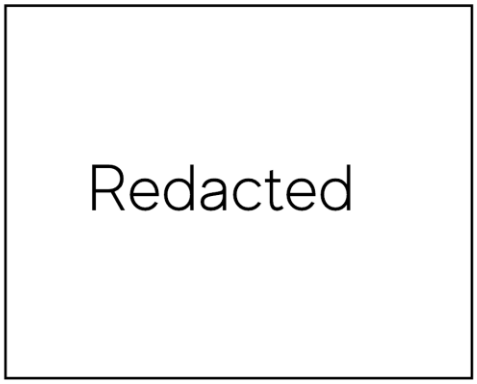
Add Election or Search for Existing Election

Election staff can search (**EA-2.1.1**) for any election in the jurisdiction and apply it as their new election workspace, or create a new election. During election creation, positions that are up for re-election for the specific election type and year are presented to the user to add to the election (**EA-2.1.3**). This process also pulls in current position holders as incumbent candidates.



SVRS Create Election – Add Races Up for Re-election (Contests)

In regards to Closed Primaries, this is actually a configuration item. The State of Virginia would simply toggle the “Open Primaries?” configuration to False (**EA-2.1.5, ERC-7.2.4**).

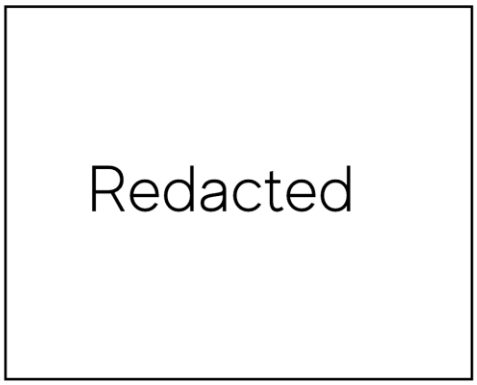


System Configuration: Open Primary = False

4.3.2.2 EA-2.2: Manage Offices

Offices are labeled as “Positions” in SVRS and “Position Holders” are the representatives that currently hold the office. This phrasing be relabeled for use by the State of Virginia.

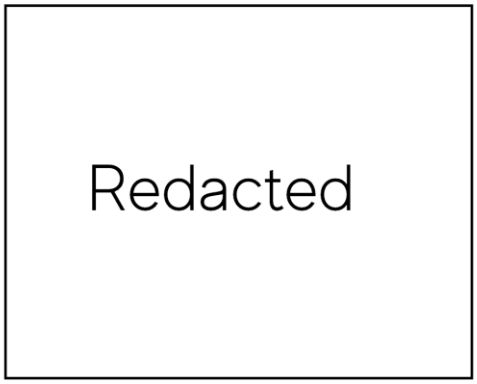
A s with all SVRS modules, Position includes a search and extract option for positions. (**EA-2.2.1**)



Position Search screen

When adding or editing a new Position, the user must define the District Type and District Name (**EA-2.2.2**), Ballot Name, Maximum Position Holders (**EA-2.2.3**), Election Type, Term Years and Start Year (**EA-2.2.4**).

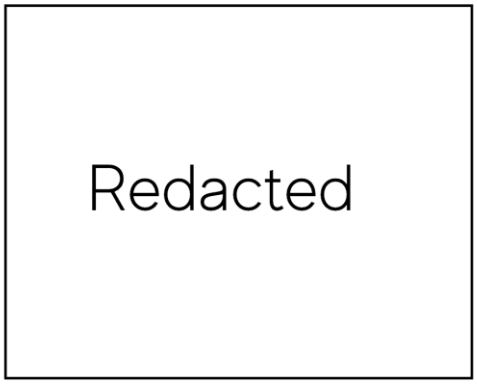
Candidate Eligibility such as specific qualifying elements, such as district residency (**EA-2.2.6**). Filing fees, candidate documents, are managed directly in the Candidate record (**EA-2.2.8**) (**EA-2.2.9**) Additional details such as venue of filing can be added for the State of Virginia (**EA-2.2.5**). Petitions can be linked to Candidate records; this can be added candidate eligibility field as well (**EA-2.2.7**).



Add New Position

4.3.2.3 EA-2.3: Manage Contests

When creating an election record, the election year and the election type displays all Positions that are up for election based on the cycle defined in the position record. This process pulls in all existing position holders as incumbent candidates by default.



Add Race (Contest) during Election record creation

Position Hierarchy is defined within the position details, this dictates the Contest's position on the ballot in SVRS (**EA-2.3.1.2**). SVRS could be customized to add in an election specific Contest Hierarchy if the order differs between elections from standard hierarchy (**EA-2.3.2**). Contests can be added, deleted, and modified (**EA-2.3.1**) until ballots are generated for the voters, including temporary/potential contests or referendums (**EA-2.3.4**). The contest title is based on the Ballot Tile defined in the Position details (**EA-2.3.3**).

The SVRS roadmap includes the ability to do shared districts with owners (**EA-2.3.5**) but it is currently separated at county boundaries. This would allow district owners to share elections and boundaries, but still maintain the voters independently for those living in differing jurisdictions (ex: School districts that cross into multiple jurisdictions) (**EA-2.3.6**)

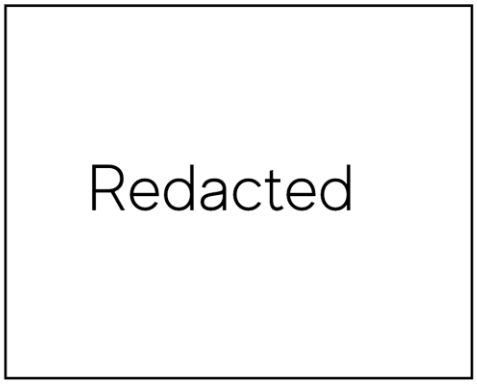
In an upcoming release, SVRS allows users to define the Alphabetical Random Draw (Candidate Ordering by Last Name) for the election, to sort the candidates. For the State of Virginia, other ordering mechanisms for how candidates are ordered on the ballot would have to be created (**EA-2.3.2.1**)

Candidate who are marked as Winners ("top of the ticket") in a primary, are pushed into the associated General election as candidates for Generals defined during Primary election creation (**EA-2.3.1.1**). Defining a running mate would be a modification of the Candidate record's "Name as Appears on Ballot" field. (**EA-2.3.1.1**).

4.3.2.4 EA-2.4: Manage Candidates

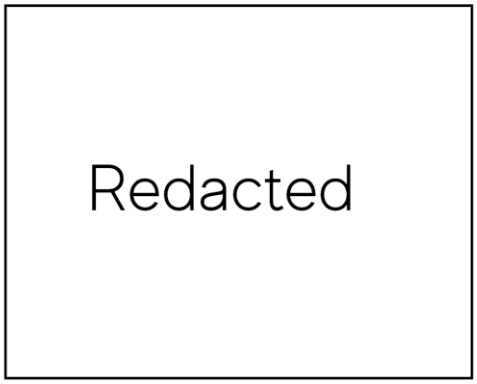
As with most SVRS modules, the Candidate module starts with a Search screen. This allows users to search, filter, and extract candidate data based on the Election Workspace (**EA-2.4.1**). During election creation, imported positions ("offices) become races ("contests") within the election and position holders become incumbent candidate records by default (**EA-2.4.6**) (**EA-2.4.2**); while incumbents are created as candidate records, importing candidates from past election workspaces is not something available within SVRS.

While qualified candidate records can be added and some details edited directly in the Election Details "Measures and Races" tab, ALL candidate statuses can be added and managed from the Candidates module (**EA-2.4.4**) for the election. A flag to locate qualified candidates that are an unopposed (**EA-2.4.7**) can easily be added to allow users to locate these contests.



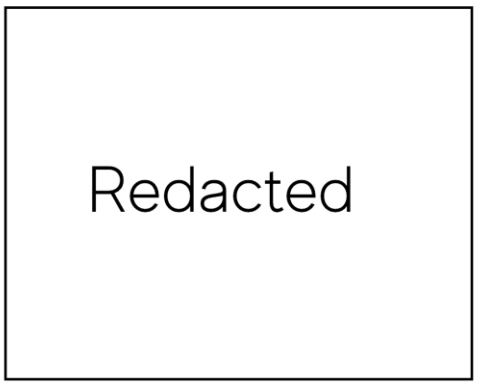
SVRS Candidate Search screen

SVRS does not currently offer a public-facing portal for candidate filing; however, this would be possible to develop as the supporting infrastructure already exists to do so. (**EA-2.4.16**)



SVRS Candidate Record Details tab

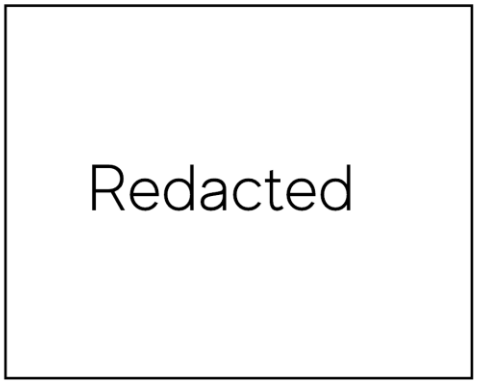
The Candidate Details tab allows users to define First, Middle, Last Name, Date of Birth, Ballot Name (**EA-2.4.14**), Political Party (**EA-2.4.5**), Candidate Status and Reason, whether or not they are a write-in candidate or an Incumbent. Candidate Status options are: Qualified, Non-Qualified, Withdrawn (**EA-2.4.13**), Pending, and Archived. Only candidates in the “Qualified” status appear in the Elections details or during Ballot Style generation



SVRS Candidate Record Qualifications tab

Placing a candidate in the Qualified status is not restricted by the system with automated validations, such as residency requirement (**EA-2.4.11**). The validation itself is manual with the present implementation and exists on the Qualifications tab.

The petitions modules (described below) are in active development, and not currently linked to the Candidate module—however, the design for the petition's module is in line with the description of the State of Virginia requirement for linking to Petitions to Candidates (**EA-2.4.11**) (**EA-2.4.10**)



SVRS Candidate Record Attachment tab

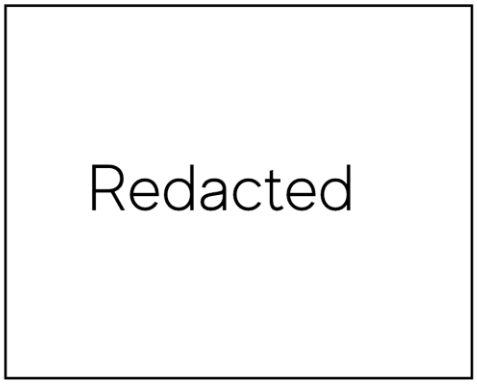
The Attachments tab allows users to upload/scan/view documents associated to the candidate for the election (**EA-2.4.8**) (**EA-2.4.9**). Specific types of documents, requirements, or individual attachment statuses would have to be defined for the State of Virginia (**EA-2.4.12.1**).

In addition to the Data Analytics ad hoc reporting engine and search/extract options, standardized reports can be created for the State of Virginia within the Report module (**EA-2.4.3**) (**EA-2.4.4.1**).

Other automated workflows (**EA-2.4.12**) (**EA-2.4.12.2**) or non-voter related correspondences (**EA-2.4.15**) would have to be added; however, the basic infrastructure exists to support these functionalities and would be prioritized for the State of Virginia.

4.3.2.5 EA-2.5: Manage Referendum

Referendums exist on the Election's "Measures and Races" tab (**EA-2.5.1**). This tab and other references can be re-labeled to "Referendums/Contests" for Virginia. All referendums are required to be associated to a District Type and District (**EA-2.5.3**) for ballot style generation. Adding or modifying Referendum data continues until voter ballots are generated (**EA-2.5.2**).

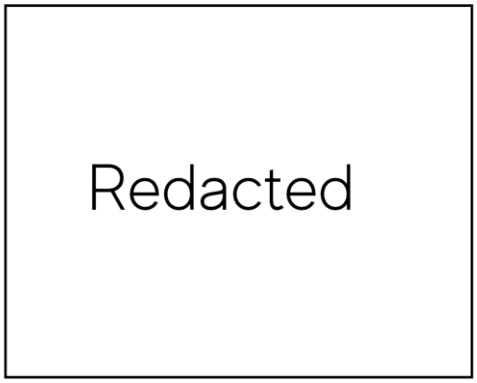


Measure and Races tab

4.3.2.6 EA-2.6: Manage Petitions

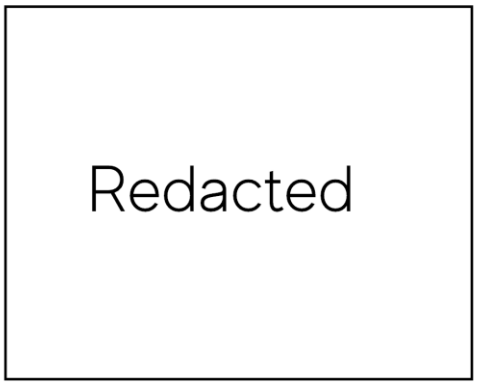
[Development] SVRS Petitions module is in active development. The images presented in this section are system design mocks to help give an idea of the SVRS Petitions module.

Similar to the SVRS Notices, Ballot Adjudication, and List Maintenance sections, the Petitions screen starts with cards of Active petitions. The user can open an active petition (**EA-2.6.2**), create a new petition, or search (**EA-2.6.1)** for all petitions in the system; active, failed to qualify, certified, or archived (**EA-2.6.2.1**).



Design Mock: Active Petition Card

Master Petitions for a multi-jurisdictional petition (ex: statewide petition to recall the governor) where each county processes the sections for the voters in their jurisdiction for the larger overall effort. This would be a later version of the Petitions module, similar to shared districts in Elections. (**EA-2.6.2.2**) Currently the system only allows jurisdictions to view voters under their purview (**EA-2.6.2.10**), this can be included for the state of Virginia.

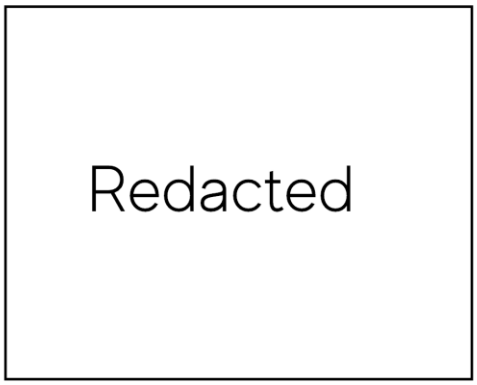


Petition Module Mockup – Add Circulator Button

Once the petition is created, the users add the Circulators to the petition. This list of circulators (**EA-2.6.4**) is used during pre-processing sheets to associate a circulator to a sheet so the user can validate the circulator.

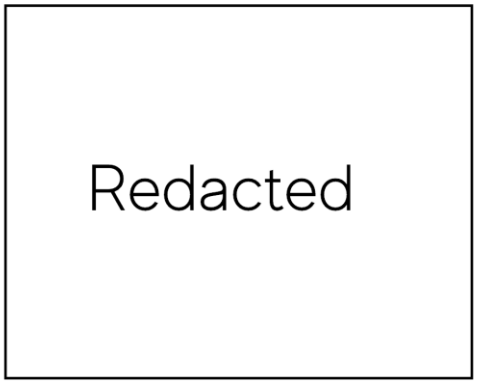
During Pre-Process Sheets, the user would scan in the petition pages and rows (**EA-2.6.2.3**) rows and capture signatures (**EA-2.6.2.8**) for side-by-side comparison with the voter signature, and validate Circulator (**EA-2.6.2.5**) on entire sheets. The end of this process results in the count valid pages and raw count of signatures (**EA-2.6.2.11**) returned for the petition.

After searching and connecting to a voter record, the row can be validated manually (**EA-2.6.2.4**); however, because the row is now linked to the voter record (example in Row 2) by performing a search for the voter within the jurisdiction. The system then performs some basic validations on behalf of the user such as underage, incorrect district, invalid status, registration date, etc.



Design Mock: Manual Row Validation

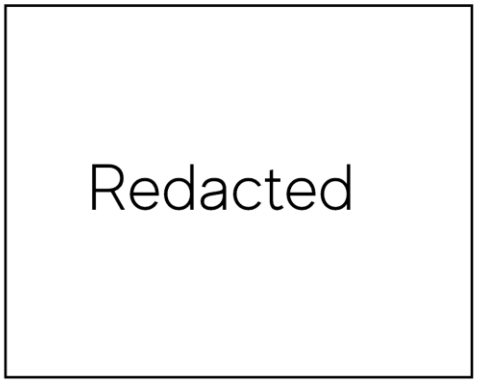
As shown in the mockup above, the user searches and selects a voter record to link to the petition row. This brings in voter record details and allows the system to perform a validation (**EA-2.6.2.6**). As the users are validating and the number of valid signatures hits the minimum threshold (**EA-2.6.2.13**), the users receive a prompt that minimum valid signature threshold was met. Additionally, the signatures remaining column decreases to zero based on the minimum threshold count (**EA-2.6.2.9**) defined is System Configuration.



Petition Mockup – Reject Reason Prompt

Rows are either Accepted, Rejected, Blank, or Unprocessed. Rejected. In addition to manually determining if a signature is rejected and selecting the reason (**EA-2.6.2.7**), the system performs basic validations on the

connected voter record automatically. These validations include if the voter resides in the petition district, underage voters, and non-active voter statuses—and the user is prompted when a row encounters one of these validations (**EA-2.6.5**).



Petition Record Details Mockup – header

Custom Voter Notifications can be created by users in System Configurations, including Petition Signee Notification. Automated Workflows to trigger these notices to generate when certain scenarios happen would need to be defined for the State of Virginia (**EA-2.6.6**).

In addition to the Data Analytics ad hoc reporting engine, standardized reports can be created for the State of Virginia within the Report module (**EA-2.6.2.12**) (**EA-2.6.3**).

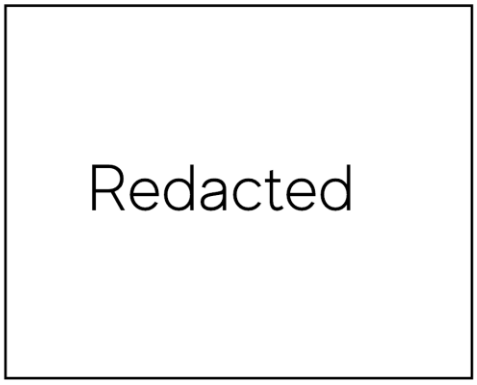
While still in active development at the time of responding to this RFP, version 1 of the petition module is planned to be ready in time for the anticipated demo timeframe.

4.3.3 Establish Voting Locations

4.3.3.1 EVL-3.1: Maintain Voting Location

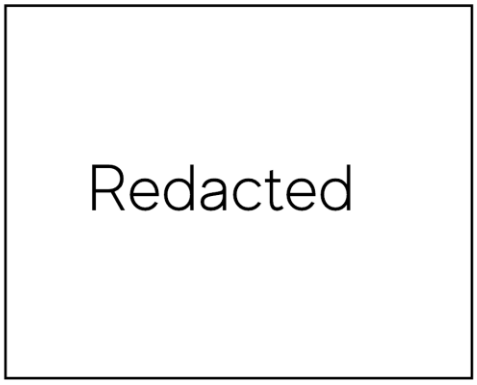
SVRS is designed with the intent of an election being managed as an event. To this end, Locations are maintained as a separate record and added to one or more elections as needed. This allows a Location record to be Archived, even if it's currently in use by an election (**EVL-3.2.5**).

To add or manage locations/drop-boxes, election staff use the Vote Location service in SVRS.



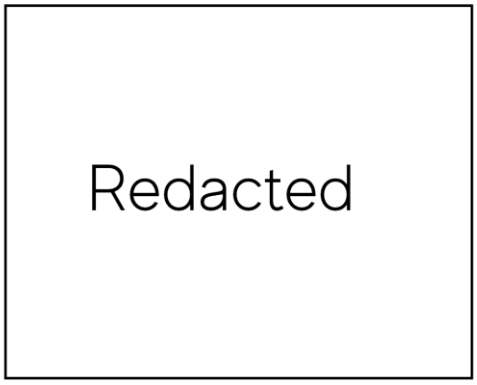
Vote Location section in SVRS

As with most SVRS screens, the user is presented with a search screen (**EVL-3.1.1**). On this screen users can search for Locations by a variety of metrics, as well as Add New Locations (**EVL-3.1.5**) (**EVL-3.2.6**).



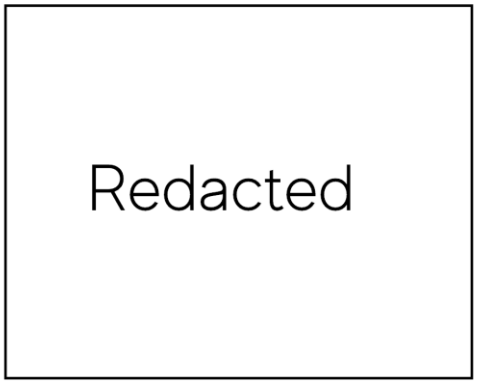
Locations Search Screen with Results

As with all other search screens in the tool, privileged users can extract the search results into a CSV format. This functions as a type of ad hoc reporting for the users.



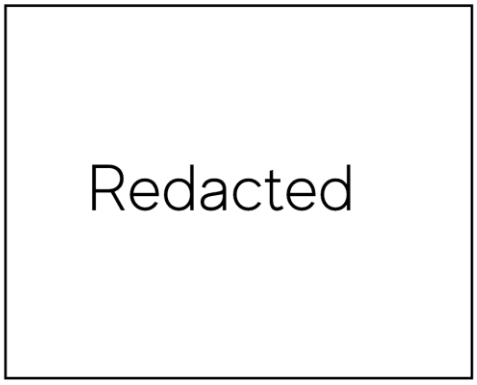
The Extract/Download button in Location search

On the Location detail's view (**EVL-3.1.2**), a user can define and update many elements of the Location (**EVL-3.1.3**). The location's Address is manually entered, but can be tied to GIS similar to Voter if desired for additional standardizations (**EVL-3.1.4**). Currently, the system doesn't restrict duplicate locations at the same building (**EVL-3.1.8**). The location type refers to the daily usage of the location: Public-School vs Private-Owned. In some states, Public Schools require advance notification to include an Election on their school calendar. SVRS allows users to define all Public-School types to be located separately to ensure this advance notice.



Election Location Type

A location's primary Purpose (use) can be defined such as a Ballot Drop box (**EVL-3.1.6**), Central Absentee Precinct (CAP), or Polling Place. The State of Virginia can define different location purposes as a System Configuration item.



Election Location Purpose

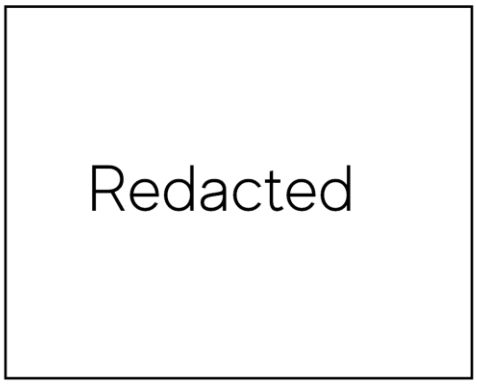
When a Location is added to an Election Event, it may offer different Services during the election event or during different election timeframes (ex: Early Voting vs Election Day).

Reversely, some Locations may be primarily used as a Ballot Drop box during Local elections, but during a Federal General this location may offer full suite of voting services and replacement ballots.

Services vary by event and by the timeframe in the event. For instance, on Election Day a location may be a full-service voting site, but during the run-up to the election, it may only be a ballot drop-box location.

For the State of Virginia, the definition of services for Full-Service, Partial Service, or a new service designation is a System Configuration item native to SVRS and can be defined at the county or state level.

Below is an image of the Location with Event Services defined:



Location Service Level tab showing "24-hour" drop box designation

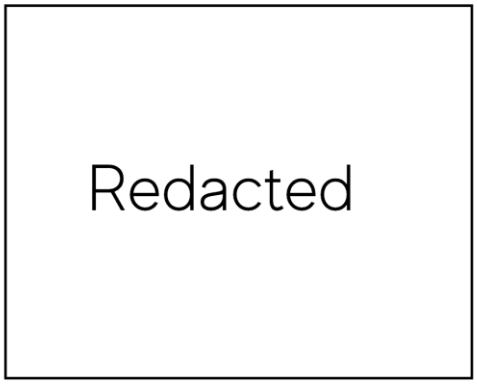
The distinction between Location Type, Purpose, and Services is a key to treating elections as events with locations as a resource for the event that are filled with resources and inventory.

The voting location(s) are displayed to a voter in the Public Portal (**EVL-3.1.6**) (**EVL-3.2.2**) and are described in that section.

Contact, Storage, and Connectivity

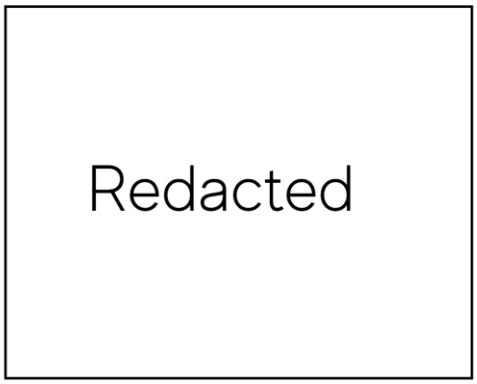
Additionally, SVRS displays (based on the Election Workspace a user is on) the Services, Workers, and Inventory for a specific location. For the State of Virginia, additional metrics such as precincts, district, and location voting history or usage can be added.

Locations can be defined to collect contact Information, after-hours contact information, site contact phone(s), ability to receive text messages, e-mail address, and business hours.



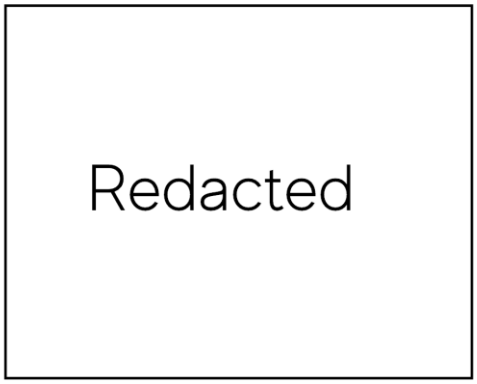
Location Contact & Notice Information Fields including School Year for "Public School" Type

It can be necessary to bring equipment to a location prior to the setup, or to store back-up materials onsite. The location page allows users to define some storage and site security details. For instance, it may be useful to know that a storage area or ballot drop-box location already has Security Cameras.



Equipment Storage information (if Applicable)

Picking a voting location for an election event is the balance between finding the most ADA compliant site, near voters, and allows for voters to get to the location. Each location can be flagged for key public access notes, such as Parking Lot/Garage is nearby, or the site is close to Public Transportation.

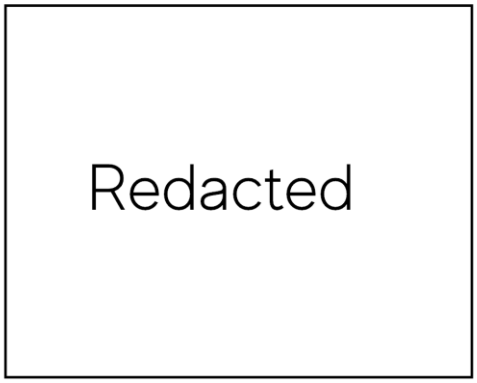


Location Parking and Public Transit Information

Managing Drop Box Locations

While the physical Ballot Drop Box is an Inventory items (e.g., they are stored in a warehouse), during an election event they exist also as a physical location which may have hours of operation (such as a drop-box inside of a building). As a result, they are managed as Locations.

When defining locations, a location can operate both as a drop-box site, a voting location, or both.

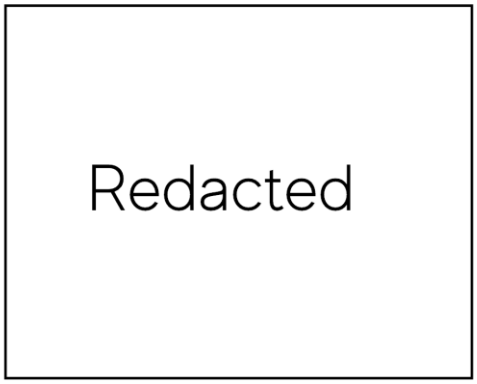


Create New Location screen

A Location record includes name, address, standard fee, status.

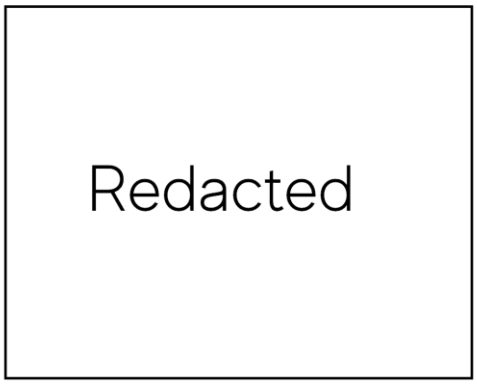
American with Disabilities Act (ADA)

Anyone who has performed ADA Compliance on a location knows that no one site perfectly meets every single item on the checklist and inventory is needed—ramps, signage to designated ADA accessible path, signage to create an ADA accessible parking spot, door stops to prop open doors, among others.



ADA Compliance Checklist

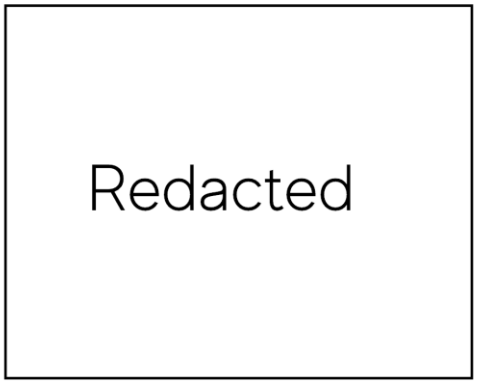
In SVRS, each locations site survey (**EVL-3.1.7**) is directly associated to the location, and can be Modified and printed from a fillable PDF that opens on another browser tab by section of the compliance report to make it easier to location modifications.



ADA Compliance Survey by Section: PDF Fillable Checklist by Location

Attachments

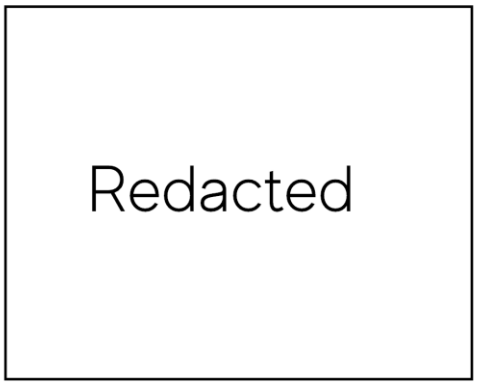
Pictures, contracts, and more can be uploaded for each location.



Image/Docs tab in Location Details

As with all SVRS Services, privileges users can export location data directly from the search screen; addition of the X/Y coordinates can be included as result columns.

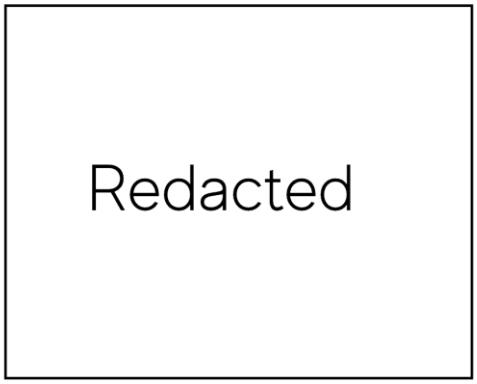
The reserved Inventory tab (**EVL-3.1.9**) allows users to add and print Inventory Lists for each Location in SVRS. However, this is not currently connected to the Inventory module, which is in development.



The Reserved Inventory tab

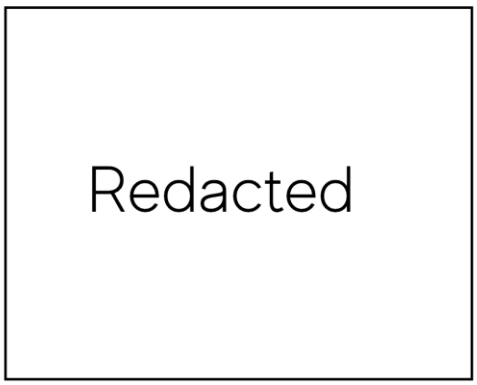
4.3.3.2 EVL-3.2: Assign Voting Location

Location Assignment is done in the Event section. Assign Locations to an Election individually (**EVL-3.2.1.1**) or use an archived election as a template to get started (**EVL-3.2.1.2**) (**EVL-3.2.1.3**). GIS integration is on the roadmap for location assignment. Automated Initial Assignment based on Home Precinct or Townships involved in the election can be built out (**EVL-3.2.1**).



Assign Vote Locations prompt

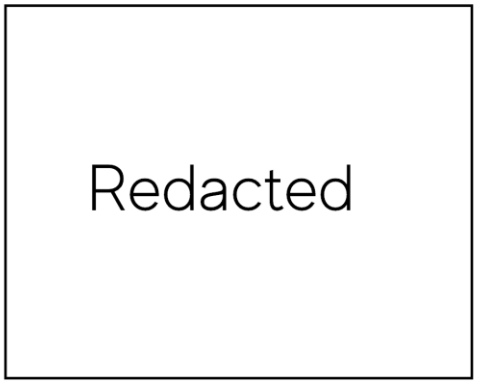
Locations can then be filtered, accessed, and edited directly from the election event. Once a Location is added to an Event it is a part of the Election Workspace. It exists separately from the Location record itself, similar to a Voter or Geographical area in the Election Workspace. As a result, the Location record can be Archived while an election is in flight. The system prompts if the user wants to Keep or Remove the location from its (**EVL-3.2.5**)



Event Details, Assigned Locations view

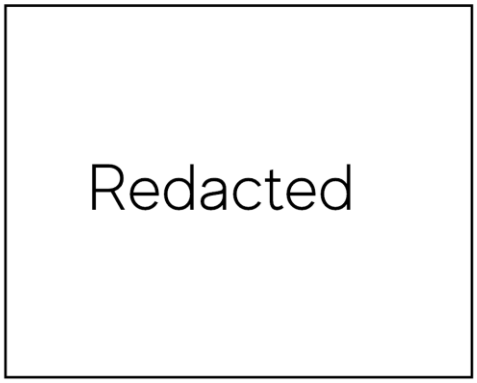
Additional workflows and validations can be incorporated for the State of Virginia. For instance, locations assigned to an election event that are outside of a precinct boundary of an election GIS area and need to be associated for voter access (**EVL-3.2.3**).

SVRS allows for the custom creation of Voter Notices as a default in the system (**EVL-3.2.4**), which would include Voting Location Assignment notices. However, automated workflows associated to Notice generation would need to be added as triggering scenarios vary.



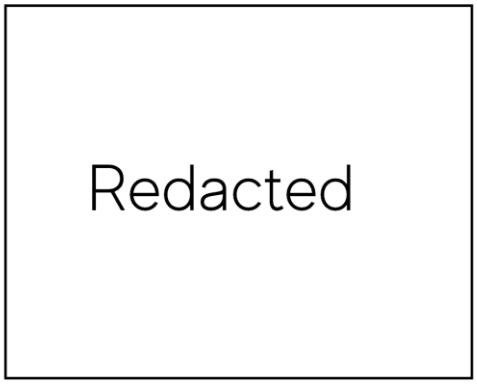
System Configuration: Notices section

4.3.3.3 EVL-3.3: Voting Location Correspondence and Reports



SVRS Standardized Reports module

As with all other search screens in SVRS, privileged users can extract the search results into a CSV format (**EVL-3.3.2**) the search functions as a type of ad hoc reporting for the users. In addition to the Data Analytics ad hoc reporting engine, standardized reports can be created for the State of Virginia within the Report module.



Locations Search Screen with Results

The existing Election Notices section can be utilized to generate location election use notices (**EVL-3.3.1**), as with all notices, automatic workflow triggers would need to be customized for the State of Virginia.

As described in more detailed in the sections detailing the Public Portal, the voter's nearest voting locations and drop boxes are displayed in an interactive map they can use to get directions to the selected site (**EVL-3.3.3**).

4.3.4 Poll Worker

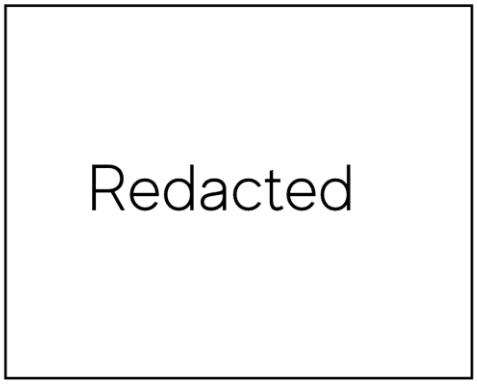
4.3.4.1 PW-4.1: Manage Poll Workers

Similar to Vote Locations, Workers represent a pool of individuals that can be included in an election event; The workers included in the Election Event may not represent the entire pool of potential workers, this is especially true for smaller elections.

SVRS maintains the Election Workers records (**PW-4.1.2**) in a separate module, and pulls them into a specific Election Event. As a result, a worker could be assigned, trained, and paid for work in multiple or overlapping elections without conflict.

The SVRS architecture is API-based microservices—this extends within SVRS to each key module ("service"), including Worker. This allows SVRS to integrate with existing Poll Worker Management technology or websites (**PW-4.1.2**), (**PW-4.1.3**), (**PW-4.1.4**), (**PW-4.1.5**). An integration with existing poll worker management technology in SVRS can be done to either extend SVRS functions or replace them entirely with existing/preferred tools. SVRS provides the Worker service as included with the full suite of tools.

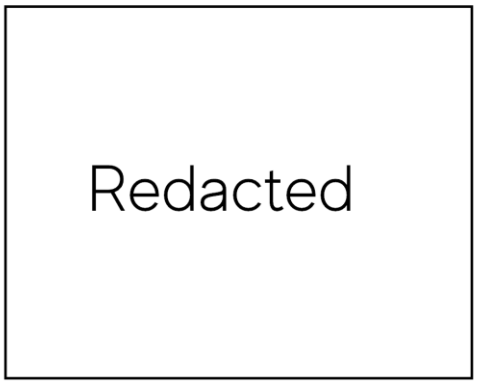
The below image shows the service icon for the Election Workers service section on the SVRS User Interface (UI).



Election Worker service icon

Worker records are created or updated manually or as a result of voter registration **(PW-4.1.5)**.

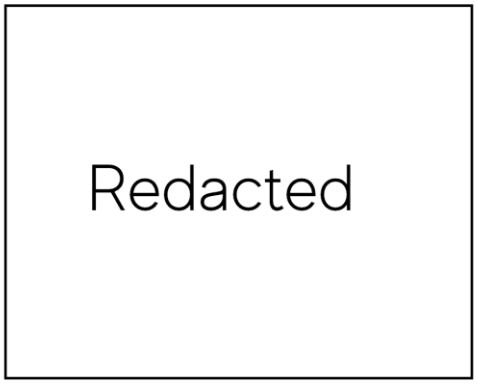
The image below shows the “Worker Update/Create” process via the paper Voter Registration process. SVRS also includes the ability for a worker to designate their language fluency during the registration process. For the State of Virginia, the Public Portal is another available source of voter/worker registration and updates.



Worker Registration/Update during Voter Registration

SVRS tracks many metrics on the records added to the Worker service (**PW-4.1.5**), including collecting information such as Effective Date (Date of Interest), Source/Origin (Public Portal, Voter Registration, In-Office, etc.) (**PW-4.1.5**) as well as Event Specific details such as Election Worked, Hours Worked, Training Assignments and Statuses (**PW-4.1.3**), and Locations Assigned (**PW-4.1.2**).

The Worker service begins with a search screen to search a variety of criteria—any data field in worker can be added as a Searchable field; additionally, the Search allows privileged users to extract of all of the result information into .CSV format.



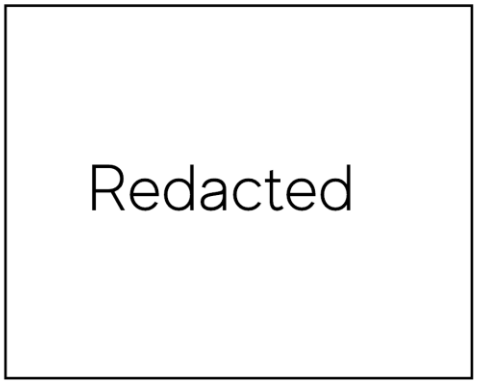
Election Workers Search screen

SVRS's ad-hoc reporting allows for even more detailed questions.

Worker Details

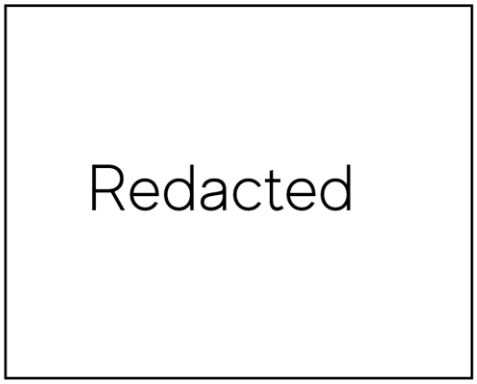
The ability to automatically link to pre-existing workers currently exists via the Voter Registration source only; however, this functionality for manually created records to automatically determine duplicates is on the SVRS roadmap, and can be prioritized for the State of Virginia. Business processes, such as searching for existing Workers prior to creating them can support this as well.

The below image shows the Workers assigned to a location for an Election Event on a Location's Workers tab.



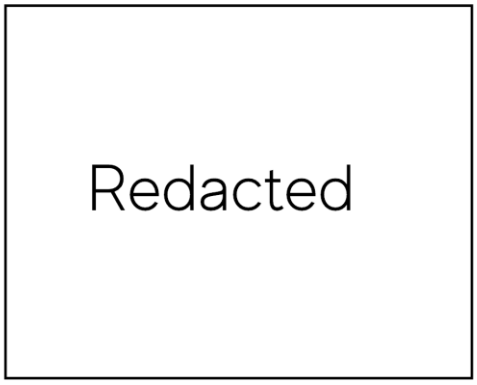
Workers view on Location based on Election Workspace

On the worker details screen, users can view worker name, contact information, and if the worker is underage (e.g. a Student).



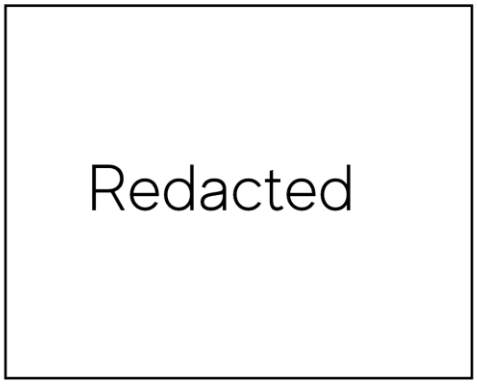
Election Worker Search with Results

And Worker status, this includes their home precinct and residential city; additional distinctions can be added. For the State of Virginia, Workers can have any additional metric, including political party preference (via the voter record) included.



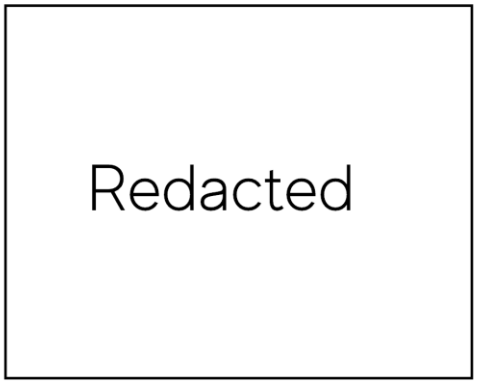
Example of Worker Details tab

The jobs/skills tab captures many additional details about the worker. The Worker Type defines if the worker is a Registered Voter, a Permanent Resident, or a Student. Any additional metrics can be captured depending on State preferences or code to ensure that a specific voting location doesn't have too many students, non-registrants, etc.



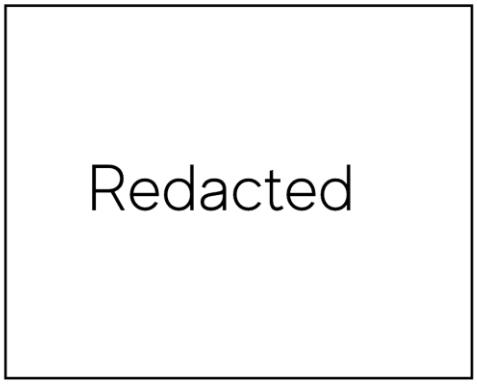
Worker Type drop-down

The primary role of the Worker does not restrict role assignment, and is a captured metric. This is customizable based on State-specific needs. Examples of other common Roles are Inspectors, Clerks, Check-in Clerk, Rovers, Students, Regional Support, etc.



Work Role drop-down

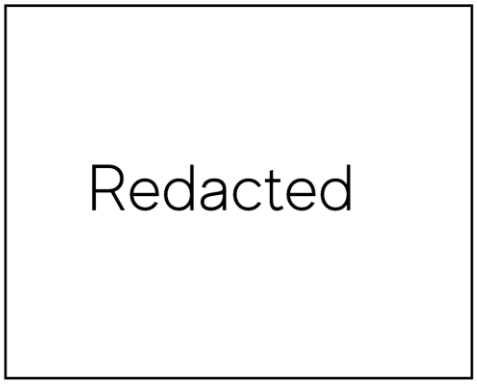
If the worker is a County Employee, the Employee number can be inputted. If the worker is a registered voter, or the record was created via a voter registration process, the Voter ID is associated to the record, if the worker is willing to travel (outside of their home precinct), spoken languages, and any number of flags deemed applicable to the State of Virginia (ex: “Can Lift 20lbs”).



Worker Jobs & Skills tab

Event Assignment

The Event Assignment tab shows all current event assignments for a specific worker—including historical event history. An individual worker can be assigned multiple roles in a single event, and is paid according to their role. A similar view is available in the election Event and Location to view all Workers assigned to a specific election event or location (**PW-4.1.2**).



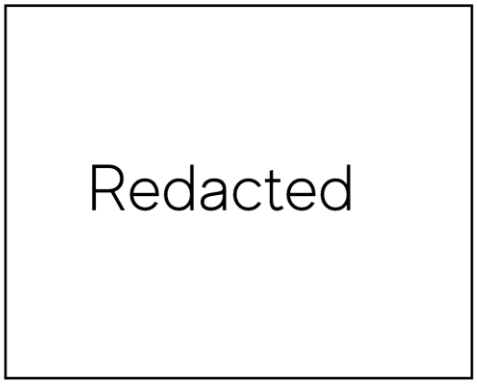
Election Worker – Event Assignment tab

Worker Correspondence

Worker Payroll

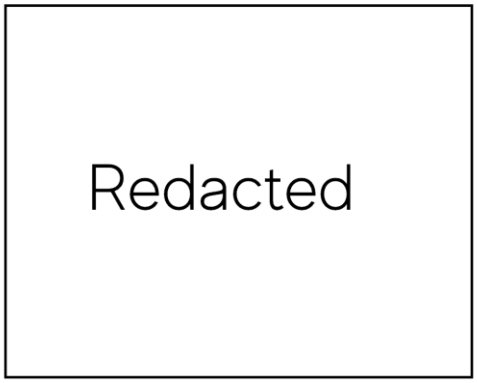
The system configuration for Workers is currently behind the scenes, and being implemented in the front end. This would be prioritized for the State of Virginia so users can designate rates of pay by role (**PW-4.1.2**) and/or by certification mileage costs, and other customizable components such as Roles or Types (**PW-4.1.2**).

On the Payroll tab, additional context can be added in the form of unavailable timeframes for the voter—which restricts the assignment of training and work during those times.



Individual Worker Payroll/Hours Screen

When adding Workers to a Location for an Election Event—SVRS does this in bulk via the Event service. This is sorted based predominantly off of the Worker's Home Precinct or physical closeness to a Location (GIS).



SVRS Bulk Task Icons on Search Page

SVRS performs Worker Bulk tasks for Notifications (**PW-4.1.1**), Training (**PW-4.1.3**), Assignment, and Payroll (**PW-4.1.3**). Payroll can be customized for extract to existing Accounting departments or for in-office printing depending on the need.

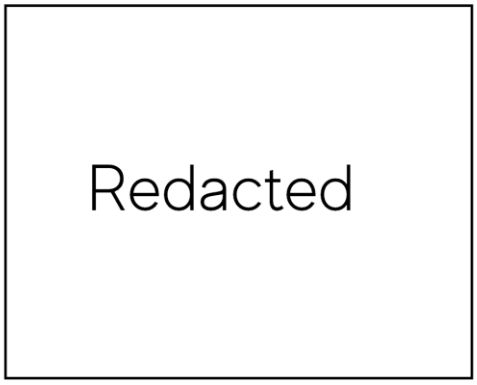
4.3.5 Election Preparation

The largest event hosted inside of the United States of America is an election. Coachella attracts the same number of attendees as there are locations at a major US election. All of those locations, staffed with workers, volunteers, machines, ballots, vendors and more come together to ensure that on election day (and before) the jurisdiction is ready to begin ensuring what seems to the voter like the simple act, engaging in democracy, is successful once again.

4.3.5.1 EP-5.1: Ballot Preparation Management & Proofing

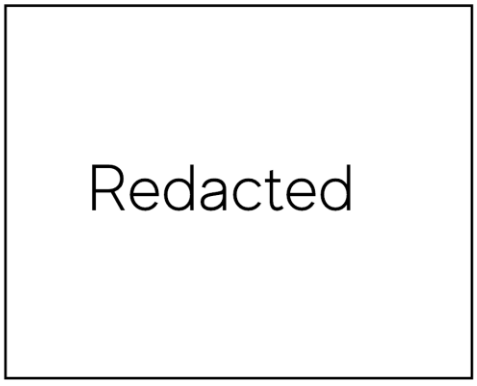
SVRS is designed to manage multiple overlapping election workspaces. For example, in California a jurisdiction like Los Angeles can conduct upwards of 50+ city, school and special district elections annually—

and that's just one county in the entire state. While SVRS prevents duplicate elections of the same Date/Type/Jurisdiction, it allows for overlapping and dual elections (**EP-5.1.11**). For example, a single jurisdiction can conduct a County Special Primary and City Special Primary both on June 8, 2021. Richmond and Fairfax can independently create a June 8, 2021 County Special Primary—and the State of Virginia, with all its jurisdictions, can host a statewide Gubernatorial Primary on June 8, 2021 as well—just to make things more exciting.



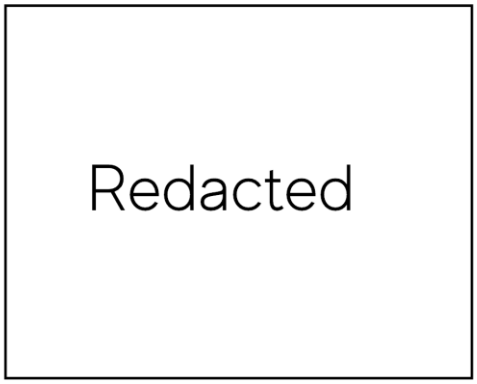
Current Default Election

SVRS controls this with Election Workspaces. The current, default election for a specific user is displayed constantly on the SVRS perennial header; and includes historical elections. Clicking on the Current Default Election in any module in the system produces a simple prompt that displays key metrics like the current e-day countdown to election day.



Default Election Details

Everything in elections is calendar-driven by key dates. These are displayed on the Default Election pop-up. The user can then switch the election, or jump to modules such as the election details, ballots, or location management.

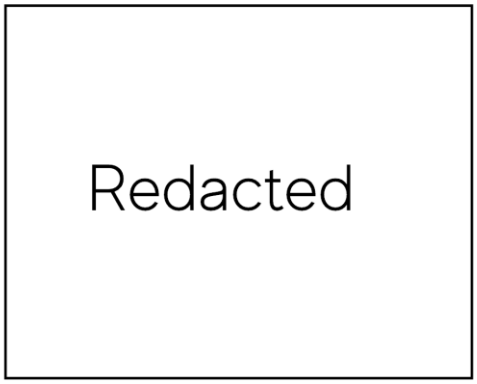


Add Election or Search for Existing Election

Election staff can search for any election in the jurisdiction and apply it as their new election workspace, or create a new election.

After the election is saved, and the election staff complete adding in Districts to the election (via Measures or via Races). Home precincts and involved districts are automatically associated based on the details of the Measures or Races. SVRS automatically determines the unique ballot styles based on selected districts in Measures and Races, this can be modified and re-run up until the Voter Workspace is generated to Issue Ballots (**EP-5.1.5**) and then the ballot styles become protected.

Ballot Styles are automatically generated (**EP-5.1.1**) from the unique overlap of election district boundaries, political parties (if applicable) (**EP-5.1.12**), and languages (if applicable) within the election (**EP-5.1.2**). On the Ballot Styles tab, users can validate each style (**EP-5.1.4**), modify the election details and re-generate as needed for corrections. Users can extract (**EP-5.1.6**) the ballot style in a format used by their tabulators (**EP-5.1.6.1**) and/or to send to the print vendor (the specific format requires configuration).



Ballot Styles tab in Election Workspace

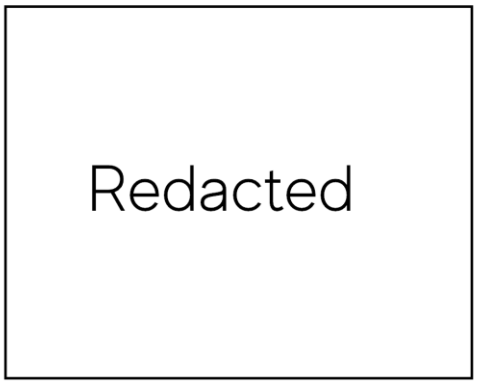
The Ballot Styles tab includes an option to print a Ballot Styles report of all election styles for validation (**EP-5.1.3**), as well as an upload process (…) for the PDF copy of the Sample Ballot per style (**EP-5.1.7**). Ballot Styles are associated to Voters; depending on the needs of the State of Virginia, the uploaded Sample Ballot

or a Report can be published to the Public Portal for a voter to view (**EP-5.1.9**); while this feature is not currently on the Public Portal, the infrastructure to support this functionality is—and can be prioritized for the State of Virginia. The workflow to "print" a ballot style for Ballot on Demand Printers would need to be defined for the State of Virginia, but is available as an option in SVRS (**EP-5.1.6.2**).

As shown in the screenshot above, SVRS currently defines "Total Active Count" and a "Buffer" count, for a ballot style. Additionally, it produces a Grand Total value (Active + Buffer, All Styles) at the top of the Ballot Styles screens to help estimate number of ballots to print per style and total for the county (**EP-5.1.8**). The default buffer is either "5% of Active Voter Count" OR "Total Inactive Voter Count" for the ballot style, whichever is larger. The buffer percentage is configurable.

The System Configuration Election Calendar allows the ability to define key e-dates to auto-populate an election based on the election date (**EP-5.1.10**). Example: Mail Ballot date is forty-five days prior to an election, the official would define "e-45". However, this template can be modified directly in the election record in event there is a unique date for a specific election that deviates from the normal calendar (**EP-5.1.10.1**) ex: a global occurrence that impacts the election cycle.

SVRS allows elections staff to define election dates, election code and name, type, primary voting method (ex: Mail).



Create Election screen.

The election staff define details of an election such as Code and Name, Election type; SVRS allows election staff to additionally define supported Official State Languages by the jurisdiction for the election. For the State of Virginia, Challenge Period definition would be added to support election certification processes.

SVRS System Configuration allows election officials to define Official State Parties (parties that generate a unique ballot for their registered voters); For the State of Virginia, primaries in SVRS generate ballots based on any party that is defined as Official. Standard Races that are scheduled to run for an election appear during election setup and will bring in the incumbent into the election workspace.

In regards to Federal Ballot Styles: while SVRS can be modified to create a Federal-Only ballot style during ballot generation for elections where the Federal district ("United States") is included, for presidential

candidates (**EP-5.1.14**). A Federal Write-In Absentee Ballot (FWAB) is sent by a UOCAVA voter into the jurisdiction more than the other way around.

If the “Federal” Ballot style is referring to federal representatives elected under the US Senate district and/or US House of Representatives district(s) (**EP-5.1.13**)—this can be incorporated for the State of Virginia, but it is not by default in SVRS.

Similar to the above-mentioned variations of Ballot Style generation, SVRS infrastructure is setup to accommodate the creation of unique ballots styles such as “Measure Only” styles (**EP-5.1.15**). However, this is not by default in SVRS.

Currently, SVRS Participation History allows users to add participation for a voter record, define the type of Ballot (ex: “FWAB”), and whether or not it was Counted.

As described in more detail in the Absentee section, SVRS does allow for the Ballot mail date to be defined, and it is modifiable (**EP-5.1.10.2**). The change is tracked in the audit log for the ballot record, and includes the user who made the change and the date/time.

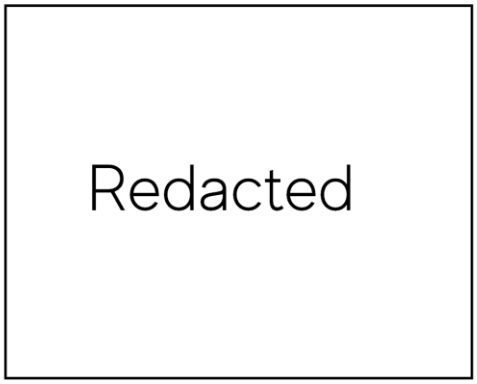
4.3.5.2 EP-5.2: Pollbook

SVRS does support integration with any Rest/API framework or Import/Export processes such as an Electronic Pollbook Applications (EPB) approved by ELECT (**EP-5.2.1**). The frequency of this integration can be defined by the State of Virginia or created as a configuration within SVRS.

The Pollbook generation process in SVRS is customized state-to-state due to differing rules and data setups. Current implementations of this include Extracts (**EP-5.2.1.1**), Supplemental Data, Absentee Status (**EP-5.2.1.2**), and Data Comparison (**EP-5.2.4**). Printing formats would be customized for the State of Virginia (**EP-5.2.5**). Data Import from integrated ePollbooks (**EP-5.2.3**) would have to be customized; however, in the event the SVRS Election Day Portal (described below) is used to collect voter credit—this is immediately consumed within SVRS.

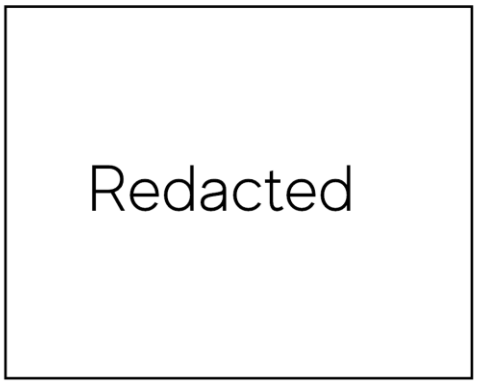
SVRS includes an Election Day Portal (EDP). The EDP is not an ePollbook, it is SVRS module that is streamlined for use by election workers or volunteers at the polls. This included module is simplified and supported by automated workflows. The EDP allows for marking a voter as voted, issuing ballots, capturing digital signatures, and even registering.

All Central Absentee Precinct (CAP) tasks can be performed without exposing the full suite of SVRS tools or data to temp support. If desired, the State of Virginia chose to integrate SVRS with existing Pollbook tools instead (**EP-5.2.2**) similar to other modules in SVRS.



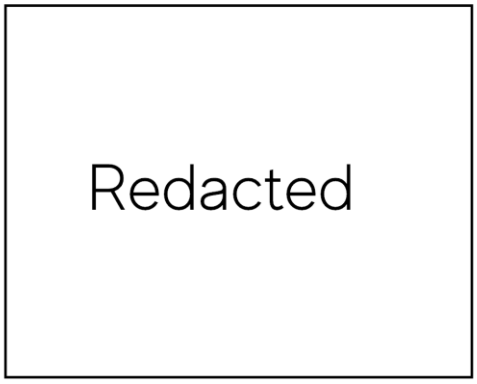
Election Day Portal (EDP) Login Screen

Like SVRS, the EDP is developed for use on a web browser. However, the UI is optimized for use on devices as small as smart phones or tablets. As a result, it offers a range of flexible options to counties to use any existing tools.



All Ballots Issued or Modified are Now Tied to the Location it occurred without Relying on Data Entry

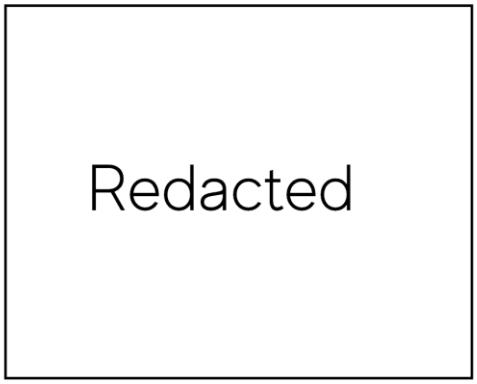
Users are associated to the Election and the Location, ensuring that all ballots issued already have their location of issuance associated, reducing the data entry performed by the end user.



Search Voter

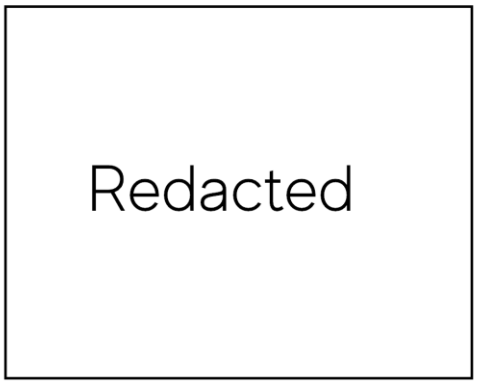
The basic EDP Workflow seeks to locate the election voter, SVRS determines for the worker whether the voter can receive a new Absentee Ballot, if the voter needs a Provisional, or mark if the voter is voting a standard ballot at the polls.

The SVRS EDP could be based for Virginia's specific workflows; For instance, with the State of California, the EDP workflows capture the Provisional application information, including digital signature capture.



EDP Signature Capture block

Since the EDP is just a streamlined SVRS, it follows the same duplicate voter rules and validations as any other registration process; including DMV validations, etc. Because all of the data about the voter is readily accessible, it can



Print Items Screen – Fully Customized

The ability to Print a ballot, ballot envelope label, a voter registration receipt, or more is customizable based on available needs. Currently, this is done with a system change. If printing isn't an option, or it is disconnected/offline (such as a Ballot printer), then this can be modified for the State of Virginia to be a checklist item for the Election Worker instead.

4.3.5.3 EP-5.3: Ranked Choice Voting

While not currently implemented, Ranked Choice Voting is simply a different way of generating ballot styles and consuming the values of votes from the tabulator, this would be easy to implement to the needs of the State of Virginia as the base infrastructure to support this is already in place in SVRS and user configurations to allow the state more autonomy as laws change. (**EP-5.3.1**)

4.3.5.4 EP-5.4: Voting Equipment

As shown in the Vote Locations section, SVRS does have an Inventory module. However, more enhancement will be necessary to include State of Virginia specific details such as “chain of custody” or to integrate with IoT solutions for location tracking. (**EP-5.4.1**)

4.3.5.5 EP-5.5: Voting Information Project (VIP)

SVRS tracks all election, referendum, office, district, street, and poll location information required by the Voting Information Project (VIP) in the overall SVRS modules, but those specific to the election in the Election Workspace for point-in-time generation of data as well. (**EP-5.5.1**)

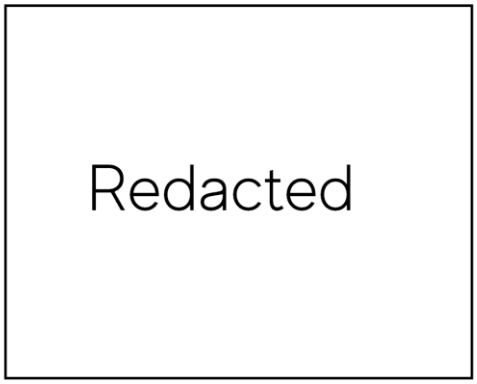
4.3.6 Absentee Voting

4.3.6.1 AV-6.1: EAVS Survey

SVRS Data Analytics tool that is utilized with ****** can be utilized to produce Election Administration and Voting Survey (EAVS) Section A Statistics (**AV-6.1.1**). EAVS Standardized Reports/Extracts are a part of the SVRS Roadmap.

4.3.6.2 AV-6.2: Management of Documents

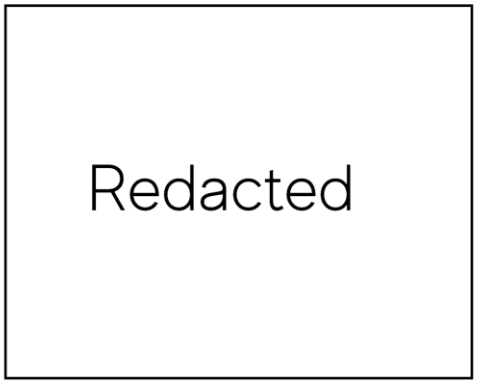
The Vote by Mail (Absentee) ballot details screen for a voter can contain multiple records, and multiple scanned images (**AV-6.2.1.1**) for the returned ballot envelope. Document metadata is currently stored in the background tables, but can be made visible on the User Interface (UI) (**AV-6.2.2**). The voter record itself does not have a limit on the number of attachments or scanned documents (**AV-6.2.3**), currently SVRS supports scanned images up to 300DPI and the common image and pdf formats and are stored as blobs (**AV-6.2.1**). Right now, e-mails would have to be attached as pdf files or image files; however, if more details are provided this can be modified for the State of Virginia (**AV-6.2.1.2**).



Ballot Details screen

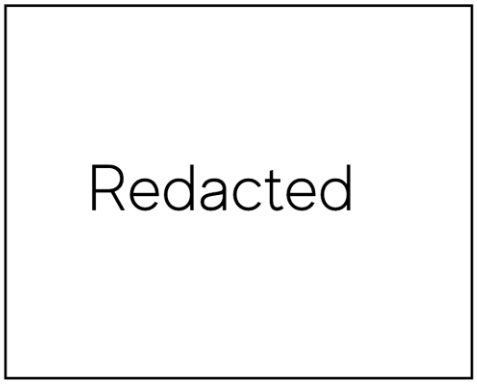
4.3.6.3 AV-6.3: Absentee Reports and Lists

SVRS Data Analytics tool that is utilized with ****** can be utilized to produce any number of ballot metric analysis reports, as well as standardized reports Election section of the SVRS report engine (**AV-6.3.1**). Specific standardized reporting for the State of Virginia can be customized (**AV-6.3.1.3**) (**AV-6.3.1.4**).



SVRS Standardized Report screen – Elections tab

As with all SVRS modules, Ballot Search and Voter Search allow a robust number of options to search voters, as well as extract data for privileged users.



Voter Search – Election History tab

As the absentee applicant process in Virginia includes elements of approval and status, the Voter Search module can be modified to include a tab for Absentee Applicant search criteria and column fields, to easily produce a workable list of Absentee applicants as well as the ability to filter this. (**AV-6.3.1.1**) (**AV-6.3.1.2**) The example above is the SVRS Election History search tab, which shows the current implementation that correspondence to the State of California needs.

4.3.6.4 AV-6.4: Transactional Metadata

Data collection for the absentee ballot application currently includes the method by which the request was received and the method by which ballot will be sent, this is performed on the Election tab of the Voter details record. Additional fields for Virginia-specific absentee applications would need to be added to SVRS (**AV-6.4.1**), but would not be complicated to incorporate.

4.3.6.5 AV-6.5: System of Inquiry

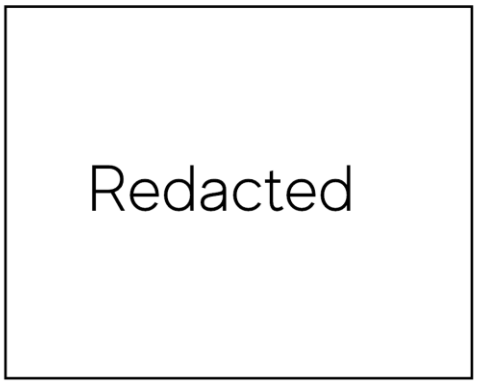
For Ballot Tracking, SVRS interfaces with BallotTrax for the State of California; it is possible to integrate with other similar tools such as Democracy Work's BallotScout. (**AV-6.5.3**). SVRS is an elections platform, and as such each module is even designed as a microservice, this extends to external Rest/API integrations. Unique integrations would need more details to define, such as for Virginia-specific needs such as Ballot Application status email or text delivery services—however, this could also be done on the Public Portal (**AV-6.5.1**) Interfacing for ADA compliance ballot delivery is possible within the SVRS infrastructure (**AV-6.5.2**).

4.3.6.6 AV-6.6: Maintain Remote Absentee Voter Applications List

SVRS supports both Polling Place (Absentee) and Vote Center (Mail Ballot) election models. In the current implementation, the design includes the ability to define UOCAVA voters and election-specific addresses. In addition to this, California allows a "Vote by Mail" flag. This flag is used to indicate whether or not the voter receives an absentee during Polling Place elections. For Virginia, this would need to be expanded to include statuses, sources, and multiple applications over the course of time. (**AV-6.6.1**) (**AV-6.6.1.1**) (**AV-6.6.1.2**) (**AV-6.6.2**) (**AV-6.6.2.1**)

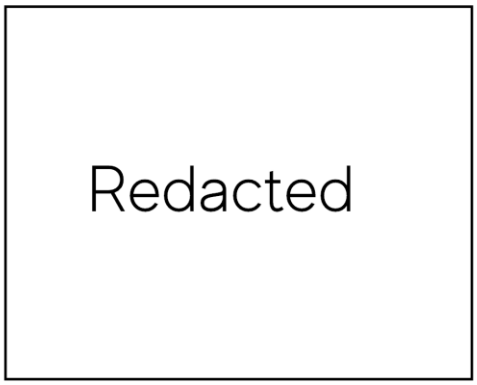
4.3.6.7 AV-6.7: Operations and Maintenance of Absentee Records

As with most SVRS modules, the Ballot screen starts with a search screen (**AV-6.7.1**) which allows users to access ballot records and produce a working list of ballots and voters who have ballots for the election workspace. Search results can be extracted for privileged users.



SVRS Ballot Search Screen – Absentee

SVRS's Perennial Header includes a Ballot ID quick search which allows users to barcode scan (**AV-6.7.1.1**) or hand-type a Ballot ID for their Election Workspace and it immediately pulls up the Ballot Details record —no need to navigate around the system first.



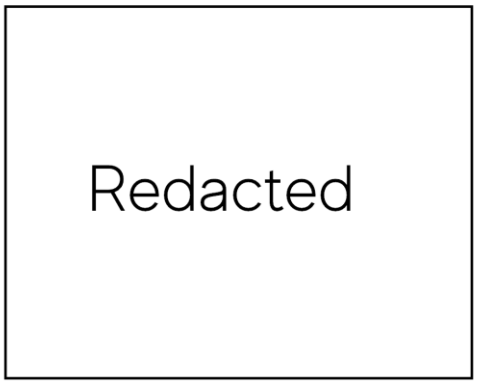
SVRS Quick Search with Ballot ID selected

The full ballot lifecycle is managed from within the Ballot module (**AV-6.7.2**). Though access to the ballot record can be accomplished directly from an individual voter record, all bulk processes are done from within the module.

SVRS supports both Polling Place (Absentee) and Vote Center (Mail Ballot) election models. In the current implementation, the design includes the ability to define UOCAVA voters and election-specific addresses. In addition to this, California allows a "Vote by Mail" flag. This flag is used to indicate whether or not the voter receives an absentee during Polling Place elections. For Virginia, this would need to be expanded to include

statuses (**AV-6.7.6**) (**AV-6.7.11**), sources, and duplicate (**AV-6.7.5**) or multiple applications (**AV-6.7.3**) (**AV-6.7.10**). While a user can create any needed voter correspondence (**AV-6.7.12.1**) for manual issuance is a default functionality in SVRS, tying it to an automated workflow trigger would require additional work.

On this same tab, the user can define one or multiple Voter Assistance metrics for voters who have a disability or impairment that requires accommodation on the ballot or at the poll site. (**AV-6.7.4**) (**AV-6.7.4.1**)



Voter record Election tab details for Voter Assistance

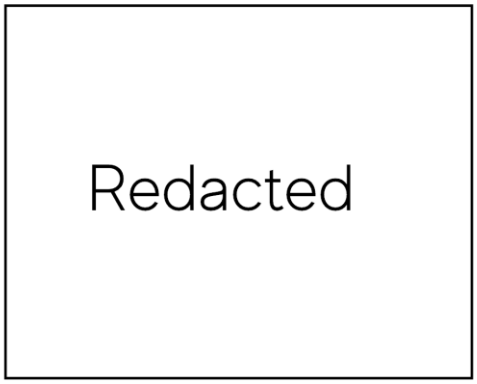
When a voter moves within the State of Virginia to another locality, that jurisdiction obtains control of the voter record when the residence address is updated to their jurisdiction through a registration process. This includes existing or historically information, even absentee ballot applications (**AV-6.7.7**) (**AV-6.7.8**); it should be noted that active ballots for the voter “transfer” too, but are automatically “suspended” when they switch jurisdictions.

SVRS currently stores ballot-specific audit logs in a transactional db, as all actions throughout the system are tracked, but this information is presently visible only behind the scenes. If this is a priority, it can be included on the SVRS ballot details screen (**AV-6.7.9**). However, a voter’s entire statewide ballot history of issued ballots is visible on the Voter record’s Elections tab in the Ballot History section.

4.3.6.8 AV-6.8: Absentee Ballot Applications, Effective Periods, and Deadlines

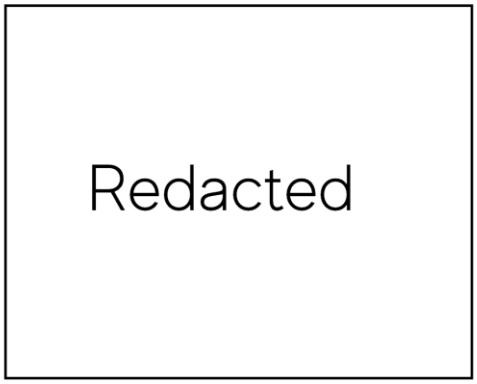
SVRS currently supports Permanent Vote by Mail, Election Specific Vote by Mail, and UOCAVA. Date-Range Absentee status would not be difficult to implement. This is similar to the process in the State of Maryland and would be incorporated with the Virginia Absentee Application process (**AV-6.8.2.1**) (**AV-6.8.2**).

Citizen residents of the state traveling for work, school, pleasure may find themselves out of state overlapping an election. For this, they may contact the election offices to let them know an election-specific mailing address (**AV-6.8.1**). In SVRS, this is done on the Voter’s Election tab.



Voter, Election tab: Add Election Specific Address section

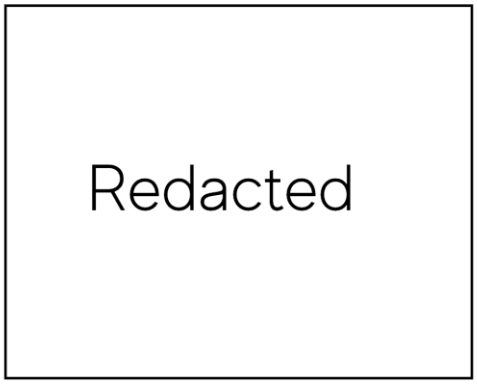
In the Voter Record, the user clicks “Add Election Specific Address” which allows the user to receive a vote by mail (absentee) ballot for one single election (**AV-6.8.3**).



Election Specific Address screen with multi-lingual script examples

The Election-Specific address is freeform data entry. This is to accommodate a wide variety of scenarios. For instance, in the event a voter is in another country where the address needs to be written in English (to exit the United States) and in a foreign script (to reach the voter in the foreign country), SVRS allows special language scripts.

FWAB ballot details are captured in the SVRS Voter History with “FWAB” as the Method (e.g., Ballot Type). This allows county elections staff to indicate that the Absentee FWAB was received, adjudicated, the source of receipt (email, mail, drop off, etc.) the date received, and apply voting history for accepted FWAB (**AV-6.8.5**).



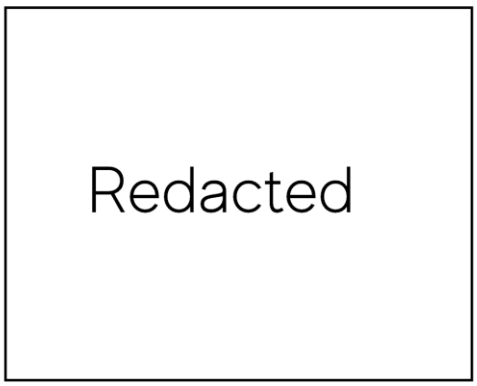
Voting History section of Voter's Election tab

While a user can create any needed voter correspondence (**AV-6.8.2.2**) for manual issuance by default in SVRS, tying it to an automated workflow trigger would require additional work.

The Absentee application process described in this VA requirements implies that the existing SVRS image scanning process used for adding an image while updating the voter record would work well for Virginia. The application fields specific to the State of Virginia absentee application could be added and would accomplish all change of address and other fields in a streamlined view for the users that would benefit from the standardization and validation processes already in-built to SVRS (**AV-6.8.9.1**) (**AV-6.8.10**) (**AV-6.8.11**) (**AV-6.8.11.1**) (**AV-6.8.11.2**)

Pending Voters (excluding valid Underage voters who are 18 on or by election day) are not included in the election workspace, even if they have an accepted absentee application (**AV-6.8.9**).

HAVA requirements are clearly marked on the Voter's Additional Information tab.



HAVA ID and Validation Fields in Voter

A five-day waiting period to delay the addition of a new voter record into an election workspace to delay ballot issuance is possible (**AV-6.8.14**); however, this is not currently in SVRS. In SVRS, the voter is validated in real-

time against Dept. of Health, Dept. of Corrections, DMV and SSA agencies in the present implementation allowing near real-time validation and issuance of ballots.

The HAVA ID required flag on the Voter record means the voter must vote in person and show ID (**AV-6.8.17**) (**AV-6.8.16**) (**AV-6.8.15**).

For Virginia, the absentee application process would need to be expanded to include statuses/reasons (**AV-6.8.7**) (**AV-6.8.7.1**) (**AV-6.8.8.1**) (**AV-6.8.8.1**) (**AV-6.8.11.3**) sources/methods (**AV-6.8.13**), and multiple applications over the course of time.

When a voter moves within the State of Virginia to another locality, that jurisdiction obtains control of the voter record when the residence address is updated to their jurisdiction through a registration process. This includes existing or historically information, even absentee ballot applications (**AV-6.8.12**).

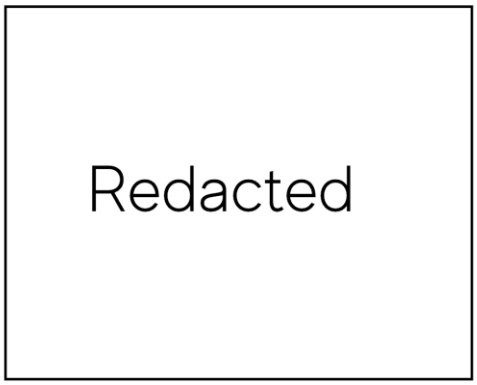
Workflow modifications, unique to the State of Virginia, would have to be defined (**AV-6.8.4**) (**AV-6.8.6**) (**AV-6.8.6.1**).

4.3.6.9 AV-6.9: In-person Absentee Voting

SVRS is optimized to Vote Centers (**AV-6.9.4**) as is the model in the current implementation in the State of California. As a result, SVRS includes an Election Day Portal (EDP) module which would be an ideal solution for In-Person Absentee Voting (**AV-6.9.1**) and any other election day or early voting activities regarding ballots.

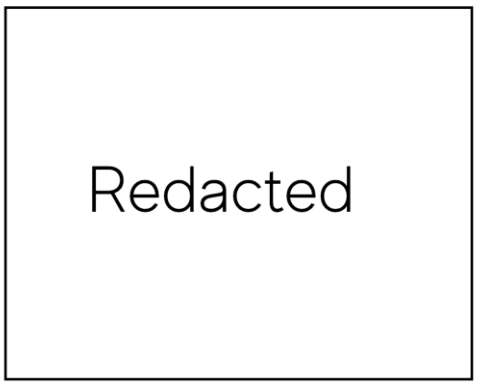
The EDP is not an ePollbook, it is SVRS module that is streamlined for use by election workers or volunteers at the polls. This included module is simplified and supported by automated workflows. The EDP allows for marking a voter as voted, issuing ballots, capturing digital signatures, and even registering. Registration or check-in workflows can be modified to include scanning or recording a voter's Driver's License (**AV-6.9.3**).

All Central Absentee Precinct (CAP) or Vote Center tasks can be performed without exposing the full suite of SVRS tools or data to temp support. If desired, the State of Virginia chose to integrate SVRS with existing Pollbook tools instead similar to other modules in SVRS.



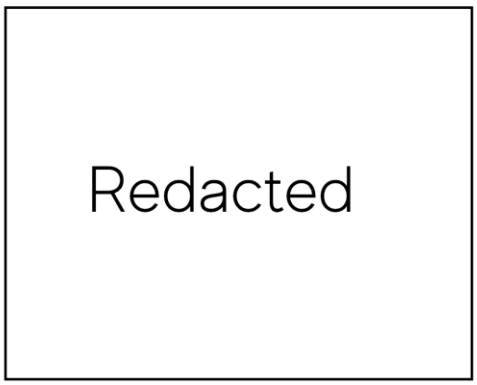
Election Day Portal (EDP) Login Screen

Like SVRS, the EDP is developed for use on a web browser. However, the UI is optimized for use on devices as small as smart phones or tablets. As a result, it offers a range of flexible options to counties to use any existing tools.



All Ballots Issued or Modified are Now Tied to the Location it occurred without Relying on Data Entry

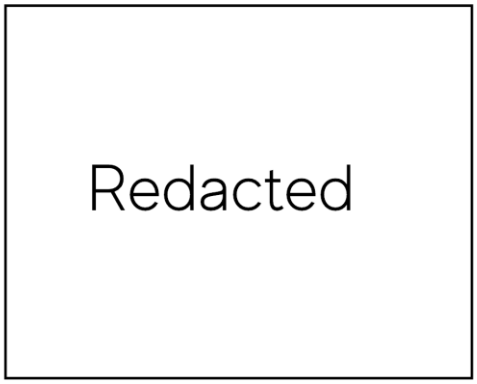
Users are associated to the Election and the Location, ensuring that all ballots issued already have their location of issuance associated, reducing the data entry performed by the end user.



Search Voter

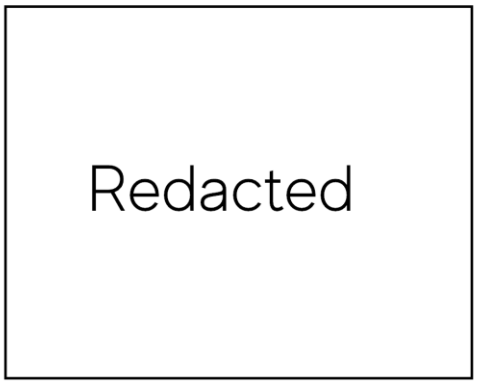
The basic EDP Workflow seeks to locate the election voter, SVRS determines for the worker whether the voter can receive a new Absentee Ballot, if the voter needs a Provisional, or mark if the voter is voting a standard ballot at the polls.

The SVRS EDP could be based for Virginia's specific workflows; For instance, with the State of California, the EDP workflows capture the Provisional application information, including digital signature capture.



EDP Signature Capture block

Since the EDP is just a streamlined SVRS, it follows the same duplicate voter rules and validations as any other registration process; including DMV validations, etc. Because all of the data about the voter is readily accessible, it can



Print Items Screen – Fully Customized

The ability to Print a ballot, ballot envelope label, a voter registration receipt (**AV-6.9.2**), or more can be customized based on available needs or state-specific workflows.

4.3.6.10 AV-6.10: Denied Absentee Voter Application

SVRS records Election-Specific Absentee Applications and UOCAVA ballot applications; however, the current implementation would require reject/denied workflows and correspondences to be developed. (**AV-6.10.1**) (**AV-6.10.2**) (**AV-6.10.3**) (**AV-6.10.3.1**) (**AV-6.10.3.2**).

SVRS comes with the Correspondence Configuration to allow the production of unique voter correspondences, which would include Denial Notice (**AV-6.10.3.3**) or Non-Registrant Absentee (**AV-6.10.3.5**) with manual issuance. Automated Issuance of such notices based on workflows would have to be customized. SVRS supports voter ballot languages, but does not currently transform custom-notices inside of SVRS (**AV-6.10.3.4**).

4.3.6.11 AV-6.11: Ballot Mailing Address Provisions

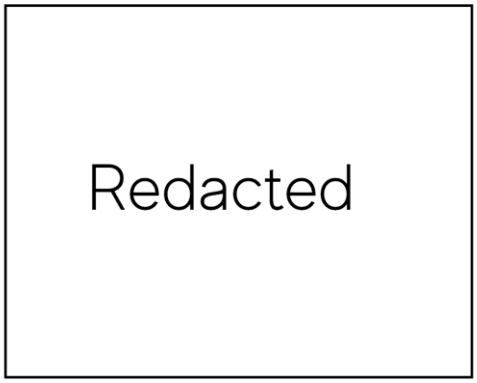
If a voter moves during the election and is voting by mail, what occurs to their ballot in SVRS depends entirely on the nature of the move (**AV-6.11.3**).

- If the voter moves out of boundaries BEFORE receiving a ballot, SVRS does not issue them a ballot.
- If the voter moves AFTER receiving a ballot, but BEFORE returning the ballot:
    - AND moves out of the state, their record is Canceled, SVRS suspends their ballot (**AV-6.11.4**).
    - AND moves out of the election boundary, SVRS suspends their ballot.
    - AND moves within the election boundary, but does NOT change ballot styles, no change.
    - AND moves within the election boundary, but DOES change ballot styles, SVRS suspends their ballot
        - IF this move is also before the “mail by” deadline for the election (ex: e-7), SVRS automatically generates a new ballot in the Pending status for the jurisdiction to print and send to the voter.

Returned or Adjudicated ballot statuses are not affected by voter record moves, as the voter returned their ballot prior to moving.

SVRS does not prevent the user from marking a Suspended ballot to the “Returned” status, nor does it prevent the jurisdiction from adjudicating the now returned ballot.

In addition to the election-specific address, the voter can have a domestic or international mailing address associated to their record. (**AV-6.11.2**) (**AV-6.11.2.1**) (**AV-6.11.1**)



International Mailing Address example

4.3.6.12 AV-6.12: Generate Absentee Ballot Applications

While a user can create any needed voter correspondence (**AV-6.12.1**) for manual issuance by default in SVRS, tying it to an automated workflow trigger would require additional work.

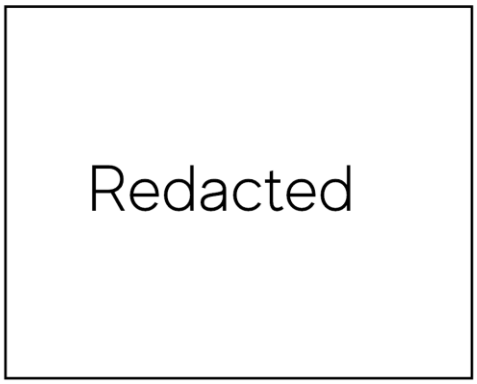
SVRS does not restrict privileged users from managing voter record details, including absentee information (**AV-6.12.2**).

4.3.6.13 AV-6.13: Sending Ballots

SVRS is designed specifically to support Mail Ballot elections by streamlining the process, as well as processing in bulk (**AV-6.13.6**). Once the ballots are generated for an election, the system automatically begins creating the records to Issue ballots (**AV-6.13.8**) to eligible voters to their correct address and for their correct style (**AV-6.13.4**).

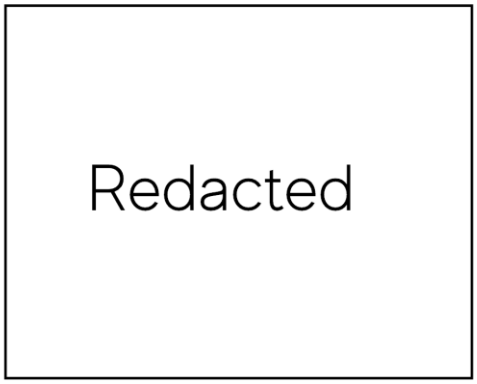
SVRS comes with the following basic ballot types: Vote by Mail ("Absentee"), Provisional, Standard ("Poll"), and FWAB. (**AV-6.13.3**) More can be added as needed; for instance, the State of California additionally as Conditional Voter Registration ballot types, which have their own unique workflows built to support Vote Center elections there.

SVRS's Perennial Header includes a Ballot ID quick search which allows users to barcode scan or hand-type a Ballot ID for their Election Workspace and it immediately pulls up the Ballot Details record (**AV-6.13.13**)—no need to navigate around the system first.



SVRS Quick Search with Ballot ID selected

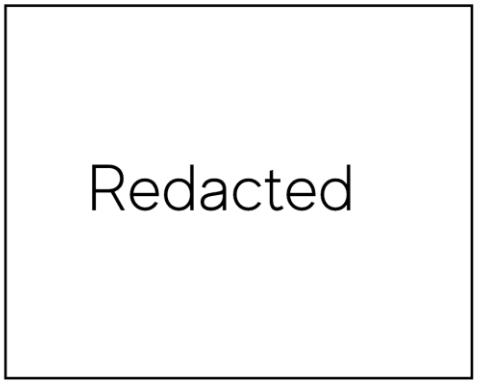
The SVRS Election Day Portal (described in more detail in AV-6.9) identifies all the materials and even envelope color for the end user. It is ideal for assembling one-off ballots (**AV-6.13.10**) In the example below, it is showing Provisional:



SVRS EDP Ballot Issuance

Similar to Voter Registration, election-specific scan templates and signature blocks are defined in System Configuration. For the State of Virginia, other scan and envelope needs can be accommodated but need additional details. It is possible to print ballot labels, ballot envelopes, or send to a ballot printer. This would need to be configured.

In the Election workflow in SVRS, the county users must generate ballot styles, then the users generate the voter workspace.

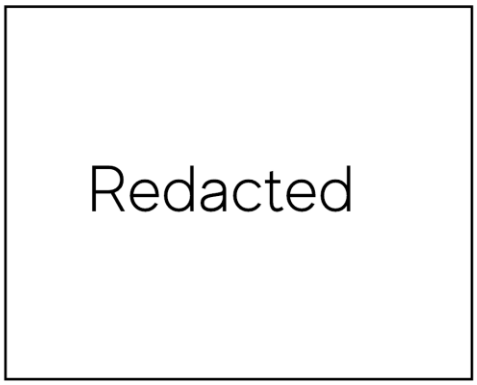


Pending Ballots Extract Data Queue

Eligible voters enter the workspace bringing with them any UOCAVA status, delivery preferences, disabilities, language preferences, and contact information. The election voter is associated to their Ballot Style based on their Home Precinct, Political Party (if applicable), and if they are a Federal-Only voter.

Ballots can be Issued (Mailed) all at once, or in smaller sections. For instance, a jurisdiction may need to have all UOCAVA ballots printed first or separate out ballots of a particular election language. On the Bulk Issue Queue (below), users input the anticipated Mail by date promised by the Print Vendor or performed within office. (**AV-6.13.14**) (**AV-6.13.14.1**)

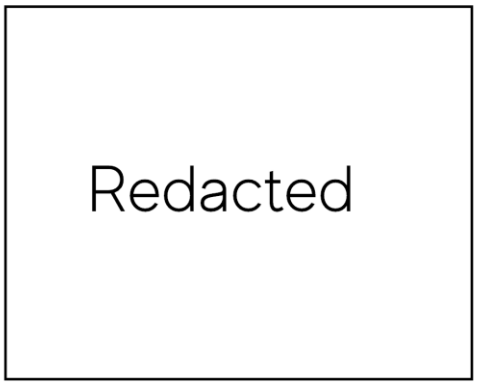
SVRS only allows one Delivery Method to be selected at a time (e.g., mail cannot mix with e-mail). This is defined based on the UOCAVA preference.



Delivery Method Options

These can be extracted for a Print Vendor (**AV-6.13.12**) or Printed Now in the office. SVRS produces a ballot envelope label (configured) (**AV-6.13.11**). Additional details to support State of Virginia unique workflows or data collection needs can be added. SVRS would need to be configured to support Ballot on Demand printers in the State of Virginia, as this varies depending on the machine (**AV-6.13.9**).

Manual Print or Replacement (**AV-6.13.15**) can be done directly from the voter's Ballot Record (**AV-6.13.7**) and includes the ability to manually indicate Email, Over the Counter, SMS (**AV-6.13.2**). Other types can be configured as needed.

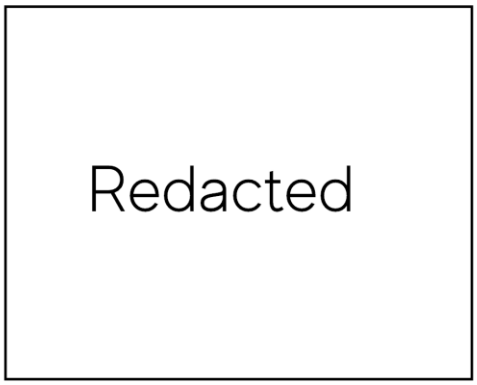


Issue Methods for Manual Issue Ballots

Alerts, triggers, workflows unique to the State of Virginia would need to be configured (**AV-6.13.1**) (**AV-6.13.15.1**);

4.3.6.14 AV-6.14: Returned Ballots

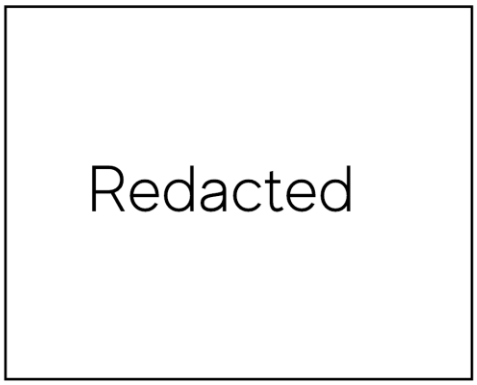
FWAB ballot details are captured in the SVRS Voter History with “FWAB” as the Method (e.g., Ballot Type). This allows county elections staff to indicate that the Absentee FWAB was received, adjudicated, the source of receipt (email, mail, drop off, etc.) the date received, and apply voting history for accepted FWAB (**AV-6.14.6.1**) (**AV-6.14.6**)



Voting History section of Voter’s Election tab

During Voter Registration update processes, jurisdictions can enter any form that is appropriate to accept into the county, including the Federal Postcard Application (FPCA) (**AV-6.14.7**) (**AV-6.14.7.1**)

On the Voter Registration form, users can define one or multiple Voter Assistance metrics for voters who have a disability or impairment that requires accommodation on the ballot or at the poll site. It is on the SVRS Roadmap 2021 to add this as a filtering option during ballot issuance and to help support RAVBM. (**AV-6.14.8**) (**AV-6.14.8.1**)



Voter record Election tab details for Voter Assistance

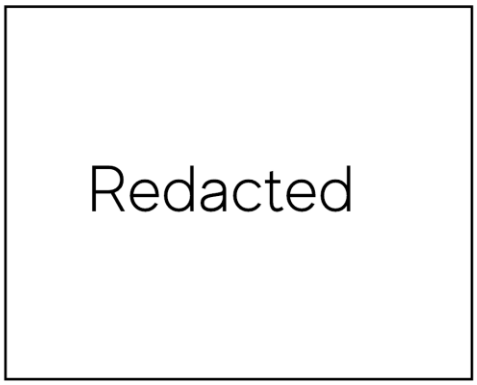
Similar to the Bulk Scanned Application processes shown with the Voter Registration Use Case, the working queue for Returned ballots is consistent with other SVRS functionality.

Returned ballots can be entered in two ways:

(1) Import from Ballot Scanning Machines
For the State of Virginia, this would be a point of configuration depending on the output of the scanning and signature verification machines used by the county offices.

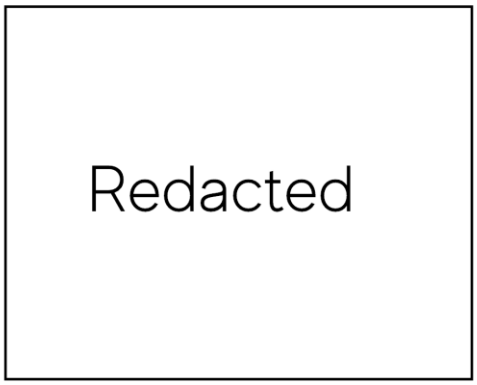
(2) Manual Scan of Ballot Envelopes (the record can be access via barcode scan (**AV-6.14.3**))

(3) Manual Marking of the Ballot as Returned without Scanning (the record can be accessed via barcode scan); while this is allowed by SVRS, it is not recommended to by-pass a scan process as this is the only way the signature image can be captured for side-by-side adjudication.



VBM Search Screen: Receive & Process Queue

After the user enters the Queue, they can select a queue to be processed, upload more data, or scan in small batch returns that are not going to go through the Automatic Signature Verification (ASV) machines.

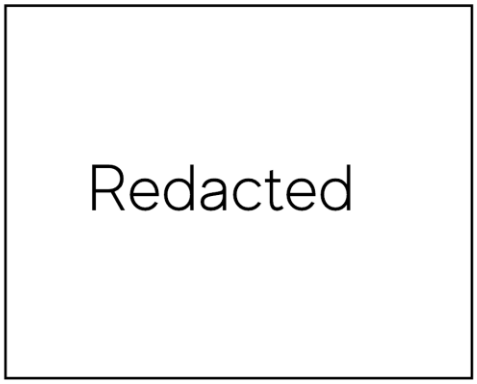


Receive and Process Queue: Upload or Scan Options

Ballot scanning and verification machines produce a document with the result of their envelope scan. 1: Accepted, 0: Needs Review. They include image of the full envelope, including the signature, and typically the ballot id, voter id, or both.

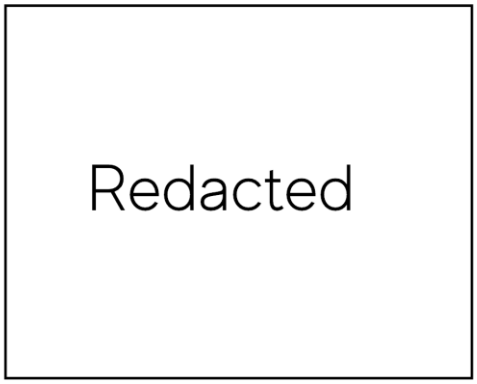
During the upload process, Accepted Ballots are automatically consumed the appropriate record is updated. In the event the Suspended ballot is accepted, the Issued ballot moves to the Suspend status. In the event that a red flag scenario is triggered, the ballot is placed into Received – Unprocessed.

Ballots that Need Review go into the Received – Unprocessed queue.



Ballot Status Queue

Similar to Scanned Image queue processing in Voter Registration, the user simply clicks Process Next. The record is only individually locked, so many people can jump into a single queue.



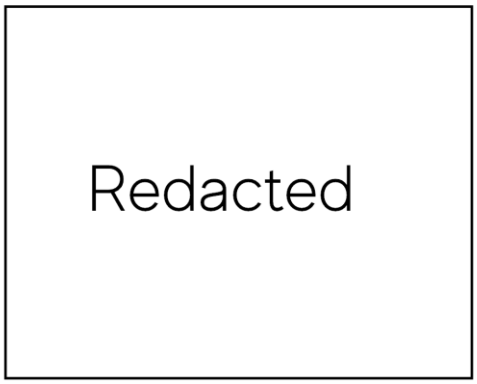
Top Half of the Ballot Queue Record

The Source, Location, Date Receive is along the top of the record. The Top half of the ballot Return form is broken into three sections: Quick Voter Details, Voter Current Signature with Signature Effective date, and Ballot Signature and Ballot Physical Location (if defined, ex: “Bin 345”)

Ballots can be manually set to the Returned from the Ballot Details screen, making it is possible to add ballots to the processing queue without an associated image—it is possible the initial scan was a poor-quality scan. As a result, SVRS provides the ability to attach or re-scan a ballot image directly while processing.

There are three sub-sections on the bottom of the side-by-side signature review for voters where they were rejected by the Automated Signature Verification machines.

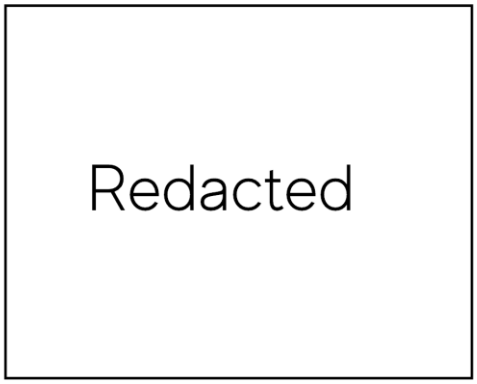
The most-important aspect of this is section is the Red, Yellow, Green Adjudication Flag (described in more detail in the following section). Green means SVRSs detailed assessment see no concerns with the voter, and the signatures are the key piece of information.



Adjudication Flag: Green "Voter OKAY" Flag

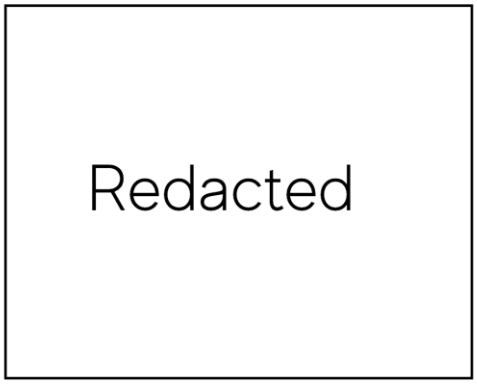
The first section is the Compare All Versions of the Voter Record. For the State of California, there is a third record displayed here "State Version" due to the nature of the system there. This is currently cropped out of the image below.

In the State of Virginia, there are two key records: Official List Record (the version of the voter in the Election Workspace frozen at Close of Registration) and the Current Voter Record (the active voter record still being affected by online registration and other automated procedures. SVRS highlights any difference between the records for a quick visual indication if there are any data differences of concern for the processing of the record.



Bottom Half of the Ballot Queue Record: Voter Record Comparison section

The user can select the active ballots button which displays ALL ballots for the voter across the entire state for the same election. This section shows Suspended status ballots from a neighboring county for a voter who may have moved into the user's county during the election timeframe. Or, if they have a Returned ballot in the current or neighboring county.



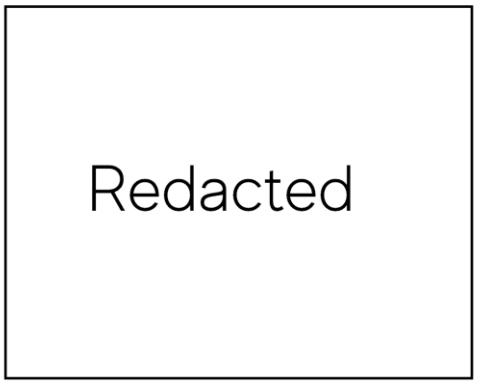
Bottom Half of the Ballot Queue Record: Ballots Issued to Voter Across the State

! Note: a frequently asked question that pops up at this time is usually—

Q: What does SVRS do if MY county Accepts a returned ballot first before another county does when a voter has two (duplicate) returned ballots for the same election?

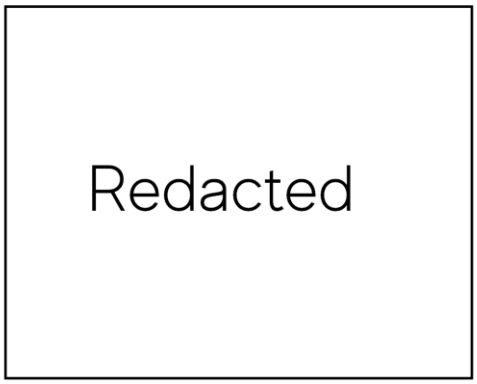
A: Because your jurisdiction accepted the ballot first, the neighboring jurisdiction now receives a Red Flag scenario if they attempt to Accept this duplicate ballot. They will be able to see the second ballot issued to your jurisdiction and that it is in the Accepted status.

The final section is the Outstanding List Maintenance. Outstanding List Maintenance is one of the yellow flag scenarios in SVRS. Users can see if there is a Low-Confidence Duplicate match outstanding, Felon records, National Change of Address, etc. They can double-click on the record and process it from the Ballot screen.

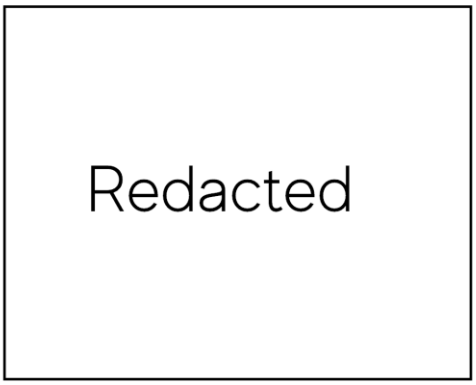
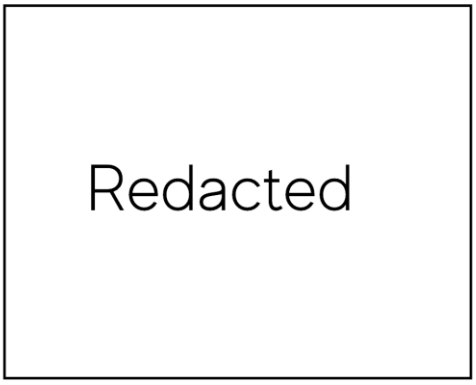


Bottom Half of the Ballot Queue Record: Outstanding List Maintenance Associated to the Voter

When completed SVRS offers a key-friendly data entry for adjudication:



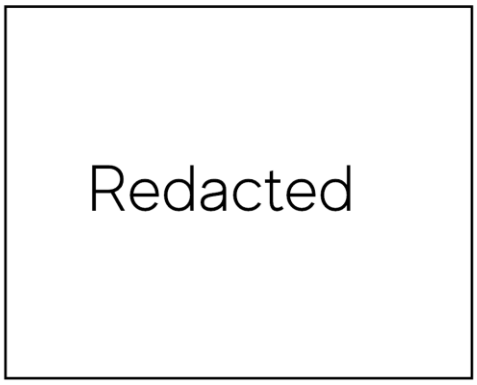
SVRS Quick Adjudication Buttons with Hotkeys

Option	Result
	Accepts ballot and moves to next to be processed. Removed from Received – Unprocessed queue.
	Rejects ballot, prompts user to select reason, then moves to the next to be processed. (**AV-6.14.12**) Moves from Received – Unprocessed queue to Reject – [Reason] queue (ex: “Missing Voter Signature”)

Option	Result
Redacted	Rejects ballot for the Reason “Needs Review” and moves to the next to be processed. Moves from Received – Unprocessed queue to Reject – Needs Review queue.

The user selects the Reject Reason (**AV-6.14.12.2**). SVRS default reject reasons include selections such as “Missing Signature” and “Ballot Not Received on Time” intended for effective reporting.

Reject Reasons are configurable in System Configuration, Virginia can expand the Description to include the business rule (**AV-6.14.12.3**). The public portal could also be expanded to tie Business Rules to Reject Reasons selected, so the voter sees more details on the Public Portal vs what an end user would select while adjudicating.



SVRS Reject Reason prompt, navigate with type-ahead or arrows

While voter and election correspondences can be configured (**AV-6.14.12.4**) for manual issuance in SVRS, automated workflow triggers would have to be configured (**AV-6.14.12.1**). SVRS does not currently offer automatic transformation of languages for correspondence; however, the extraction for a print vendor can include the language column if they offer this service (**AV-6.14.12.5**).

Ballot Issuance and Adjudication Flags

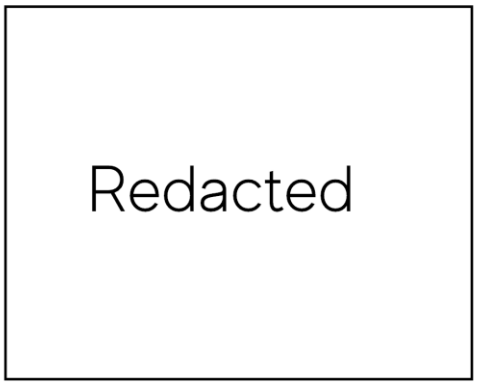
Issuance and Adjudication flags affect both Provisional and Mailed ballots.

Flag	Triggers
Redacted	• NO Voter record exists • The Voter is NOT a valid voter in the Election Workspace (ineligible) • The Voter has an ACCEPTED ballot for the SAME election anywhere in the County/State/Country Provisional ballots CAN be Issued without override. Mail ballots can NOT be Issued without a user who has the Ballot Override Privilege. Provisional and Mail ballots can NOT be Accepted without a user who has the Ballot Override Privilege. Redacted Red Alerts do not require the Override privilege to Issue a Provisional type ballots.

Flag	Triggers
Redacted	• The voter has outstanding List Maintenance activities (such as potential duplicate records, potential felon, potential NCOA). *Duplicate Match: "An unprocessed duplicate match record exists for the voter. Please processed this before continuing [Open Duplicate Match Record Link]."* • The voter exists as a "Poll Voter" in another county participating in the same election. This is a voter who was an Active voter in another county prior to the close and has no VBM ballot. This voter is now attempting to register again in the current county after the Close of Registration. SVRS does not prevent Issue or Accept status for yellow flag situations. This is informational. The user has the option to back-out and do more research if desired. Redacted
Redacted	There are no red or yellow alert issues. County user can issue a ballot. This is informational.

Provisional ballots are a ballot option that is used to ensure voters who fall into odd scenarios (did not receive their ballot and have not returned one, do not have a standard address, or voters who do not qualify for a ballot

based on system information) to have an opportunity to receive a ballot, while providing county officials an opportunity to post-process the unique scenarios.

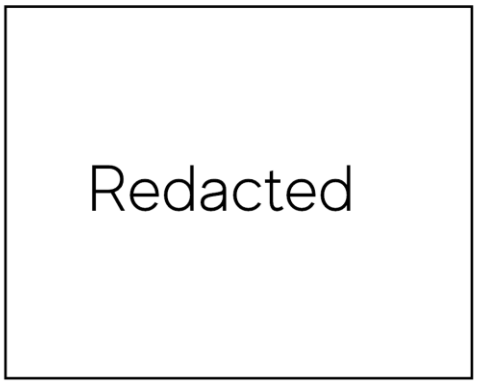


Provisional Ballot Search screen

While the Election Day Portal (EDP) is the ideal tool to issue Provisional Ballots, the EDP is just a simplified SVRS interface, and it is equally possible to issue Provisional ballots to person via SVRS.

Issue a Provisional Ballot (**AV-6.13.5**)

Issuing Provisional ballots via the system or EDP ensures that a readable Envelope label with a barcode is included on the potential voter's envelope to ease the effort to post-process.

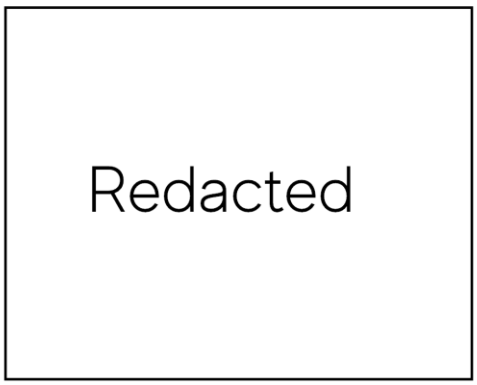


SVRS "Add Provisional" Record Search Screen

SVRS strives to connect the Provisional envelope with the record in advance vs during post-election processing. This serves to reduce the burden of data entry from poor handwriting samples. However, SVRS allows the entry of a Provisional envelope post-election.

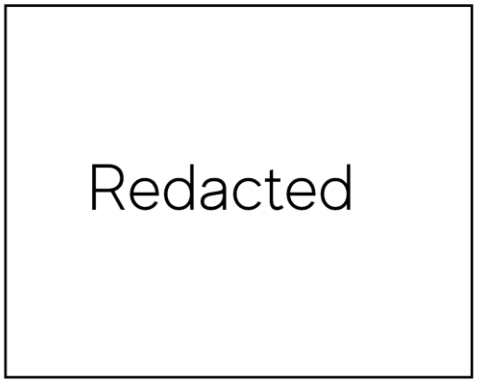
Where no record can be found, SVRS produces a label with all the collected data to generate a provisional label and record at the moment of ballot issuance. This is done via both the SVRS EDP and SVRS primary UI.

On the Provisional Record search screen, the user would Add Provisional. SVRS then produces a second search screen which can search both the Election Workspace, or the county's entire Voter database to associate a Voter to the Provisional.



Issue Ballot to a Voter

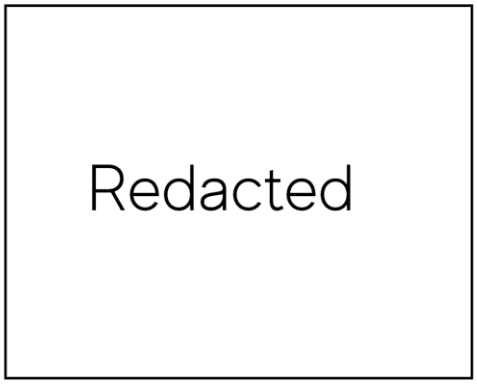
Or, in the event NO voter record can be found in the search results, the user can select "Voter Record NOT Found" button and add the Issue Reason to the No Voter Provisional form as well.



Add Provisional: No Voter Record Found scenario

The user enters the provisional application information including ballot style. The only required field is Last Name. In the EDP, the screen can be passed to the voter.

In both scenarios, a Provisional Ballot record is created. SVRS does not restrict the number of times a voter is Issued a Provisional Ballot. SVRS creates a unique ballot ID for each Issuance attempt.

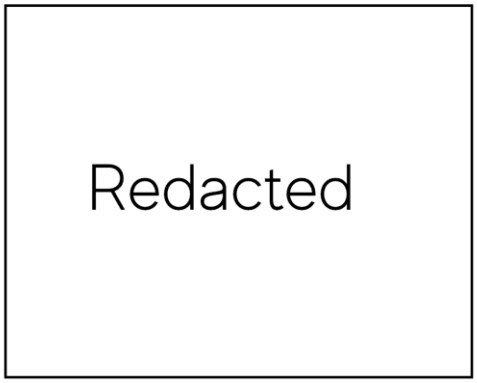


Multiple Provisional Ballots for the Same Voter

All Provisional Ballots appear in both the Voter Record and on the Mail Ballot “All Ballots” section.

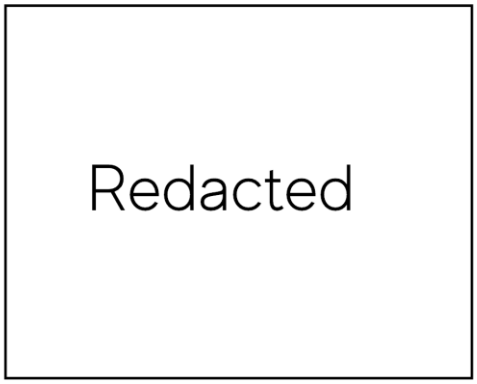
Receive Provisional Ballot

The Receipt of a Provisional Ballot is very similar as in Mail ballots. Users can Scan or Upload images (likely scan) to capture the image. The ability to process documents with optical character recognition is on the roadmap.



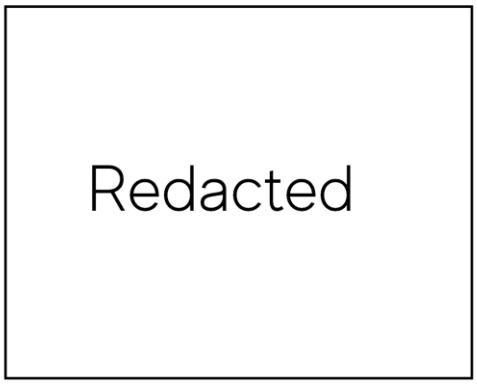
Provisional Ballot details

The record can be manually adjudicated directly on the Provisional ballot record—and the scanned signature can be compared to the voter record; the record can be Returned and added to the Received – Unprocessed Queue. Similar to Mail ballots. Other validations can be customized.



Provisional Ballot Status Options

Unlike Mail ballots, there is no concept of a “Pending” or “Suspended” Provisional ballot. SVRS allows a “Returned” status to indicate the ballot still needs to be adjudicated but is no longer in the possession of the potential voter. Otherwise, the user can indicate **** if the ballot is Accepted or Rejected (Reasons Configurable).



Red, Yellow, Green Flag

Similar to Mail ballots, the red, yellow, green adjudication flags appear on the Provisional record. In this sample case, the record for James Russel is an eligible voter in the election workspace, has no other accepted ballots in the state for the election, and no outstanding list maintenance.

There are six ballot statuses in SVRS: Pending, Issued, Suspended, Returned (**AV-6.14.4**) (**AV-6.14.2**), Accepted, Rejected. There are only two statuses for Returned ballots: Accepted or Rejected (**AV-6.14.10**). Closed Primaries affect ballot type associated to a voter; the “Party” field is an available search option in Ballot Search (**AV-6.14.9**). Spoiled/Undeliverable/Reissue Ballots are in the Suspended status, “Unused” can be added as a Reason for the Suspended status. (**AV-6.14.1**).

The location of ballot processing is not restricted in SVRS. This would include processing at Central Absentee Precincts, if that is desired. (**AV-6.14.5**) This is an option in California as well at Vote Centers. In regards to curing, this varies depending on state needs—In California, they have opted to display Reject Reasons on public websites (e.g., Rejected, Missing Signature). Voters would have to view this and independently come

into the office to cure—SVRS does not restrict the modification of Reject to Accept status if the voter comes in and resolves issue (**AV-6.14.11**)

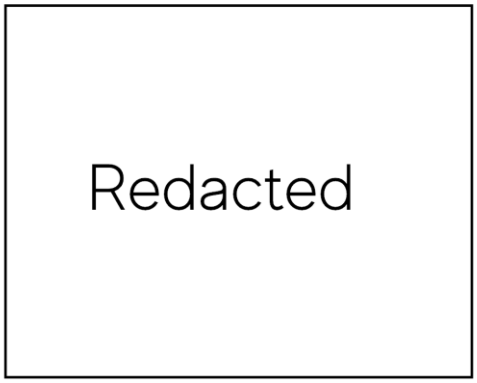
4.3.6.15 AV-6.15: Registration Interactions

If a voter record is Canceled (ex: Death) during the election and is voting by mail, SVRS moves Issued status ballots to the Suspended status. (**AV-6.15.3**). For details on how SVRS reacts to voter changes, see the response provided in AV-6.11 (**AV-6.15.1**)

Returned or Adjudicated ballot statuses are not affected by voter record status changes, as the voter returned their ballot prior to having the status changed, even in the event of a death.

SVRS does not prevent the user from marking a Suspended ballot to the “Returned” status, nor does it prevent the jurisdiction from adjudicating the now returned ballot. In the scenario of a voter death, sometimes a family member may return a voted and signed ballot envelope on behalf of the now deceased voter, or it was in the mail at the time of the voter’s death.

In regards to List Maintenance during ballot adjudication (**AV-6.15.2**), SVRS includes an Outstanding List Maintenance details. Outstanding List Maintenance is one of the “Yellow flag” triggering scenarios in SVRS for ballot adjudication. In the List Maintenance details, users can see if there is a Low-Confidence Duplicate match outstanding, unprocessed Felon records, National Change of Address, etc.



Bottom Half of the Ballot Queue Record: Outstanding List Maintenance Associated to the Voter

To expand on this, SVRS includes validations called “Red, Yellow, Green” adjudication flags.

Flag	Triggers
Redacted	• NO Voter record exists • The Voter is NOT a valid voter in the Election Workspace (ineligible) • The Voter has an ACCEPTED ballot/participation for the SAME election anywhere in the Jurisdiction/State Provisional ballots CAN be Issued without override. Mail ballots can NOT be Issued without a user who has the Ballot Override Privilege. Provisional and Mail ballots can NOT be Accepted without a user who has the Ballot Override Privilege. Redacted Red Alerts do not require the Override privilege to Issue a Provisional type ballots.
Redacted	• The voter has outstanding List Maintenance activities (such as potential duplicate records, potential felon, potential NCOA). *Duplicate Match: "An unprocessed duplicate match record exists for the voter. Please processed this before continuing [Open Duplicate Match Record Link]."* • The voter exists as a "Poll Voter" in another county participating in the same election. This is a voter who was an Active voter in another county prior to the close and has no VBM ballot. This voter is now attempting to register again in the current county after the Close of Registration. SVRS does not prevent Issue or Accept status for yellow flag situations. This is informational. The user has the option to back-out and do more research if desired.

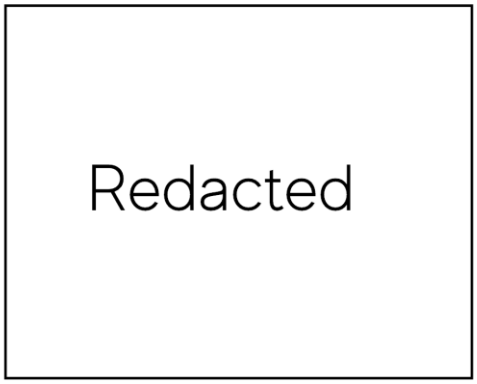
Flag	Triggers
	Redacted
Redacted	There are no red or yellow alert issues. County user can issue a ballot. This is informational and does not produce a prompt.

4.3.6.16 AV-6.16: Voter Credit

“Standard” ballots (voted at a voting location) added via the EDP are immediately added as Voter Participation History for an election as COUNTED; all other ballot types (ex: Absentee) are added as voter history when an election is Certified in SVRS which posts all participation history to the voter’s record (**AV-6.16.1**)

On the Voting History tab, the system records Date Voted, Election Name, Vote Location, County (Locality Name), Voting Precinct, Method, Counted, Reason, and Action (Delete/Edit). (**AV-6.16.1.1**)

The ballot style issued for a particular absentee ballot to a voter is contained in the ballot details screen, but isn’t on the Voting History section (**AV-6.16.2**).



Voter Record – Elections tab "Ballots" and "Voting History" section

4.3.7 Election Results & Certification

4.3.7.1 Vote Results Reporting Administration

SVRS provides authorized staff the following system administration functionalities related to election results reporting:

- Control Vote Reporting – Start/Stop vote reporting, set timeframe to prevent entering election results, change reporting mode (auto/manual) (ERC-7.2.8)
- Configure county election reporting – For each county configure the report mode such as Key Data Entry (KDE county) and File based (Auto reporting county). Enable/disable receipt of vote report from county.
- Track the Election Certification Process (ERC-7.1.1)

4.3.7.2 Unofficial Voting Results

On an election night, unofficial vote results are received on SVRS via Key Data Entry (KDE), Auto County Template File, or Auto County API. The three types of reports used throughout the unofficial canvass period for election night are Regular, Final, and Update. Regular reports are submitted during an election night. Final report is submitted when 100% of precincts in the county have reported on election night. If the final report is not processed successfully, the county is not considered to have completed its Election Night process. As needed county can send update reports to final. (ERC-7.5.2)

Periodically throughout the election night, locality staff can use KDE to submit valid and invalid (overvotes) votes cast for each candidate, including write-in, and referenda by precinct and voting method (e.g., in-person, curbside, absentee, central absentee precinct, provisional). Alternatively, county can upload the results from tabulation machine as file through secure portal. Depending on the local business process, vote results can be aggregated and reported from County headquarters or results can be reported directly from preconfigured voting locations and central absentee precincts. (ERC-7.2.1, ERC-7.2.2, ERC-7.2.3, ERC-7.2.5, ERC-7.2.7)

As vote results are received, SVRS aggregates the precinct level reports and determines which precincts have reported vs not reported, if reported the last report time. The precinct reporting and individual precinct level reports are made available to the SVRS Election Night Reporting and to external agency via API Reporting Feeds. (ERC-7.3.1, ERC-7.3.1.1)

SVRS conducts data validation check on incoming vote results and rejects the file uploaded if critical error exists. SVRS provides an history log of all the vote results received along with the status of submitted or rejected. If rejected, the error messages are logged for the staff to review and resolve. (ERC-7.2.6, ERC-7.2.6.1).

Update Reports are used only after a successful Final Report has been submitted by the county. Any updates after that can be made only with the proper reason code. (ERC-7.2.8).

4.3.7.3 Provisional Ballot

Provisional ballots can be issued seamlessly through SVRS Election Day portal (A mini-VR Solution with ePollbook workflow), the voter's registration information is recorded at the time of issuance. After the adjudication, the status of the ballot as counted or not is recorded. If not counted then a reason is captured. The non-counted reasons include non-registrant, already voted, and so on. ELECT can extract the statistics on Provisional Ballot broken down by status as counted vs non-counted and the various reasons for completing the EVAS reports. (ERC-7.4.2, ERC-7.4.2.1, ERC-7.4.2.2)

When provisional (or absentee ballots) are adjudicated as not-counted (or rejected), based on the reason code specific notices are queued automatically for the voter. Staff can print these notices from the notice service queue. (ERC-7.8.1).

4.3.7.4 Ballot Status Reports

Within the unofficial canvassing period when all unprocessed ballots are adjudicated, counties can submit updates to the unofficial canvass vote tally. As part of the Update Report, locality staff submits the total number of unprocessed ballots by ballot type such as Absentee, Provisionals, and Other.

Staff can use the ballot reconciliation reports within SVRS to publish ballots status reports for election night reporting. For absentee ballots, staff can confirm the counts on ballot reconciliation report and submit the ballots issued, counted, and received on or before election day vs after election day up-to acceptance period. For provisional ballots, staff can confirm and submits the ballots issued, remaining, counted, and non-counted with reasons including ineligibility (non-registrant). (ERC-7.5.3, ERC-7.5.4)

4.3.7.5 Voter Turnout Reports

Within the unofficial canvassing period, staff can use the voter turnout statistics report to confirm the voter turnout. The statistics are automatically generated from the vote results entered via KDE or loaded via tabulation files. The reports breakdown at the precinct level the number of voters by voting method such as in-person, curbside, absentee, provisional. (ERC-7.4.1, ERC-7.4.3)

4.3.7.6 Election Night Reporting

When ELECT turns on the Election Night Reporting for an election, SVRS begins to generate an Inquiry Dataset at a configured interval typically every five (5) minutes. Inquiry Dataset is a 'snap-shot' of the most recently received vote reports that are segmented and aggregated by Contest and County. Inquiry Dataset contains the contest and vote data necessary for creation of Vote Results that are published on the Election Night Reporting public website. Inquiry Dataset can be designated as either official or unofficial. When a data issue is noted on a vote report, ELECT can stop the generation of inquiry dataset, roll-back to a good dataset, and re-start the process after the issue is resolved.

Extract of the election results information are produced from Inquiry dataset. Extracts include the election information, precinct reporting, absentee ballot status report, provisional ballot status report, and vote results by candidate and referenda. SVRS provides a secure and interactive Election Night Reporting website that public can view the election results information. Election Results can be downloaded from website in tabular formats. (ERC-7.5.1, ERC-7.5.2, ERC-7.5.3, ERC-7.5.4, ERC-7.5.5, ERC-7.5.6)

4.3.7.7 Certify winners

After staff confirms all votes have been entered or uploaded, they can proceed to the SVRS vote results tab within Election module to review, update, and confirm the winners determined by SVRS' for contents and referenda. Vote results tab notifies the staff if there are unconfirmed contests or contest that should be marked for recount. At the end of canvassing period, Staff certifies the winners. This queue notices for winners. Staff can print these notices out to be mailed. Winners are also automatically added as incumbent to the offices they contested for. (ERC-7.5.7, ERC-7.5.7.1)

4.3.7.8 Post Voter Credit

For localities using the SVRS Election Day Portal and Ballot Management, this is going be a one click process. Staff can use the Post Vote Credit option on Election module to initiate the process, SVRS then compiles the absentee and in-person voting data and credits each voter who has participated. The voter participation includes the election, voting method (in-person, absentee, provisional, curbside), counted vs not counted, party preference, and voting location. The voter participation is associated to the voter record and can be queried for client services requests or analysis. If an inactive voter participated in an election, they are automatically activated.

When locality uses an e-pollbook then SVRS provides import option to seamlessly retrieve the election day activity including who voted, voting method, and if provisional voter then the voter registration information. If locality has captured the election day voting using traditional paper poll book, staff can upload a text file with the list of voters who voted at the polls. The text file can be created with Windows Notepad application and barcode scanner. Staff can also post voter credit manually for one-offs.

(ERC-7.6.1, ERC-7.6.2, ERC-7.6.3, ERC-7.6.4, ERC-7.6.5, ERC-7.6.5.1)

4.3.7.9 Close Election

Staff can ‘close’ the election and prevent any further edits. However, the election results are available for extract, query, and analysis. Under the Election Results, user can review the votes cast by candidate and referenda filter by precinct, voting method, office and so on. Under the Election Data Analytics services, staff can create ad-hoc queries for analysis. (ERC-7.7.1)

When staff needs to make edits to election results, an authorized staff should reopen the election with the reason for why they are doing so. (ERC-7.7.2)

SVRS can handle multi-year voting history and election results so has not found the need to archive results. However, Team Canton will work with ELECT to provide an archive workflow. (ERC-7.7.3)

4.3.8 Correspondence & Reports

SVRS solution offers Notice service and Election Analytics to manage Correspondence and Reports requirements respectively. The following section discusses the capabilities of Notice Service and Election Analytics and how it addresses SVRS requirements.

4.3.8.1 Notice Service

Notice service allows ELECT and localities to send notices for voters, candidates, office holders, and so on. This involves notice template setup, queue voters to be sent notice, sending the notice, and tracking responses from voter.

Manage Notice

SVRS Notice services provides multi-channel communication with voters including email and text messages. Jurisdictions can save cost and avoid sending communication to invalid email or phone number.

Notice Service under System Configuration in SVRS solution, allows ELECT or locality to define statewide and local notices respectively. ELECT can share the common template to localities. Staff can setup notice workflow to automatically generate notices based on status reason codes. Notices can be controlled to not generate for administrative changes to voter records such as updates to middle initial. (CAR-8.1.8).

As shown in Figure 6: Create Notice, staff creates a notice template for each domain such as voter, election and so on.

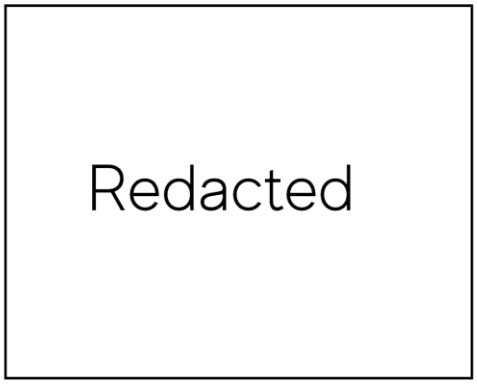


Figure 6: Create Notice

As shown in **Error! Reference source not found.**, staff can determine how many days to wait for voter's response. If voter fails to respond what status to move the voter to.

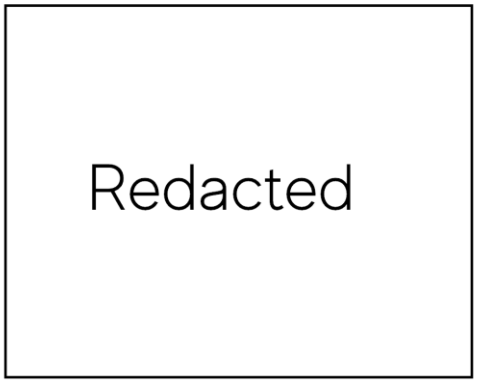


Figure 7: Notice Response Rules

As shown in Figure 8: Notice Priority Rule , staff can also determine the priority rules between notices so voter is not queued for more than one notice at a time.

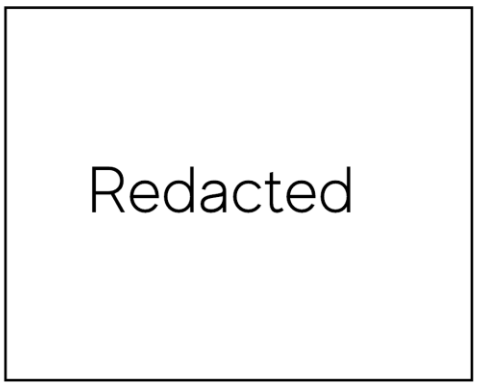


Figure 8: Notice Priority Rule

As shown in Figure 9: Notice Mail-Merge Fields below, notice service provides hundreds of data fields by domain (e.g., Voter, Election, Offices, Candidates) to be used as merge-fields in the mail-merge template. (CAR-8.3.1, CAR-8.3.3, CAR-8.7.2.1) The fields include rule-based address namely Effective Mailing Address. SVRS comes configured with rules to determine voter effective mailing address, ballot effective mailing address, and so on. ELECT can review and revise the rules to meet the needs. Here is an example of the address hierarchy used to determine Ballot Effective Mailing Address for Elections:

(1) When Election-Specific Address exists, use it
(2) If none, and when Ballot Mail Date falls within Temporary Address timeframe then use Temporary Address
(3) If none of above, then use Mailing Address
(4) If none of above, then use Residence Address

Notice Service under System Configuration in SVRS solution allows staff to configure between CSV, Tab limited, and JSON for notice. Additional file-types can be added as needed. (CAR-8.3.2.1).

As shown in Figure 9: Notice Mail-Merge Fields, staff configures the sort order of the notice such as Precinct, ZIP Code, Voter Name alpha by moving up and down until the order column meets the need. (CAR-8.1.11.1).

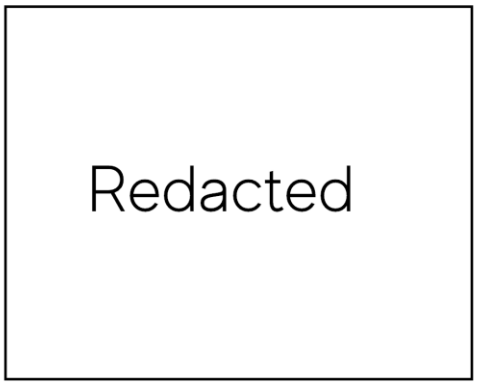


Figure 9: Notice Mail-Merge Fields

As shown in Figure 10: Notice Mail Merge Preview below, for mailed correspondence staff uses ****** Mail-Merge template to design notice layouts including postcard, labels, letters, and so on (CAR-8.1.13, CAR-8.7.2). Microsoft Mail-Merge is a proven method to design forms and smaller fragments with imported files and comfortably meets the requirement for 'what you see is what you get" (WYSIWYG) capability (CAR-8.1.1, CAR-8.6.1.1, CAR-8.6.1.2). Staff controls what should and should not be printed on the notices including determination of not to print date/timestamp and staff name on the actual correspondence by default (CAR-8.1.6). Staff can associate type of mail class and ancillary service endorsements into the mail template. (CAR-8.7.1). Solution supports inclusion of fields (e.g., Voter ID, Ballot ID) as barcode in different types including UPC, Code 39, and Code 128 (CAR-8.7.3). For the State of Virginia, Team Canton develops the support for Intelligent Mail Barcodes (IMb) for absentee ballots and notices in SVRS (CAR-8.7.3.2, CAR-8.7.3.3).

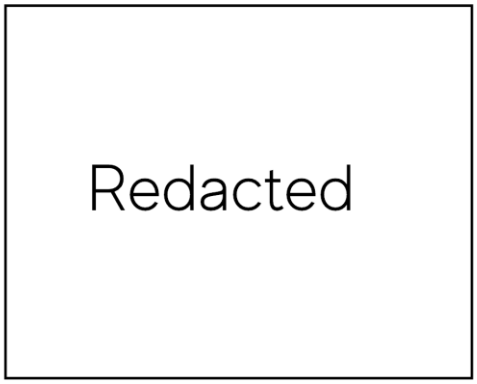


Figure 10: Notice Mail Merge Preview

Support of minority languages – As shown in Figure 11: Minority language support below, SVRS currently supports in California minority languages including Spanish, Chinese, Hindi, Japanese, Khmer, Korean, Tagalog, Thai, and Vietnamese. (CAR-8.2.1)

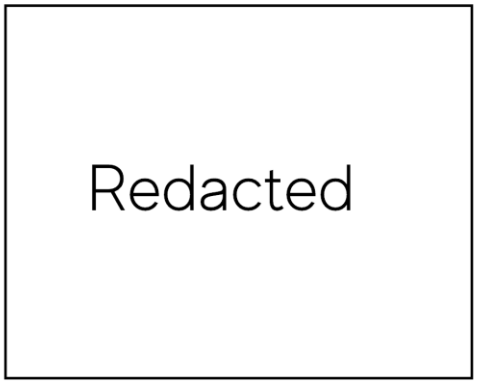


Figure 11: Minority language support

System Configuration Service in SVRS solution allows to configure minority languages at the locality and state level. ELECT can configure the languages supported at state-level, capture voter preferences on Online Voter Registration Portal, and send statewide mailing based on voter preference on file. At the locality level, during the election setup locality can select only those languages that are applicable to them to send election materials. Languages not supported within locality defaults to English. (CAR-8.2.2, CAR-8.6.4)

System Configuration service allows Staff to configure the translated texts for the standardized codes and descriptions such as Voter Status Change Reason Code, VBM Ballot Rejected Reason Code, and Provisional Ballot Reject Reason Code. The translation text is then dynamically used during the mail-merge and generation of notices including out of state move, cancellations, change of address, absentee rejections, and provisional ballot rejections. The translation text can be imported or exported in interoperable localization formats. (CAR-8.2.3.1, CAR-8.6.5)

During the notice template setup, Staff configures mail-merge templates in minority languages. SVRS uses voter's language preferences on file to select the correct notice template and the translated text for the standardized values to issue the notice. (CAR-8.2.3)

Team Canton proposes to add capability in Notice Service under System Configuration for Staff to indicate preference order between communication methods of mail, email, and text message (CAR-8.1.1, CAR-8.1.2, CAR-8.6.2, CAR-8.6.3). For the email and text message correspondences, currently the messages are designed by System Administrator. We propose to add 'what you see is what you get" (WYSIWYG) capability for Staff to design the message themselves similar to mailed correspondence. (CAR-8.1.1, CAR-8.6.1.1, CAR-8.6.1.2).

Issue Notice

SVRS provides staff to issue notices on-demand and in bulk. As shown Figure 12: Issue Notice On-Demand below, staff can navigate to the notice tab within voter detail screen, click on the print or preview icon. The preview icon shows the notice without issuing the notice (does not trigger a reprint). Also shown is the Notification History table are the logs for the notices send to the voter. The print icon displays the pop-up as shown in Figure 13: Notice Print Now or Later below, staff can select to print now or queue to be printed in a batch. (CAR-8.1.5).

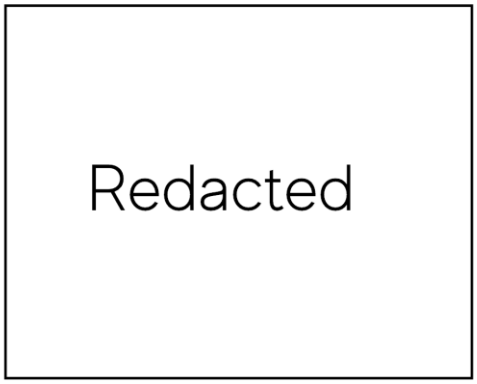


Figure 12: Issue Notice On-Demand

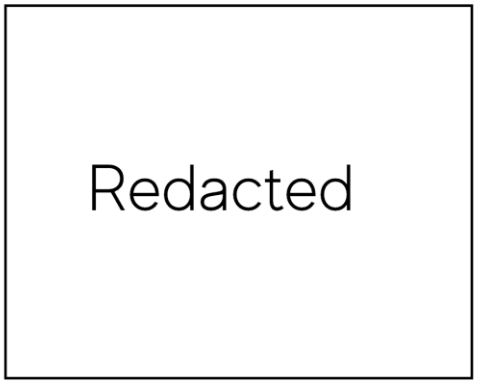


Figure 13: Notice Print Now or Later

As shown Figure 14: Send Notice Queue, the notices awaiting to be issued in bulk are available in the notice queue. (CAR-8.1.11).

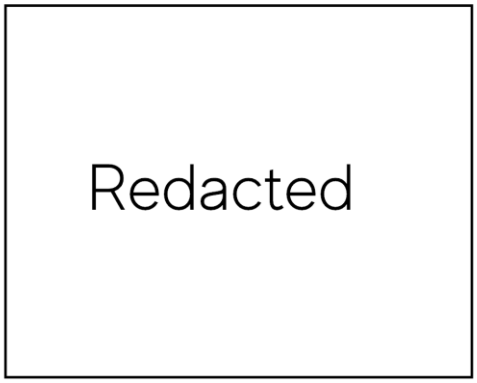


Figure 14: Send Notice Queue

As shown in above Figure 15: Notice Batch Criteria, staff can prioritize notice batches by preferences such as only international or out of state domestic only, and delivery method (CAR-8.1.11). Staff can use the eye icon to preview the notices (CAR-8.1.5). The notices can be extracted for third-party mailers or printed at the locality on the configured notice template in CSV, Tab Delimited, JSON, and PDF. Any other file formats for export can be implemented for the State of Virginia (CAR-8.1.12, CAR-8.6.6).

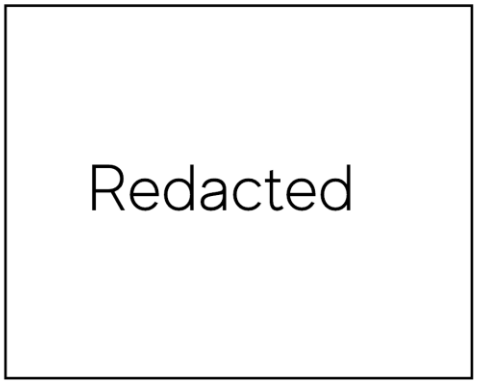


Figure 15: Notice Batch Criteria

When notice is issued, the transaction is captured in the Notice history log. As shown in the Figure 16: Notice History Log, the notice log currently displays sent date and sent by. For the State of Virginia, we can modify that to also include the delivery method (mail, email, text, etc.), resent date, and disposition. (CAR-8.1.3, CAR-8.1.9, CAR-8.1.10).

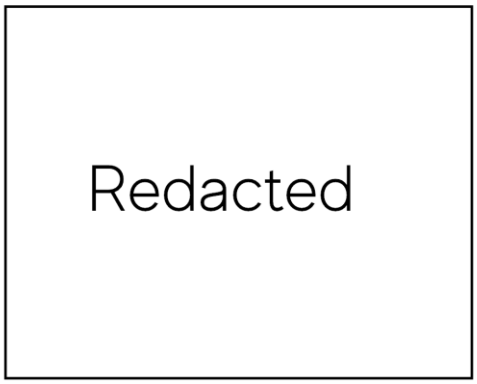


Figure 16: Notice History Log

Return Notice

When voter responses from voter are received, staff can upload it as part of voter documents as shown in Figure 17: Add Correspondence Image. (CAR-8.1.9).

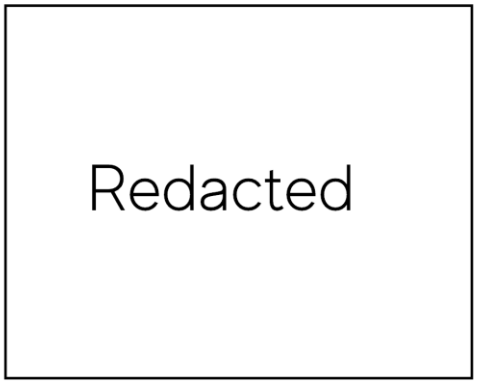


Figure 17: Add Correspondence Image

Any other correspondence such as call logs are captured as comments on voter record as shown below (CAR-8.1.9).

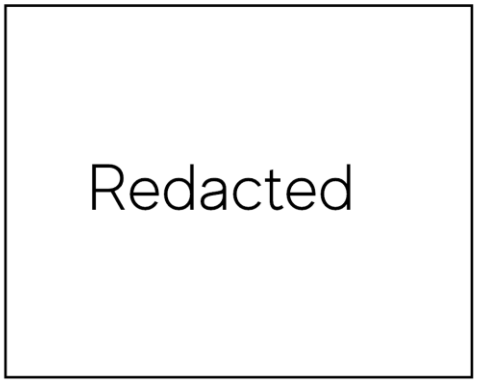


Figure 18: Voter Call Log Comments

Returned notice is further discussed under sections Process returned notice and Process undeliverable notice.

4.3.8.2 Election Analytics

SVRS Election Analytics includes Standard reports, Ad-hoc reports, and Advanced Analytics. Election Analytics provides a best in class, near-real time, ease of access to any information within the solution. Access to data can be secured at the column and row level including Personal Identifiable Information (CAR-8.4.8). SVRS is built on an industry leading underlying technology, ******, that is already familiar to ELECT (CAR-8.5.7).

Standard Reports

Generally, data that are print ready for publication and those used commonly for verification and validation falls under standard reports. Based on our experience working with other election offices, a good solution should be

dynamic to the changing needs where agency should be able to add new reports and remove once that are no longer used. SVRS, allows to you to manage those that are most relevant to your current needs.

Standard reports are grouped and categorized by domain such as voter, election, ballot, and so on. Staff has access to a list of pre-defined reports (CAR-8.4.1, CAR-8.4.7, CAR-8.4.11), filter by report name and description, and designate report as favorites (CAR-8.4.2, CAR-8.4.4). Staff can select the report, input the parameters such as registration date range, districts, precincts, elections, party preference, and so on (CAR-8.4.5), and execute the reports.

Reports are generated on the webpage however staff can save the report into various formats such as PDF, XLSX, CSV, PNG, and PowerPoint. Any other format currently not available, can be added as needed. (CAR-8.4.3, CAR-8.4.10).

Staff can use subscriptions to email snapshot of a report at scheduled times. As shown in Figure 19: Report Subscription Schedule, staff can specify a custom schedule to receive the reports. Staff can unsubscribe from an existing subscription, or make changes to a subscription's format, schedule, subject, or empty view mode. (CAR-8.4.14).

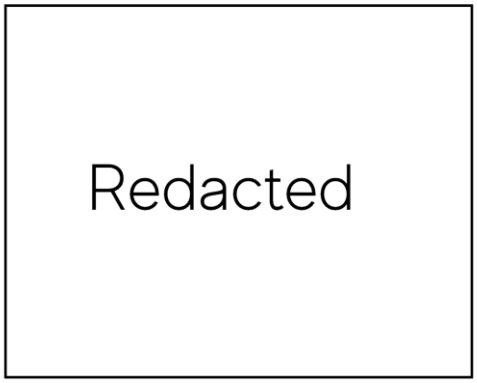


Figure 19: Report Subscription Schedule

Administrators have access to the usage metrics of reports on including who uses them, how many times used, what parameters are generally selected, and so on. This data can be used to improve user experience and clean-up unused reports. (CAR-8.4.2, CAR-8.4.6).

Data Validation Report

SVRS captures the data validation logs generated when processing voter registration data from external sources such as DMV, ERIC, etc. Staff can search for those records that have any data validation issues. The validation errors are shown on the voter record themselves for staff to resolve. For the State of Virginia, these data validation logs can be added to the Analytics Data Model for staff to create custom analysis and reports. (CAR-8.4.3.2)

Specialized Extracts

SVRS solution offers specialized extracts as described in this section, these include Public Voter Registration Data Request, Jury Wheel Extract, Report of Registration, Official Voter List, Voter Information Guide, and so on. As needed, these extracts support generation of supplemental data (includes only delta), scheduling, exclusion of confidential voters, and so on. These extracts are generated in CSV, Excel, PDF, and XML (CAR-8.4.12).

Client Services Extracts

SVRS solution offers Public Voter Registration Data Request (PVRDR) and Jury Wheel Extract (JWE) services, these two services combined satisfies the requirement for Client Services Extracts (CES).

PVRDR Service

SVRS PVRDR service is a robust solution to track, manage, extract, publish, and audit request for voter registration data from candidates, political parties, not-for-profits, and so on.

As shown in Figure 20: PVRDR Search, Staff can search for previous PVRDR request. They can view, edit, or copy. The copy option preserves and transfer requestor information and criteria selection from a previous request. This is great for most frequently received requests.

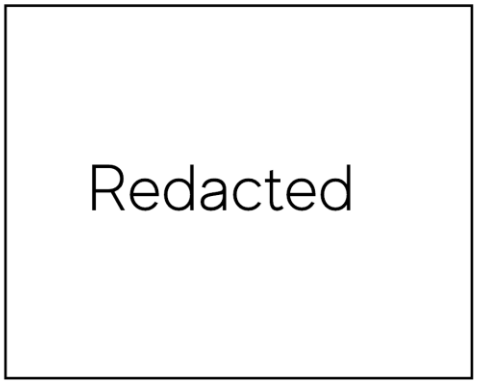


Figure 20: PVRDR Search

As shown in Figure 21: PVRDR Requestor Information, Staff can track who requested the data, to which organization they belong to, and the third-party information, and if the request was made on behalf of a client.

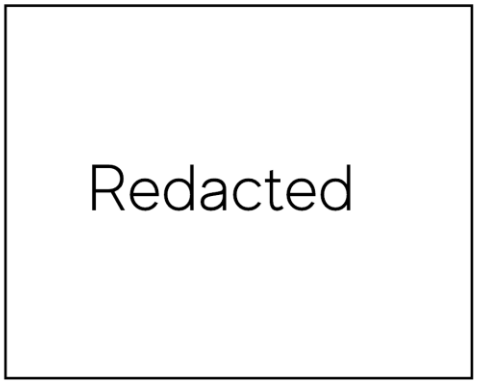


Figure 21: PVRDR Requestor Information

As shown in Figure 22: PVRDR Payment Information, Staff can track the payment details. Digital payments can be integrated with State of Virginia's desired payment gateway (CAR-8.4.12.2).

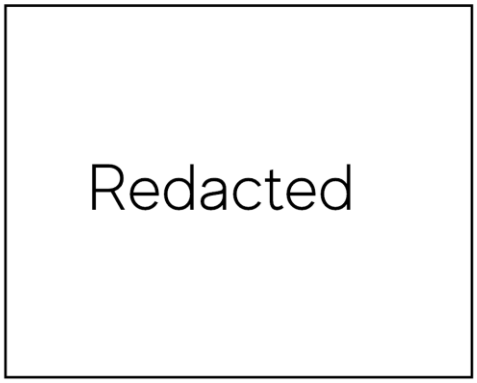


Figure 22: PVRDR Payment Information

As shown in Figure 23: PVRDR Shipping Information, Staff can capture the shipping details of the extract.

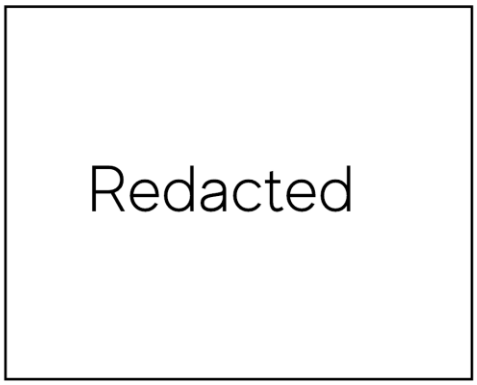


Figure 23: PVRDR Shipping Information

As shown in Figure 24: PVRDR Eligibility, Staff can capture the Eligibility of the request.

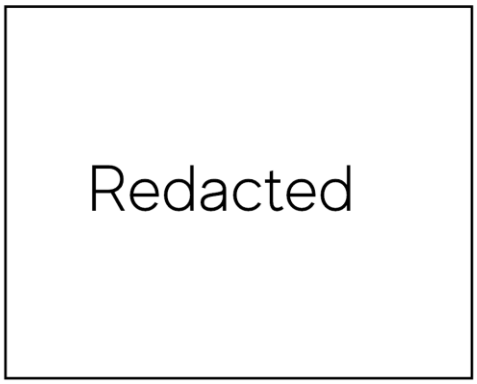


Figure 24: PVRDR Eligibility

As shown in Figure 25: Application Status and Dates, Staff can accept/reject request, track the dates for when applied, request received, and fulfilled.

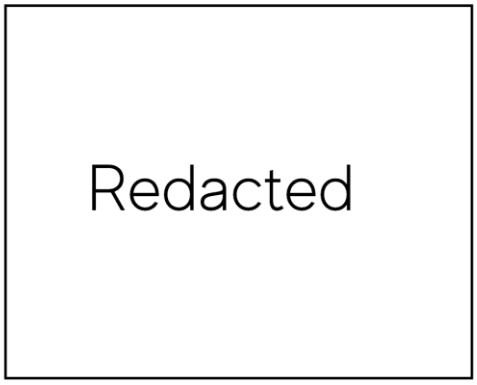


Figure 25: Application Status and Dates

As shown in Figure 26: PVRDR Voter Extract Criteria, Staff an input the voter extract criteria including date of Birth range, gender, language preferences, political party, registration date range, districts, precinct, and voter participation data for elections.

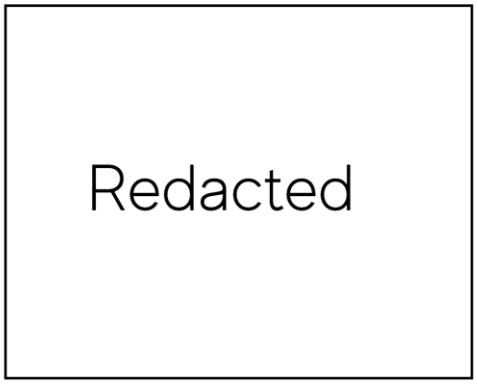


Figure 26: PVRDR Voter Extract Criteria

As shown in Figure 27: PVRDR Files , Staff can optionally include files for political districts map and voting history for selected elections. When the files are available, they can be directly downloaded to user selected location (CAR-8.4.12.2). Solution also tracks when and who downloaded the files.

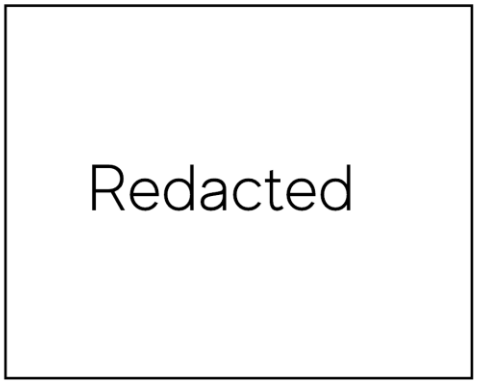


Figure 27: PVRDR Files

Jury Wheel Extract Service

SVRS JWE service is a robust solution to track, manage, extract, publish, and audit request for voter registration data from Courts.

As shown in Figure 28: Search JWE, staff can search for previous JWE request. They can view, edit, or copy. The copy option preserves and transfer requestor information and criteria selection from a previous request. This is great for most frequently received requests.

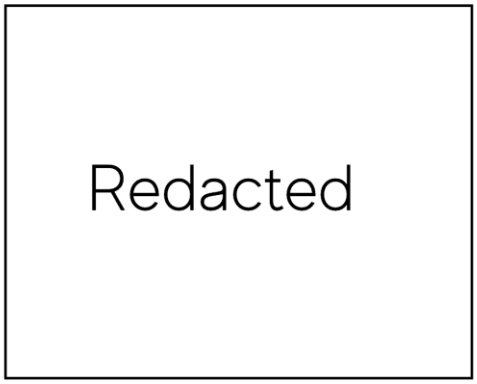


Figure 28: Search JWE

As shown in Figure 29: JWE Request Information, staff can track who requested the data and enter the voter extract criteria requested. Staff can input to select every n^{th} record starting from m^{th} record to produce a randomized list.

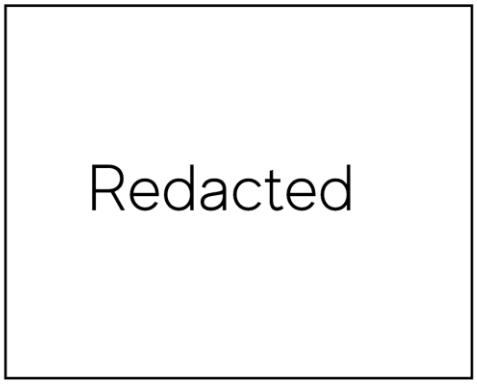


Figure 29: JWE Request Information

As shown in Figure 30: JWE Files, when the files are available, they can be directly downloaded to user selected location (CAR-8.4.12.2). Solution also tracks when and who downloaded the files.

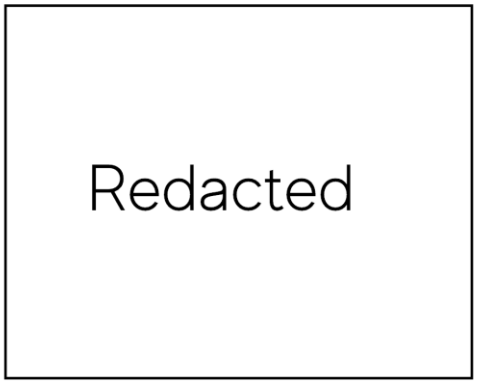


Figure 30: JWE Files

Elections Assistance Commission (EAC) Reporting

When it comes time to responding to EAVS survey, ELECT can be self-sufficient and run out-of-box extracts that draws on necessary data to fill out the EAVS section A-F (CAR-8.5.1, CAR-8.5.3). The required data that is summarized includes voter registration status reason (e.g., USPS notice, confirmed per response to notice), type of registration (e.g., regular, pre-registration, UOCAVA), method and source of voter registration, type of ballots (e.g., regular vote-by-mail, in-person, UOCAVA vote-by-mail), ballot disposition, and so on. (CAR-8.5.4, VR-1.2.1.19, LM-9.5.3). Advanced Analytics service allows ELECT to enhance the data captured within SVRS to keep up with changes to EAVS survey (CAR-8.5.2). Team Canton can advise the State of Virginia on the best practices and rubrics to address data discrepancies that come up due to the constant movement of data from registration activity.

While the data is already available within SVRS solution, for the State of Virginia Team Canton proposes to build following features to enhance the coordination between ELECT and localities and track historical reporting –

a) Screens for localities to input answers for survey questions for which data is not available within voter and election data. Screens for localities to indicate readiness for the ELECT to generate the EAVS reports on their behalf. (CAR-8.5.6)
b) Option for ELECT and localities to view previously completed EAVS surveys. (CAR-8.5.5)

Out-of-State Move-in Report

When voter registration data received contains previous registration information, SVRS compares the previous residence address to determine if voters are moving in from another State. These voters are flagged to be included in the Out-of-State move-in report. This report is scheduled to be sent to the report subscriber. (CAR-8.4.9)

Electronic Registration Information Center (ERIC)

The Team Canton has experience working with ERIC format in Washington, D.C. and Maryland. For the State of Virginia, Team Canton configures the data pipelines via Partner Data Integration Service to periodically export encrypted voter records to the ERIC sFTP site.

Ad-hoc Query

Every module in SVRS such as Voter, Election, Candidates, Ballots, and so on provides robust search for staff to create ad-hoc query. Staff is able to dynamically drag and drops the output results column including unstructured data such as signature images (CAR-8.4.13) to produce the extract. The ad-hoc query capabilities in-build within each module is explained within their respective sections for Voter Registration, Election Administration, Absentee Voting, and so on.

Advanced Analytics

SVRS is the only VREMS solution in the market that offers advanced analytics out of the box. ELECT and localities can connect to our published data sources to power analytics, create stunning data visualizations, and share those insights with others. The published data sources use simple easy to understand business terminology for user to drag and drop fields into reports. Report authors can seamlessly connect to the web-based report authoring tool using the single login credential (SSO), based on their privileges access to data is securely served.

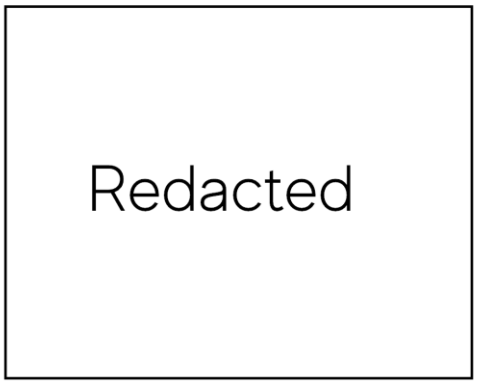


Figure 31: Report Authoring

1. Workbook name. A workbook contains sheets. A sheet can be a worksheet, a dashboard, or a story.
2. Pages shelf, Filters shelf, Marks card - Drag fields to the cards and shelves in the workspace to add data to your view.
3. Columns and Rows shelf - Drag fields to the cards and shelves in the workspace to add data to your view.
4. Toolbar - Use the toolbar to access commands and analysis and navigation tools.
5. View - This is the workspace where you create your data visualizations.
6. Side Bar - In a worksheet, the side bar area contains the Data pane and the Analytics pane.
7. Go to the data source page.
8. Sheet tabs - Tabs represent each sheet in your workbook. This can include worksheets, dashboards, and stories.

4.3.9 List Maintenance

4.3.9.1 LM-9.1: Manage Data Resources

SVRS supports Master Data Management. We implement data matching using our own home-grown state-of-the-art ****** Service which is built using Machine Learning and Data Science. We don’t perform expensive relational queries, but created our own models or algorithms for matching. Our AI model also supports explain-ability properties to inform what were the features that contributed more to the matchings. This service receives data in a canonical format from our Partner Data Integration Service and together with the records in the system database proceeds to transform the features (matching variables) into a vectorized representation for

the matching data and the database records. Then performs a search optimization process by reducing the search space, and finally executes a similarity to find the matching results as shown in the image below. search.

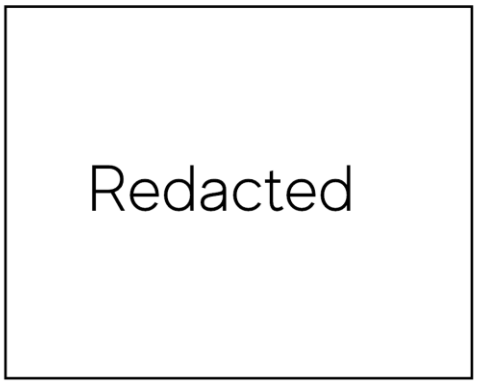


Figure 32: People Identification Service

This process basically requires the identification and integration of master data across sources. (LM-9.1.1). The intel Match Service, as explained above has intelligence built in to perform matchings between entities across disparate resources (files, data bases, etc.) (LM-9.1.2). The capability to explain by quantitative telling what features influenced more in the matching decisions meets requirement (LM-9.1.3). The level of confidence is intrinsic to the algorithm process, the vectorization and similarity search, to find the “k” best matches. The initial weights of the process are determined, but don’t have a huge effect in the final decision, since these are being re-calculated internally by the model. By making this process almost independent of human weights inputs, we assure a more optimized and accurate matching process that only depends on the data of the system. We meet or exceed expectations on these requirements (LM-9.1.4.1), (LM-9.1.4.2). We plan to add matching groups model in future releases of the intel Match Agent. (LM-9.1.5) SVRS supports creating or merging a master record from data elements collected from validated and authorized data sources. Survivorship rules (merging rules) are defined and implemented during implementation time (LM-9.1.6).

Team Canton will work with ELECT to identify a data collection of name alias that can be imported into SVRS. The intel Match service is then configured to include the name alias as part of match criteria. (LM-9.1.7).

Real time identification is supported in the application views. The data to be identified is entered into the system, and a REST API is called to send the data to the Partner Data Integration Service, which in turn sends the data in canonical format to the intel Matching Agent. The results are returned to the client via call backs. (LM-9.1.8)

All records have a source of origin in the system. Some of them will be from data conversion, others entered via the application UI, others from external partners. We record the origin of the data in the data lake from the replicated database, or from the Partner Data Integration service. This metadata is kept in a data repository in

our data lake and any transformations to the data is saved to the repository, in this way any user can trace the records lineage (LM-1.9.9).

If there is a need to load and integrate data from additional data sources into the system without requiring code changes, this means that this is a pass-thru data and no validations, transformations are needed and the underlying logic to handle this data is already built in the system; so yes, we support this model. Otherwise, the data will need structural and data format validations extra code, as well as mappings and logic to handle the data. (LM-9.1.10)

With respect to sharing voter registration data with authorized entities; the application provides rules for users to have this privilege. Also, the system can use the Partner Integration Service to share (send) data to other authorized entities via the outbound interface (LM-9.1.11).

The Team Canton has experience working with ERIC format in Washington, D.C. and Maryland. For the State of Virginia, Team Canton configures the data pipelines via Partner Data Integration Service to periodically export encrypted voter records to the ERIC sFTP site. (LM-9.1.11.1)

For information related to integration with Jury Wheel Extract, please refer to section 0 **Jury Wheel Extract Service**. (LM-9.1.11.2)

4.3.9.2 LM-9.2: Enterprise Data Integration

4.3.10 As general reference on how our Enterprise Application Integration Service works (Partner Integration Service), please refer to 4.3.13. General Administration

4.3.10.1 GA-12.1 Data and Audit Records

GA-12.1.1 Metadata, Audit and Master Records

The system supports storage of data within the system. We use a relational database and blob storage to persist transactional data, audit records, documents, images, files, etc. (**GA-12.1.1.1**). The audit records are stored in the database or file storage, secured by security features and policies. The audit records can be purged following retention policies (**GA-12.1.1.2**). Any interaction with transactional data is recorded using triggers in the database or logs in the request and data layer recording contextual information such as the user, date timestamp, operation type and state of the data before the operation. For group of changes and distributed transactions we have a correlation id to group the changes associated to a specific set of transactions. (**GA-12.1.1.3**), (**GA-12.1.1.4**), (**GA-12.1.1.7**), (**GA-12.1.1.9**). In the next release we will have the ability to capture click counts per user for user processing time stats (**GA-12.1.1.6**).

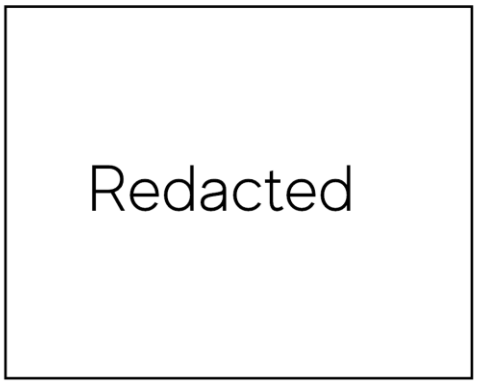
The component where the operations are performed are intrinsically recorded by the audit records (**GA-12.1.1.5**). Audit records of user accounts are stored in journal tables associated to the user related tables. User

request reports can be generated from these records in the Reporting/Data Analytics module. (**GA-12.1.1.8**). The audit records cover PII information. Audit records are only visible to administrators of the system. We provide tools to monitor and visualize continuously the logs and audit records. We can also filter the logs to hide PII information. (**GA-12.1.1.10**), (**GA-12.1.1.11**). The solution uses the Cloud (Azure) Network Time to synchronize all the times across the resources of the solution. (**GA-12.1.1.12**). Our ****** Agent Service support management of master data for voters, felons, deceased, adjudicated incapacitated, and non-citizens. Voter data is dynamically linked to the incoming external agency data to provide an up-to-date voter registration data. (**GA-12.1.1.13**).

GA-12.1.2 Documents (Enterprise Content Management)

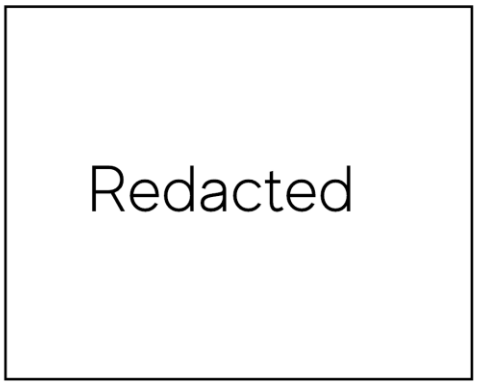
Documents are persisted as files under their original format, but whenever a document needs to be persisted as an image, we use standard image formats. (**GA-12.1.2.2**). We support Optical Character Recognition (OCR) by using industry leading UiPath Automation. This feature allows to extract plain text from the images (**GA-12.1.2.3**). The solution supports entering special characters in the input field of the Voter Registration windows. This allows to capture nonstandard ascii code (**GA-12.1.2.4**). During conversion, we should be able to digitalize data from registration cards; this is a manual or semi-manual process. The solution provides capability to store correspondence records in PDF format.

SVRS supports the attachment of documents in Voter (**GA-12.1.2.5**), Candidate, Locations. (**GA-12.1.2.1**). SVRS allows common image formats and PDFs to be attached (**GA-12.1.2.7**). Timestamps for upload date are added to all attachments (**GA-12.1.2.8**)



Voter image tab

Correspondences are issued (**GA-12.1.2.8**) and tracked on the Notices tab of voter (**GA-12.1.2.6**), but all images/attachments are added on the Images tab.



Voter Notices tab

GA-12.1.3 IT Lifecycle Management

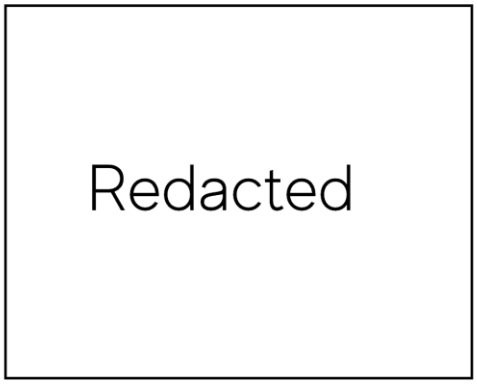
The design of SVRS includes support for retention of data. This is a procedure that must be defined during implementation of the solution. We can easily add functionality in the administration module to provide scheduled of data retention as a configuration parameter. Currently the data that passed their retention schedule is purged using ad hoc scripts. (**GA-12.1.3.1**), (**GA-12.1.3.1.1**), (**GA-12.1.3.1.2**). SVRS architecture implements the Reactive Architecture Principles: The solution is responsive, scalable, resilient, and not monolithic.

Responsive: We assure that the architecture and infrastructure allow for timely responses to request independent of the number of users or load size.

Scalable: SVRS architecture supports horizontal scalability of the micro services based on the container size and capacity. When a microservice is reaching its capacity, our platform creates a new microservice to support the excess of load.it knows how to route the requests to the microservices automatically.

Resilient: When a microservices health is deteriorating, the platform fires a new microservice to replace the failing one. This allows our system to be 24X7 up and running.

Message Driven: Each microservice maps a functional domain and only knows how to do that business logic. It does not persist state (stateless), so the API requests from the clients do not interfere with each other. The microservices therefore are decoupled services communicating via messaging to achieve workflows.



The above high-level description of our architecture assures that SVRS is performant, according to the SLAs, regardless of the amount of data entered in the system (**GA-12.1.3.1.3**).

GA-12.1.4 Code Tables

SVRS data model includes code tables. The access to the code tables is via a specialized GUI or using direct SQL. Only a user (administrator) with the right permission can maintain (create, Update, Inactivate/expire) the code tables. Expired codes are filtered out for their use in the application. The code tables are structured having in mind the jurisdiction hierarchy (state, county). Some codes apply to the entire hierarchy while others are subject to be unique at their level. This overwrite capability is supported by SVRS. (**GA-12.1.4.1**), (**GA-12.1.4.2**), (**GA-12.1.4.2.1**).

4.3.10.2 GA 12.2 Usability

GA 12.2.1 Functionality

SVRS client runs on a web browser. A user can have multiple sessions of the application in the same computer on multiple web browsers running concurrently. SVRS stores client preferences when the user initiates a session. The solution persists the user location (IP address, time stamp on the session initiation, user level (roles)). A privileged user (administrator) can lock access to functions of the application to lower-level users via the user management and user roles management views. (**GA-12.2.1.1**), (**GA-12.2.1.2**), (**GA-12.2.1.2.1**).

GA-12.2.2 User Interface

In general, our design is a Domain Driven Design (DDD), where we decompose the elections domains in sub-domains, preferably independent of each other. Specifically, our User Interface design includes the use of user-centered design best practices. We follow an Agile Process where the product owner, a person with extensive knowledge in the election’s arena, is a member of the design team. We also included feedback from actual users of a County in California. We used simple archetypes (personas) to represent election users. The Ideation process includes use of visual aid designs and workflows (mockups). The information architecture is defined in our DDD process to support the Object Model that implements the business logic. Our View Models

represent what the user needs in their workflows and maps the aggregates (object model) in the business layer. This mapping validates the structure and data being used in the UI. (**GA-12.2.2.1**). SVRS implements the Object Action Pattern, in which the user selects first the object (voter) and then the action to perform on it (create, update, etc.) (**GA-12.2.2.2**) (**GA-12.2.2.3**), (**GA-12.2.2.4**), (**GA-12.2.2.5**)

SVRS limits the access of functionality and data to the locality of the user. The application uses the properties of multi-tenancy with locality identifier (jurisdiction id) to segregate data access to the appropriate locality (**GA-12.2.2.6**)

GA-12.2.3 Search and Filter

SVRS provides with powerful searching and filtering capabilities. We offer a permanent search feature in the UI. A user can only search based on assigned privileges. User's roles are designed to allow users to search only in their localities (jurisdictions) or cross jurisdictions. A user can search statewide (unified search) or in multiple localities if they have the appropriate privileges. Searching is around functional criteria. SVRS offers searching with single or combined criteria. System attributes such as primary key, are not included in SVRS because these values have only meaning internally to preserve data integrity or facilitate searches. Allowing users to search by primary can lead to users accessing data that is not allowed to access, such as data from other jurisdictions or simply PII or confidential information. But, adding search by primary key would be very simple, because of the expandability of SVRS. (**GA-12.2.3.1**), (**GA-12.2.3.2**), (**GA-12.2.3.3**) (**GA-12.2.3.3.4**).

GA-12.2.4 Redaction, sub-setting, and substitution

ELECT to define set of data that is considered Personal Identifiable Information (PII), the SVRS solution encrypts the data and is only available on UI and reports for users with permission to view PII data otherwise it is masked (redacted). The level of masking can be setup at field level such as to show **** for SSN4 vs **/**/1987 for date of birth. (**GA-12.2.4.1, GA-12.2.4.2**). In case of protected (confidential) voter, the mailing address is required in addition to residential address, however mailing address is always used in the reports and labels. (**GA-12.2.4.3**). SVRS preserves PII information obscured or masked in all the environments. (**GA-12.2.4.4**).

4.3.10.3 GA-12.3 Process Management

GA-12.3.1 Transactions

No, we do not add comments at the transaction level. SVRS records the transactions and meta data about the transactions for audit purposes. By looking at our historical data you can determine what changed (**GA-12.3.1.1**).

GA-12.3.2 Election Calendar

SVRS has its own calendar support to describe calendar events and occurrences, such as pre & post milestones. This calendar is inherited from our UI framework in the concrete Windows (UI pages) (**GA-12.3.2.1**)

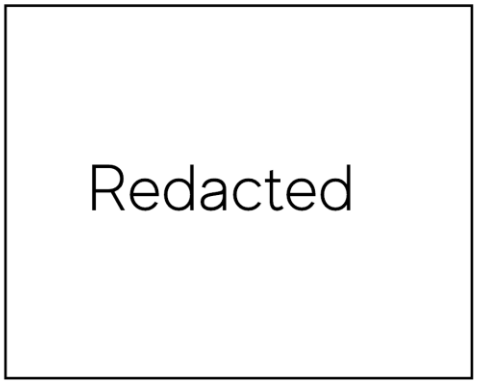
GA-12.3.3 Workflow & Work Allocation

To understand how SVRS effectively manages large number of voter and ballot workloads, please see sections 4.3.1 Voter Registration and section 4.3.5 Absentee Voting for more information (**GA-12.3.3.1**) (**GA-12.3.3.2**)

One user assigning workload to specific users is not something currently in-built; however, pushing work into workable queues for processing is very much part of the SVRS solution. (**GA-12.3.3.3**)

SVRS is built on the premise of “data in, data out”, as a result all primary screens in SVRS start either with an Advanced Search or a Queue card (**GA-12.3.3.4**)

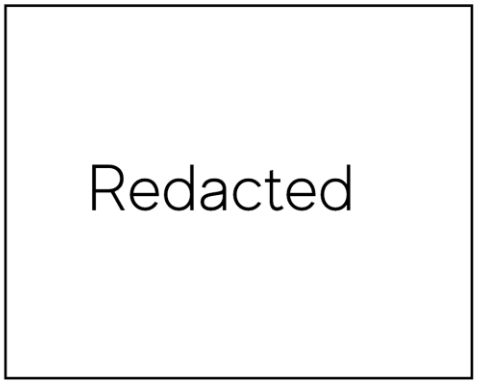
The Election Calendar is a System Configuration, and unique calendars can be created for each election type. This automatically calculates all key election dates during election creation. This acts as a template, and does not restrict modifications in the event a date needs to be modified for a unique circumstance. (**GA-12.3.3.5**)



SVRS System Configuration: Add Election Calendar details

While alerts can be built-in, SVRS comes with a homepage workspace which is intended to take the place of alerts or many reminder e-mails. It strives to answer the question “What should I do today?” or “Where is my jurisdiction at?”

At present, there is one default workspace. There are five homepage workspaces in active development: Odd Year Cleanup, Election Voters, Election Ballots, Election Event (Workers), and Administrative. (**GA-12.3.3.6**)



SVRS Homepage Workspace (Default)

4.3.10.4 GA-12.4 Relationship Management

GA-12.4.1 Authority

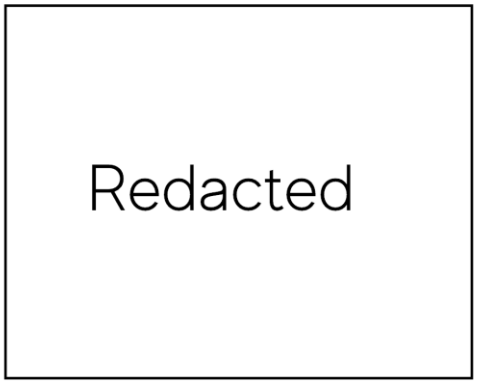
SVRS allows information updates to users with the appropriate privileges based on the user's locality (**GA-12.4.1.1**).

GA-12.4.2 Position holder Contact

For information on how SVRS handles Positions and Position Holders please see 4.3.2 Election Admin, EA-2.2 Manage Offices (**GA-12.4.2.1**) (**GA-12.4.2.2**) (**GA-12.4.2.3**) (**GA-12.4.2.3.1**) (**GA-12.4.2.3.2**) (**GA-12.4.2.3.3**) (**GA-12.4.2.4**) (**GA-12.4.2.4.1**) (**GA-12.4.2.4.2**) (**GA-12.4.2.5.2**). Additionally, on the Voter Record a user can select the "Election Officials" button to view the list of position holders for a particular voter.

In regards to importing, the initial data migration would include the setup of historical and current position holders; however, maintaining this is done via Election or manual maintenance using the Add/Edit feature (**GA-12.4.2.5**). SVRS does not currently create a "vCard" file format, however does maintain Position Holder contact information, including social media sites (**GA-12.4.2.5.1**). SVRS allows a search and extract data that can be used to print labels or report member information, depending on the business needs more specific standardized reports or correspondences can be created easily (**GA-12.4.2.6**) (**GA-12.4.2.7**) (**GA-12.4.2.9**).

Positions have the option to be "deleted" in SVRS; however, this is really an archive process which removes the position from the front-end UI and reporting. (**GA-12.4.2.10**) Current position holders become "Former Representatives" when their term End Date is populated—this can happen due to loss of an election, retirement, or other means of vacating a position. Reason Codes can be added to express if it was due to loss of election, retirement, death, withdrawal, recall, etc. (**GA-12.4.2.11**)

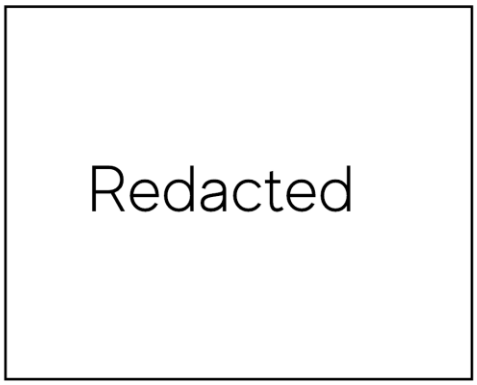


Position Holder details – Current Position holder tab, Former Representatives tab

Candidate allows for image capture; this can be expanded into Position holder—but is not currently a feature of that module; image storage options are expandable into Positions (**GA-12.4.2.8**)

GA-12.4.3 Office Contact Information

In the Jurisdiction details section of System Configuration, users can enter the jurisdiction contact and operating hours. (**GA-12.4.3.1**) (**GA-12.4.3.1.1**) (**GA-12.4.3.7**) (**GA-12.4.3.8**)



Jurisidiction Details (please note, the VoteCal tab would not be visible in VA implementation)

The ability to define contact information for multiple, year-round satellite locations outside of an election event is not currently in SVRS. However, users would be able to use SVRS if the staff and site are provided access via user management and/or Single Sign-On (SSO) integration.

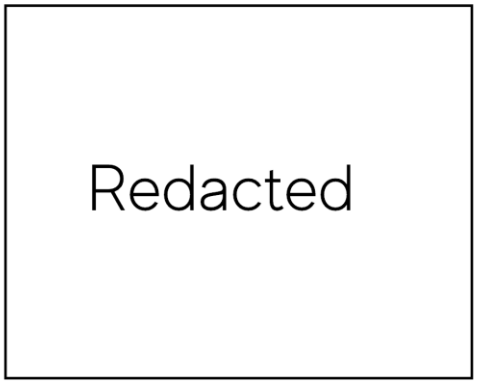
Another variant of a satellite office is a sub-jurisdictional user, who only have access to a particular district within a locality, can be created in SVRS. This allows these users the ability to manage an election and generate poll lists, issue and manage ballots for the assigned district—but keep other parts of the system either blocked or read-only. This is more common in larger jurisdictions for city district special elections (**GA-12.4.3.2**). User information can be extracted from the db (**GA-12.4.3.3**).

Importing would be done during implementation and maintenance would be via SSO or manual user interaction (**GA-12.4.3.4**). Integration with Active Directory (AD) would be a type of import as well.

Election locations can provide services to voters; full or partial services. For more information on Locations in SVRS, see please see 4.3.3 Establish Voting Locations (**GA-12.4.3.5**).

The ability to define contact information for other states is on the SVRS roadmap, this is a System Configuration development item, and is planned to be implemented for the use of contacting other states for any reason that is deemed appropriate for the State of Virginia (**GA-12.4.3.6**)

Labels and envelopes can be printed in SVRS, the specific-business need may need to be configured for State of Virginia (**GA-12.4.3.9**).



System Configuration – User Manager – Add User with Sub-Jurisdictional User Selected

4.3.10.5 GA-12.5 Maintenance, Documentation and Training

GA-12.5.1 Training

SVRS supports the transparent creation of training environments. We create environments in the cloud using scripts. To this automated process we follow with a process of tightening the platform for security using security guidelines/check lists and vulnerability reports. Our Continuous Integration Continuous Deployment (CI/CD) process which includes the creation of Docker images and together with our microservice architecture deployed in Docker Containers can automatically release the application (images and other artifacts) to the newly created training environment by just modifying the configuration of our CI/CD pipelines to point to the proper environment (**GA-12.5.1.1**). The data for the training environment is created using day zero scripts where we configure the training jurisdiction(s), code tables, etc. This is a separate process from the creation of the environment per se. When upgrading training environments, we do not need to refresh the data from the production environment. (**GA-12.5.1.2**).

GA-12.5.2 Documentation

SVRS provides documentation using multiple channels, being the most common printing with options for PDF or actual printing to paper. (**GA-12.5.2.1**) The current state of the system is in the transactional database. This is the single source of information for the distribution channels. The state of the system is replicated in our Data Lake solution in near real time for data analytics and some reporting purposes. (**GA-12.5.2.2**) Our agile Process principles value the product over documentation. But we also understand that we need to comply with the client requirements. We use Microsoft DevOps to manage our requirements in the form of Epics, Features, User Stories and Tasks. Here we also have a place to attach technical and functional designs to implement the features. We use the DevOps Repository to manage our source code. The DevOps pipelines for building and deploying the application are a set of scripts that help automate this process. Beside this we have developed a set of technical and functional documents such as Software Design Document, Application Design Document, Security Test Documents, Technical Infrastructure Design, Step by Step guidelines for the building and deploying environments, Change Control for Releasing (CCR) documents, etc. (**GA-12.5.2.3**)

4.3.10.6 GA-12.6 System

GA-12.6.1 System Design

Our data migration process includes retention of historical data from VERIS and SVRS. This data can be available to users using query tools to view them in their native format. Since we are converting this data to our Data Model, we also provide access to the converted data via our data analytics service or via the application for viewing the current state of the system. (**GA-12.6.1.1**) SVRS Architecture from the design implements configurability as much as possible. We can configure the application functionality, user permissions, code tables data. With this feature we can modify the application behavior without making code changes. Of course, there will be times where a new functionality or a modification to process flows are needed that will require code changes. In this case we follow Agile process to schedule and deliver these modifications incrementally without affecting the application. Only the affected microservice (modules) will be re-deployed in real time. We do not need to re-deploy the entire application just because one module changed. We just deploy the affected components. (**GA-12.6.1.2**)

GA-12.6.2 System Alerts

SVRS has standard communication channels. One is the application messages displayed on the user screen letting the user know what the action is to perform, or information about their transactions. The other channel of notifications is for system notifications. Out Infrastructure architecture includes a standard notification via email of any security risks, or performance of the application. Users with the role of administrators or appropriate privileges receive these notifications so they can react to them in a timely manner to avoid issues such as performance degradation, attacks, etc. We also offer a real time monitoring visualization windows help in determining when and what microservice is not performing as expected. (**GA-12.6.2.1**)

External Interfaces.

Our Partner Integration Service supports data ingestion and integration from multiple agencies and information sources. We support multiple data communication protocols and data formats in our end points for data ingress and data egress. (LM-9.2.1). Our External Partner Integration supports data ingress in bulk using standard data interchange formats such as XML, JSON, CSV as mentioned in section 4.3.14 (LM-9.2.1.1). SVRS supports ingestion of new data via insert statements in the database. It also supports key-based data ingestions, using reference fields as in update_at_columnname. (LM-9.2.1.2). Data deduplication is supported by our system using key matching features thru our intel Matching Service (LM-9.2.3).

4.3.10.7 Eligibility Prequalification

SVRS Solution's Partner Data Interface Service supports API and File based integration to ingest external agency data, such as DMV, VSP, and Courts (LM-9.3.1). ****** Services maintains a master record of prohibited list data used to check for voter ineligibility, such as Felon, Adjudicated Incapacitated, Deceased, and non-citizens. Staff can search and manage the master record of prohibited list data including –

- Add or correct record such as felon, deceased, adjudicated incapacitated (LM-9.3.2)
- Mark a felon or adjudicated record as restored (LM-9.3.3, LM-9.3.4)

When new voter registration is received, SVRS automatically completes in near-real time the eligibility check against the prohibited list data including –

- Deceased Check
- Felon Check
- Adjudicated Incapacitated Check
- Non-Citizenship Check

When a high-confidence match is found with prohibited list, the voter record is automatically cancelled with the reason. The source of data used for the determination is associated to the voter record for audit.

4.3.10.8 Manage Duplicates

Duplicate **** Service Agent**

SVRS solution offers duplicate ****** Service Agent which is an accurate, fast, and reliable matching solution driven by machine learning models (LM-9.4.1). SVRS overcomes the drawbacks of conventional matching process that includes:

- Based on a best guess, agency must define match criteria and arbitrarily assign confidence level (e.g., first full last name, first 5 characters of first name, DOB, OR DMV number & last 4 digits of SSN has a confidence of 95% match)
- Due to performance consideration, agency must limit the number of match criteria, match basis (e.g., Exact Match, First 5 characters, Soundex), and number of data fields (maybe 4-5 fields)

SVRS' ****** Service Agent uses power of mathematics and machine learning to curate high performance matching models which will:

- Intelligently navigate around typos, Soundex, "Smartnames", partial data and so on. This is particularly important when receiving voter registration data from external agency.
- Agency only needs to define these two categories of inputs:
    - Match fields - The fields that can be considered for match (e.g., Last Name, First Name, DOB, OR DMV number, last 4 digits of SSN, and so on)
    - High and Low confidence match thresholds - Any match that returns a score above the high confidence threshold is considered a valid match. Matches with score between high and low confidence threshold is considered a potential match, needs manual review. ELECT can define the confidence threshold to manage automated vs manual resolution of duplicates (LM-9.4.1.2).
    - Agency can crawl, walk, and run towards automation. Agency can conservatively start out with the high confidence threshold at 99%. Based on real-world performance, the threshold can be lowered. As more duplicate matches are automatically processed, the Staff is freed to focus on other important activities.

Duplicate ****** Service Agent swiftly returns 'matching pairs', and 'possible matches' for incoming voter registration record based on thresholds. Agent can be invoked in real-time from anywhere within the application such as while processing electronic or paper voter registration. Agent can be initiated for just one voter, one county, or an entire state.

SVRS solution can be modified to import and consolidate duplicate matches identified from external system with those identified from duplicate ****** Service Agent. Team Canton recommends the duplicate matches from external sources minimally follow these rules: uses VoterIds as it exists in SVRS; and includes the basis of each match, confidence level, and match identified date. (LM-9.4.1.1)

High Confidence Duplicate Match

SVRS' workflow engine supports orchestration of work-flow path to automatically process a high-confidence Duplicate match. Here is an example of workflow for real-time duplicate voter-voter matches:

1. Staff initiates to process a voter registration using 'Add Voter' screen.
2. Staff inputs Last Name, First Name, DOB, OR DMV number, and last 4 digits of SSN.
3. Duplicate ****** Service Agent displays a single high confidence duplicate voter from another county.
4. Staff confirms to overwrite and save the existing voter record with new registration information.
5. SVRS moves the record to new county and the old county is sent a Voter Moved notification. The old county voter record is cancelled.
6. SVRS links the voter and election history from existing record is to the new voter. This history includes previous registration, images, list maintenance, election participation, ballots, candidate, and so on. The original state registration is preserved.

Possible Duplicate Match

When locality Staff logs into SVRS, on the List Maintenance dashboard they will find the total counts of pending duplicate matches, as shown in Figure 33: List Maintenance Queue below.

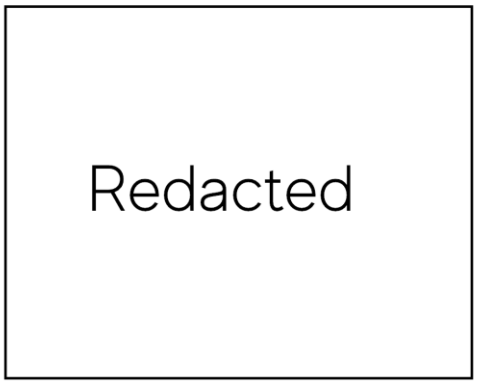


Figure 33: List Maintenance Queue

Staff navigates to the Process Duplicate queue and begins to work them on a FIFO method. Staff can concurrently work the queue without interfering in each other's records. As shown in Figure 34: Duplicate Voter Matches below, Staff can search and sort through the queue by Received dates, Match Status, Voter Name, or Voter ID. If already processed, they can search by Processed Dates, Processed By, and if there were any errors in the process. Staff has the option to export the list. (LM-9.4.1.2.1)

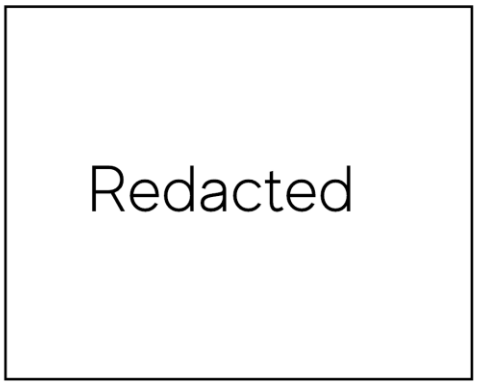


Figure 34: Duplicate Voter Matches

As shown in Figure 35: Process Duplicate Match below, the primary voter record and all matched voter records are available to be processed at the same time. Staff can compare voter registration detail, images, election participation, and activity history to review the information, compare, and quickly determine if the validity of the match as valid or invalid. The voter that is a confirmed match is automatically merged. Matches that are determined as invalid matches, the matching pair is removed from the work queue. Staff can remain in the work queue, process next, skip a record, or go to previous without having to navigate back to another screen.

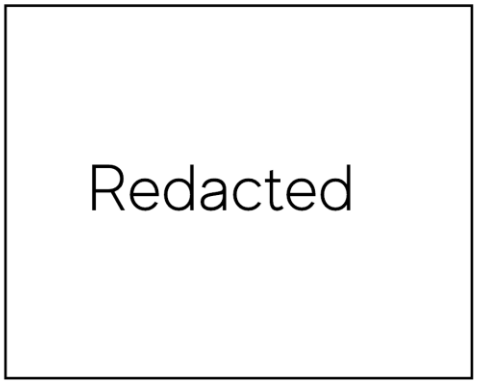


Figure 35: Process Duplicate Match

Voter intel-Merge Service Agent

SVRS' intel-Merge Service Agent uses Master Data Management patterns to match and merge duplicates while preserving the data lineage (LM-9.4.1.3). The drawbacks of conventional merge process include:

- Merge is limited, at best it copies specific historical voter data such as voter participation, images, and so on from non-survivor to survivor record. It is expensive to add new data categories as it involves implementing it for historical merges.
- Unable to undo a merge or un-merge is complex.
- Both merge and unmerge creates duplicate and/or orphan records.
- With the loss of data lineage, statistics reports produce inconsistent data for voter, ballot, and so on.

SVRS' intel-Merge Service Agent successfully resolves the drawbacks of conventional merge and much more. The benefits include: ELECT can determine the rules for which voter record becomes the survivor (e.g., Voter with most recent registration date or voting history). (LM-9.4.1.3.1)

With consistent data lineage, staff can seamlessly time travel backwards on a merged record. The reports produce consistent data.

- During a merge, ALL historical information from the non-survivor is automatically merged, not duplicated, to the survivor. This includes previous registration, images, list maintenance, election participation, ballots, candidate, and so on. This makes it easy to maintain the solution in long term.
- If records were merged in user error, Staff can undo a merge at any point in time without loss of data, duplication, or orphan records. The non-survivor record is restored to a state as it existed before the merge. This includes cross county data. (LM-9.4.1.3.2)
- After the merge, the non-survivor record virtually ceases to exist as a separate record so it does not appear as canceled voter or otherwise. The impacted counties are notified about the merge. (LM-9.4.1.3.3).

SVRS Election Analytics provides voters voted more than once report, this incudes voter who voted when they had separate records but were merged after. (LM-9.4.1.4)

4.3.10.9 Managing Accurate List

SVRS ****** Services supports list maintenance processes as required by federal, state, and local election laws including (LM-9.5.1.1) –

- Merge duplicates voters
- Move voter based on change of address using data from USPS, DMV, or Voter Notice
- Update voter status based on voter activity (or inactivity for a period)
- Cancel voter due to ineligibility of felon, deceased, citizenship, or residency

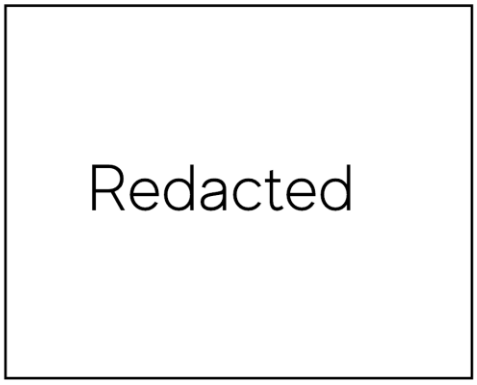
In addition to above mentioned List Maintenance process, SVRS also supports ingestion of data from Voter registration agencies as designed by NVRA to ensure eligible voter list is up to date. (LM-9.5.1)

Change of Address (COA)

In the State of California and Maryland, SVRS supports processing of NCOA records from USPS (LM-9.5.1.1). SVRS provides functionality a) export the voter data USPS change of address processing b) import the results of change of address processing. SVRS' NCOA Partner Data Integration Service filters from incoming file those records that do not show voter has moved or voter move is older than what is on SVRS. The voters who have moved are queued for processing, as shown in Figure 33: List Maintenance Queue.

SVRS uses the change of address information (e.g., Forwarding Information, Forwarding State, and Parsed Forwarding County Code attributes) to determine the type of move such as a) In-Locality Move b) Out of Locality Move c) Out of Commonwealth Moves (LM-9.5.1.1.2).

In-Locality Move: When forwarding information is available in the incoming response, SVRS solutions auto-precincts the parsed address from NOCA response and uses the effective change date to update the voter record. The voter is queued for in-locality move residency confirmation notice. If records are not precinctable, then the record is flagged for manual processing. (LM-9.5.1.1.3)



Out of Locality Move: Voter is automatically queued for out of locality move residency confirmation notice. ELECT can choose to enable the forward to the new locality option to send the NCOA record to the new locality (LM-9.5.1.1.4).

Out of Commonwealth Move: Voter is automatically queued for cancellation notice for out of state move (LM-9.5.1.1.5).

For the State of Virginia, Team Canton modifies SVRS to integrate with VA DMV to receive Out-of-State Notices, voters who are matches with the DMV data are sent two cancellations notice one. One to the new forwarding address and another to previous address (LM-9.5.1.1.7).

Residency conformation notices are sent to the new forwarding address but if no forwarding address was available, then the notice is sent to the old address. ELECT can configure by notice type the number of days (e.g., 30 days) to wait for response from voter before updating the voter to the configured status and status reason (LM-9.5.1.1.6)

Process returned notice

Staff can scan the completed confirmation notices returned from voter into the voter registration processing queue and process them similar to any voter registration form. As shown in Figure 36: Voter Documents Scanned Batch below, the voter registration queue can accommodate various types of return notices. (LM-9.5.2.4)

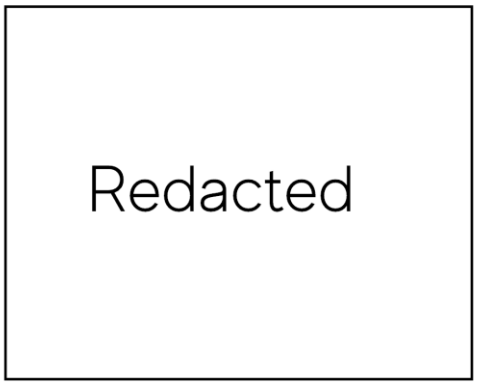


Figure 36: Voter Documents Scanned Batch

Process undeliverable notice

Staff can process incoming undeliverable notices in bulk or individually either manually or using scanned barcode (CAR-8.1.7, LM-9.5.2.3). As shown in Figure 37: Alternative confirmation mailing, staff can print label for alternative confirmation mailing on-demand from voter record or send to print queue for later (LM-9.5.2, LM-9.5.2.1).

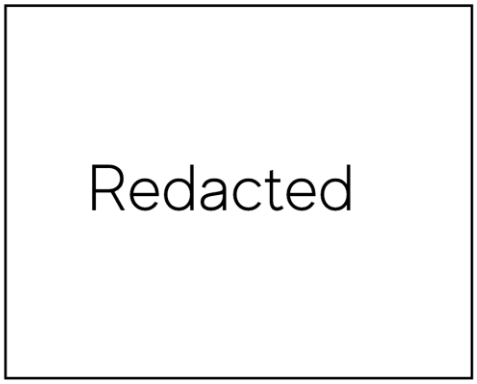


Figure 37: Alternative confirmation mailing

As shown in Figure 38: Export Alternative Mailing Notice below, Staff has option to export the mailing data for a third-party mailer (LM-9.5.2.2).

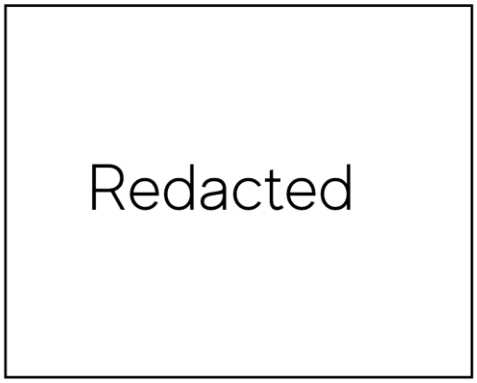


Figure 38: Export Alternative Mailing Notice

Reactivation of Voter

SVRS workflow engine is configured to automatically update the voter record based on both activity or lack of activity of the voter within a qualified time. Here are examples of scenarios where voter is activated automatically (LM-9.5.4) –

- Voted in an election
- Responded to a notice
- Signed a petition
- Updated the registration via Citizen Portal

Cross-Locality Notification

SVRS Messaging Service supports notification between localities and ELECT. Notifications are flagged as those that require response from locality versus informational only. Notifications are used to communicate list maintenance process such as matches (e.g., Felon, deceased, NCOA, DMV), merges, and voter moves. All counties affected by the voter move or merge are sent notifications. Typically, we have automated the localities response to voter move out and voter merges. (LM-9.5.5, LM-9.5.5.1)

4.3.10.10 Removing Voters Under Law

This section describes how SVRS meets the requirement to remove voters under law.

Inactive voters not voted in two-federal elections

SVRS Mass Update Service provides capability for locality staff to run a job and get a list of voters who are currently inactive and has not participated in at-least two federal elections. They can bulk update the status of the voter to cancelled with the reason of ‘Inactive – not voted in two federal elections. These voters are automatically queued for cancellation notice. (LM-9.6.1)

Out-of-State Moves

SVRS provides Data Integration Services design architecture that supports both real-time and batch data processing of change of address information from multiple data sources such as NCOA, DMV, and ERIC. SVRS supports algorithm based matching and geo-data to determine voters who moved within locality, within state, within commonwealth, outside of commonwealth. ELECT and locality can configure custom notices to be auto-generated based on move type. For example, two type of notices per voter shall be auto generated one for the new address and the other for old address when voter is determined to have moved out. (LM-9.6.2)

Voter request to Cancel

Staff can cancel the voter on SVRS after they scan and attach the cancellation request form to the voter record. As shown in Figure 39: Cancellation request form document Information below, SVRS captures document metadata including receive data and description to include if document was notarized.

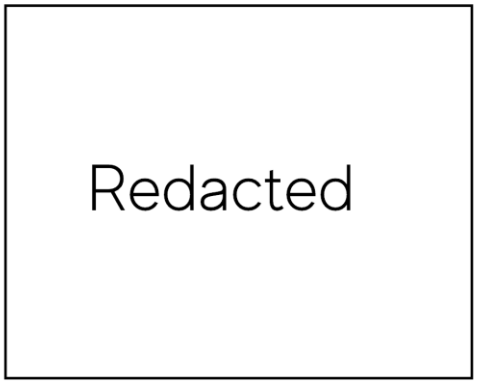


Figure 39: Cancellation request form document Information

Deceased Check

The following section describes how SVRS meets the Cancellation of Deceased requirements (LM-9.6.7, LM-9.6.7.1)

4.3.10.10.1 Partner Data Integration Services

SVRS' Partner Data Integration Services supports real-time (API) and batch (File) ingestion of deceased data from Bureau of Vital Statistics (BVS). A master data of the deceased records thus ingested is maintained within SVRS for matching, audits, analysis, and reporting. (EL-13.2.1)

Team Canton is familiar with Electronic Registration Information Center (ERIC); team will configure a pipeline on Partner Data Integration Service to ingest ERIC data via Rest API. The pipeline with scan for new data and automatically merges the new data into deceased master data. (EL-13.4.2)

4.3.10.10.2 Deceased ****** Service Agent

SVRS offers deceased ****** Service Agent an accurate, fast, and reliable matching solution driven by machine learning models.

4.3.10.10.3 Deceased-to-Deceased Matching Model

When records come from more than one sources, the deceased-to-deceased matching model links the related records, at the master data level, if they belong to same deceased individual. This helps to avoid duplicate work for the staff. This model also helps to merge and incrementally changed data. (LM-9.2.1.2).

4.3.10.10.4 Voter-to-Deceased matching Model

The voter-to-deceased matching models swiftly determines incoming records as 'matching pairs', 'possible matches', or 'not matching' based on thresholds. Oregon Votes' matching models can intelligently navigate around typos, Soundex, "Smartnames", partial data and so on. Initially, Voter-to-deceased matching model can

be created based on Name, date of birth, DMV Number or last 4 digits of SSN. If Agency chooses, matching model can be revised to consider additional data points for match basis.

4.3.10.10.5 High Confidence Deceased Match

SVRS' workflow engine supports orchestration of work-flow path to automatically process a high-confidence deceased match. Here is an example of workflow to process 'matching pairs' of voter-to-deceased:

1. Matching Service links deceased individual record to the matching voter record.
2. List Maintenance Service updates the status of voter to 'Cancelled' with reason of 'Deceased'. Captures the effective date of status change and deceased date.
3. Notice Service queues the record for 'Deceased' notice.
4. Election Management Service updates the voter is ineligible on the upcoming election's workspace.
5. Ballot Service suspends the ballots.

4.3.10.10.6 Possible Deceased Match

When Staff logs into SVRS, on the List Maintenance dashboard they will find the total counts of pending deceased matches, as shown in Figure 33: List Maintenance Queue.

Staff navigates to the Process Deceased queue and begins to work them on a FIFO method. Staff can concurrently work the queue without interfering in each other's records. As shown in Figure 40: Deceased Match Queue , Staff can search and sort through this queue by Received dates, Math status, Voter Name, and/or Voter ID. If already processed, they can search by processed dates, processed by, and if there were any errors in the process. Staff has the option to export the list.

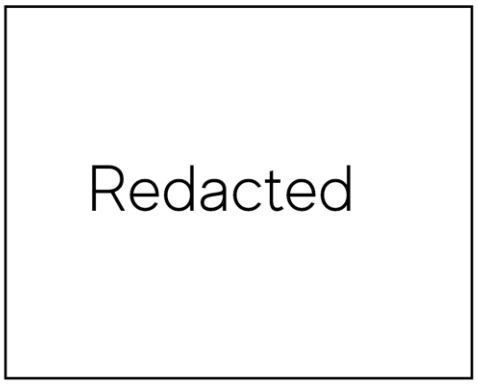


Figure 40: Deceased Match Queue

The voter record and matching deceased individual record are shown next to each other. Staff can review the information and quickly determine the validity of the match as valid or invalid. The voter that is confirmed deceased is automatically marked as 'Canceled' with reason of 'Deceased' and date when record was processed as effective date. Matches that are determined as invalid matches, the matching pair is removed

from the work queue. Staff can remain in the work queue, process next, skip a record, or go to previous without having to navigate back to another screen.

SVRS preserves the deceased match activity. When Staff must undo a deceased match, because an individual previously deemed deceased is not actually deceased or user error, they can do so from the Voter Detail screen.

4.3.10.10.7 Undo Deceased Match

To undo deceased match from Voter Detail screen, Staff can do so as follows:

1. Using Quick Search feature on the top bar, search the with voter name or voter ID.
2. On the Voter Detail screen, go to the List Maintenance Tab.
3. Click on 'Deceased Match' icon to filter historical deceased matches to the voter. Select the 'Undo' option and provide a reason for the un-match.
4. SVRS uses Voter Snapshot to restore the record to the previously known good state of the voter.

The history of deceased match and the subsequent un-match is preserved for audits, easily accessible on voter record.

Felon Check

The following section describes how SVRS meets the Cancellation of Felons requirement (LM-9.6.5)

4.3.10.10.1 Partner Data Integration Services

SVRS' Partner Data Integration Services supports real-time (API) and batch (File) ingestion of felon data from Virginia State Police (VSP). A master data of the felon records thus ingested are maintained within SVRS for matching, audits, analysis, and reporting. (EI-13.2.1)

SVRS' Partner Data Integration Service is scale-able to ingest felon records from new agencies. The Canton Group would configure a new data pipeline to scan for incoming files from the partner agency, as files become available, they are automatically merged into felon master data.

SVRS provides a data entry screen to manually add or restore felon records.

4.3.10.10.2 Felon ****** Service Agent

SVRS' offers Felon ****** Service Agent which is an accurate, fast, and reliable matching solution driven by machine learning models.

4.3.10.10.3 Felon-to-Felon Matching Models

SVRS uses the same design pattern as explained under section Deceased Check to manage a master data of felons using Felon-to-Felon Matching models. This helps with scenarios such as avoiding a duplication of work for Staff and updating the incarcerated status so an individual who is no longer in custody is not included in the

list of incarcerated felons for matching. This model also helps to merge and incrementally changed data. (LM-9.2.1.2).

4.3.10.10.4 Voter-to-Felon Matching Models

SVRS uses the same design pattern as explained under section Deceased Check to determine incoming voter records as ‘matching pairs’, ‘possible matches’, or ‘not matching’ based on thresholds. Here Voter-to-Felon matching models is used.

4.3.10.10.5 High Confidence Felon Match

SVRS’ workflow engine supports orchestration of work-flow path to automatically process a high-confidence felon match. Here is an example of workflow to process ‘matching pairs’ of voter-to-felon:

1. Matching Service links felon individual record to the matching voter record.
2. List Maintenance Service updates the status of voter to ‘Cancelled’ with reason of ‘Felon’ and captures the effective date of status change as well as the Sentence/Incarcerated date.
3. Notice Service queues the record for ‘Felon’ notice.
4. Election Management Service updates the voter is ineligible on the upcoming election’s workspace.
5. Ballot Service suspends the ballots.

4.3.10.10.6 Possible Felon Match

SVRS uses the same design pattern as explained under section Deceased Check to show the queue of possible felon matches so staff can process effectively.

The voter that is confirmed felon is automatically marked as ‘Canceled’ with reason of ‘Felon’ and date when record was processed as effective date.

4.3.10.10.7 Undo Felon Match

SVRS uses the same design pattern as explained under section Deceased Check to undo felon match and resolved to the previously known good state of the voter.

Non-Citizenship Check

The following section describes how SVRS meets the Cancellation of Non-citizens requirement (LM-9.6.4)

4.3.10.10.1 Partner Data Integration Services

SVRS’ Partner Data Integration Services supports batch (File) ingestion of non-citizen data from Department of Motor Vehicles (DMV). A master data of the non-citizen records thus ingested are maintained within SVRS for matching, audits, analysis, and reporting. (EI-13.2.1)

SVRS provides a data entry screen to manually add or remove non-citizen records.

4.3.10.10.2 Noncitizen ****** Service Agent

SVRS' offers non-Citizen ****** Service Agent which is an accurate, fast, and reliable matching solution driven by machine learning models.

4.3.10.10.3 Noncitizen-to- Noncitizen Matching Models

SVRS uses the same design pattern as explained under section Deceased Check to manage a master data of non-citizen using Noncitizen-to- Noncitizen Matching models. This model helps to merge and incrementally changed data. (LM-9.2.1.2).

4.3.10.10.4 Voter-to-Noncitizen Matching Models

SVRS uses the same design pattern as explained under section Deceased Check to determine incoming voter records as 'matching pairs', 'possible matches', or 'not matching' based on thresholds. Here Voter-to-Noncitizen matching model is used.

4.3.10.10.5 High Confidence Noncitizen Match

SVRS' workflow engine supports orchestration of work-flow path to automatically process a high-confidence non-citizen match. SVRS uses the same design pattern as explained under section Deceased Check.

4.3.10.10.6 Possible Noncitizen Match

SVRS uses the same design pattern as explained under section Deceased Check to show the queue of possible non-citizen matches so staff can process effectively.

The voter that is confirmed non-citizen is automatically marked as 'Canceled' with reason of 'non-Citizen' and date when record was processed as effective date.

4.3.10.10.7 Undo Noncitizen Match

SVRS uses the same design pattern as explained under section Deceased Check to undo Noncitizen match and resolved to the previously known good state of the voter.

Adjudicated Incapacitated Check

The following section describes how SVRS meets the Cancellation of Adjudicated Incapacitated requirement (LM-9.6.6)

4.3.10.10.1 Partner Data Integration Services

SVRS' Partner Data Integration Services supports batch (File) ingestion of adjudicated incapacitated data from Courts: The U.S. District Courts, local Circuit Courts, other states' District/Circuit Courts. A master data of the adjudicated incapacitated records thus ingested are maintained within SVRS for matching, audits, analysis, and reporting. (EI-13.2.1)

SVRS provides a data entry screen to manually add or remove adjudicated incapacitated records.

4.3.10.10.2 Adjudicated Incapacitated ****** Service Agent

SVRS' offers Adjudicated-Incapacitated ****** Service Agent which is an accurate, fast, and reliable matching solution driven by machine learning models.

4.3.10.10.3 Adjudicated Incapacitated -to- Adjudicated Incapacitated Matching Models

SVRS uses the same design pattern as explained under section Deceased Check to manage a master data of non-citizen using Adjudicated Incapacitated -to- Adjudicated Incapacitated Matching models. This model helps to merge and incrementally changed data. (LM-9.2.1.2).

4.3.10.10.4 Voter-to-Adjudicated Incapacitated Matching Models

SVRS uses the same design pattern as explained under section Deceased Check to determine incoming voter records as 'matching pairs', 'possible matches', or 'not matching' based on thresholds. Here Voter-to-Adjudicated Incapacitated matching models is used.

4.3.10.10.5 High Confidence Adjudicated Incapacitated Match

SVRS' workflow engine supports orchestration of work-flow path to automatically process a high-confidence adjudicated incapacitated match. SVRS uses the same design pattern as explained under section Deceased Check.

4.3.10.10.6 Possible Adjudicated Incapacitated Match

SVRS uses the same design pattern as explained under section Deceased Check to show the queue of possible adjudicated incapacitated matches so staff can process effectively.

The voter that is confirmed adjudicated incapacitated is automatically marked as 'Canceled' with reason of 'adjudicated incapacitated' and date when record was processed as effective date.

4.3.10.10.7 Undo Adjudicated Incapacitated Match

SVRS uses the same design pattern as explained under section Deceased Check to undo adjudicated incapacitated match and resolved to the previously known good state of the voter.

Send Cancellation Notice

SVRS Notice Service is configured to automatically queue a voter record for notices based on Voter Status and Reason. When voters are marked as cancelled, the workflow engine uses the configuration to automatically queue the voter for Cancellation Notice. The reason for cancellation is dynamically populated on the cancellation notice template. (LM-9.6.8)

List of Cancelled Voters

As part of EVAS report, SVRS Election Analytics provides voters who were cancelled within a period based on the provisions of NVRA Act program. (LM-9.6.9)

4.3.10.11 List Maintenance Calendar Restrictions

For the State of Virginia, Team Canton can configure additional restrictions to the List Maintenance based on time periods in order to comply with Virginia Law. These examples of the restrictions include to add a pause to the job for processing NCOA, ERIC, DMV Out-of-State, and so on during this pause period configurable by ELECT. (LM-9.7.1)

4.3.10.12 List Maintenance Reports

SVRS Election Analytics offers ELECT and Locality both ad-hoc queries and advanced analytics option to produce custom reports such as –

- Pending list maintenance activities past a user specified age
- List of list maintenance activities completed within a specified range and user
- Performance assessment of ****** algorithms

SVRS sufficiently supports ELECT's list maintenance reporting needs. (LM-9.8.1)

4.3.11 Online Citizen Portal

4.3.11.1 Online Citizen Portal General Considerations

SVRS Online Citizen Portal Solution is a modern web and mobile friendly for citizen to securely register to vote, lookup voter information, submit absentee request application, and print voter card.

SVRS as implemented in State of Maryland is compliant with the following requirements:

Multi-language support – Currently it supports English and Spanish however support for additional language is a configuration. Team Canton works with the ELECT's translation vendor to receive and import translated text equivalent to English. Once the new language file is published, it is accessible to the user.

Accessibility and Optimization – SVRS is built on reactive design so it scales up and down to different mobile phones. Team Canton works with ELECT to ensures the citizen portal is optimized for search engines and ranked on top for key words associated to voter registration within State of Virginia.

Changes to Election Regulations – When Election Regulations changes, they may require updates to Online Citizen Portal, these updates can be anywhere from revision of displayed text, adding new feature, or changing the workflow. SVRS solution can handle these with our flexible system design such as displayed texts are configurable attributes, features are modular so changes minimally impacts other components, and rules behind the workflow are centralized, reused, and easy to manage.

User-Centered Design – We understand various user groups including ELECT, locality, voter advocacies, and other state agencies (e.g., governor's office) can potentially influence the design of the Online Citizen Portal.

The input from the user groups is not always available before the start of the design. Our design methodology is iterative and flexible to incorporate feedback throughout the process.

Voter Friendly Language and Terms – The display texts within Online Citizen Portal are configurable attributes so can be updated with input from ELECT.

Compliance with Compliance with Americans with Disabilities Act (ADA) 508 – SVRS meets the WACG 2.0 and ADA section 508. Our developers are equipped with the compliance checkers so they validate the work from the design through deployment.

Request or Print Voter Card, Notices – As shown in Figure 41: Online Citizen Portal - Print Voter Card below, voter can print voter card from the voter look up screen.

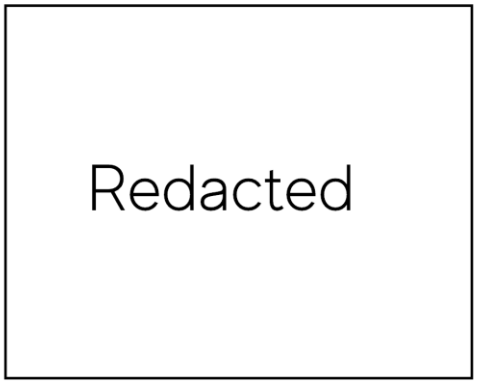


Figure 41: Online Citizen Portal - Print Voter Card

4.3.11.2 Online Portal Voter Registration

SVRS Online Portal Voter Registration (OLVR) solution, allows eligible Virginia residents to complete voter registration online securely.

As shown in figures below, OLVR solution offers a wizard style online form that takes the user step-by-step to successful fill out both voter registration application and absentee ballot application. (OCP-10.2, OCP-10.3)

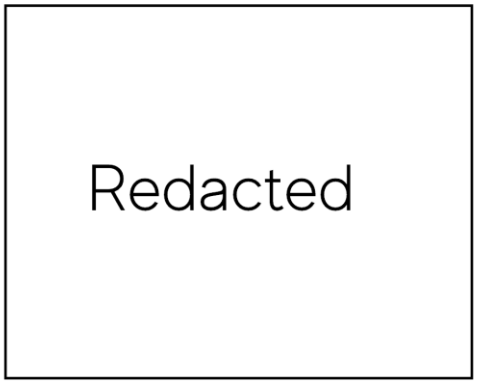


Figure 42: OLVR Voter Type

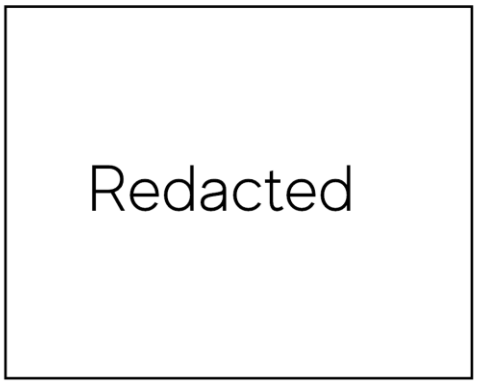


Figure 43: OLVR Voter Instructions Page

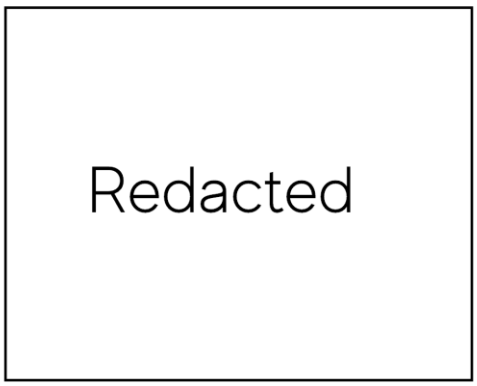


Figure 44: OLVR Step 1

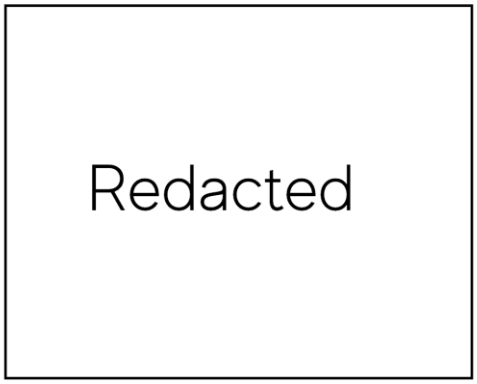


Figure 45: OLVR Step 2

SVRS OLVR solution minimizes duplicate registration by checking for any existing voter registrations using the Name, Birthdate, SSN4, and DMV Number received in the first two steps of the process. When an existing voter registration is found, SVRS provides a message to user that a previous registration was found and populates with the recent information on voter file. User can review and update the registration. In step 3, when user enters the residential address SVRS calls the GIS service to validate against known addresses within the locality. When address match is found then address is standardized, precinct, precinct-split, and locality are assigned. When address is not found, user can proceed to continue with non-standard residential address and manually choose the locality. Additionally, for VA, we can prepopulate residential address with address received from DMV. As user progresses through the OLVR steps, the partially completed application is saved. At the end of final step, SVRS provides the user with a receipt that includes the confirmation message, an affidavit number, and the date and time of submission. User can print this receipt. (OCP-10.2.1, OCP-10.2.2, OCP-10.2.5, OCP-10.2.6, OCP-10.2.7, OCP-10.2.8, OCP-10.2.9, OCP-10.3.2).

On step 9 Mail-in Request, user can review the status of their absentee ballot application along with election name, application status, and received date. If voter is not already an absentee for an upcoming election, the regular user (non-UOCAVA) completes the absentee request including absentee type such as one-time (select an upcoming election) or annual absentee, primary election party ballot preference, and indicate if they need assistance with ballot and type of assistance. UOCAVA user, can fill out the ballot preference between mail, email, and fax. User can check the status of the absentee ballot for an election along with when ballot was received. User, who is an absentee voter but not yet voted in an upcoming election, can request a replacement ballot after providing a valid reason. Team Canton ensure the Online Absentee Ballot request process is compliant with Federal and Virginia rules and regulations. (OCP-10.3.1, OCP-10.3.3, OCP-10.3.4, OCP-10.3.5).

Currently, OLVR as implemented in State of Maryland supports multiple languages including English and Spanish. However, OLVR is design to support more than those two languages. Team Canton imports the translation text received from translation vendors to enable to support additional languages and text. (OCP-10.2.3)

OLVR as implemented in State of Maryland is WCAG 2.0 complaint that means voters with visual impairments and print disabilities can use common screen readers to complete the voter registration. (OCP-10.2.4)

4.3.11.3 Look up Voter Information

SVRS Online Portal Voter Lookup Information (OLVI) solution provides following capabilities for registered voters of State of Virginia –

- Review voter registration record
- Find out where to vote
- Find out voting districts
- Find locality contact information
- View sample ballot
- Request a duplicate voter registration card
- Download Federal and Commonwealth Election Forms

As shown in Figure 46: OLVI Voter Search below, users enter first name, last name, date of birth, and Zip Code to access the voter look up portal. Voters view their registration and election activity in an accordion with information neatly grouped into collapsible sections.

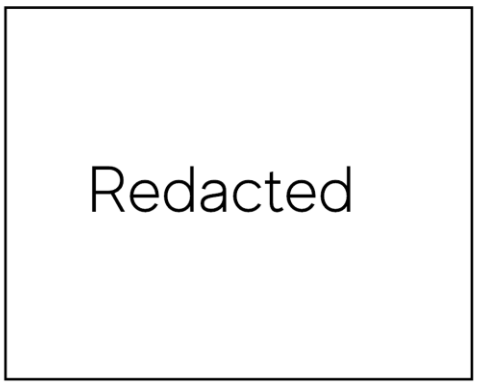


Figure 46: OLVI Voter Search

As shown in Figure 47: OLVR Voter Registration Record below, voter's name, residential address, mailing address, party affiliation, and most recent registration date are displayed. (**OCP-10.6.1)**

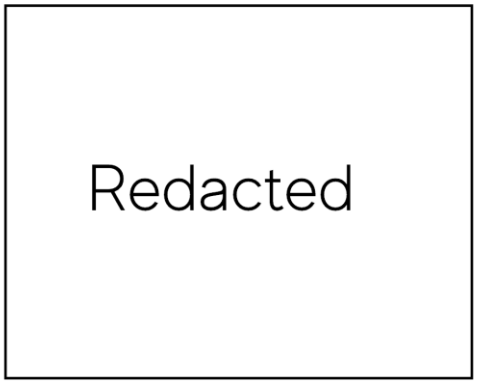


Figure 47: OLVR Voter Registration Record

As shown in figure below, my past voting history displays the past elections the voter voted in. (OCP-10.4.1)

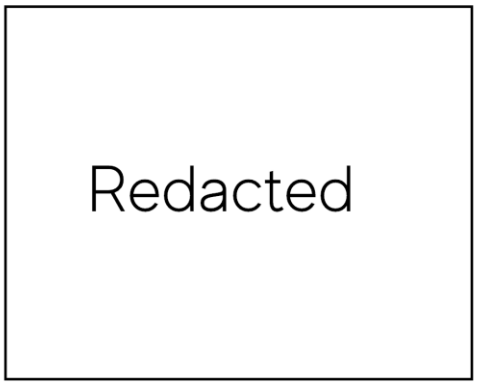


Figure 48: OLVI Voting History

As shown in figures below, My Voting Center Information and My Voting Districts displays the voting location and districts. (OCP-10.5.1)

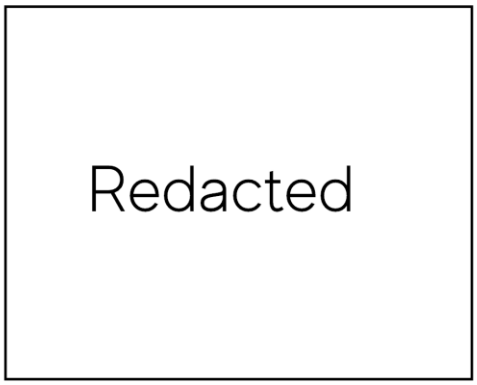


Figure 49: OLVI Voting Center

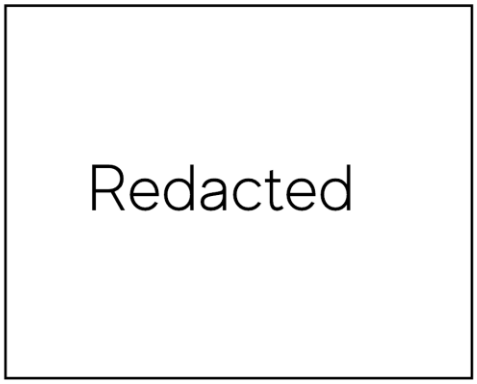


Figure 50: OLVI Voting Districts

The location board of Elections information will be customized for State of Virginia to include Locality contact information. Additional sections are added to view electronic sample ballot for a voter and download Federal and Commonwealth Election Forms. (OCP-10.7.1, OCP-10.8.1)

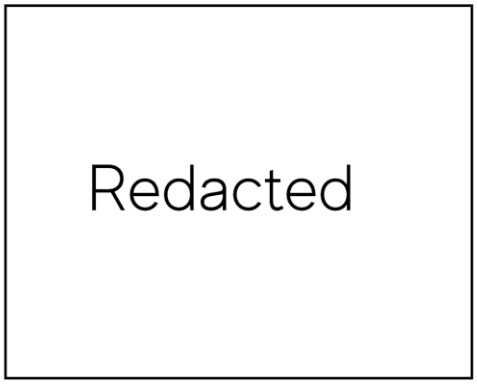
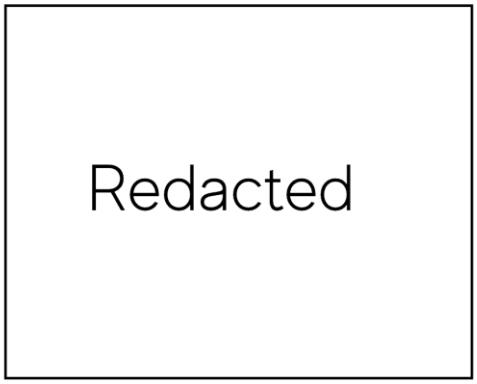


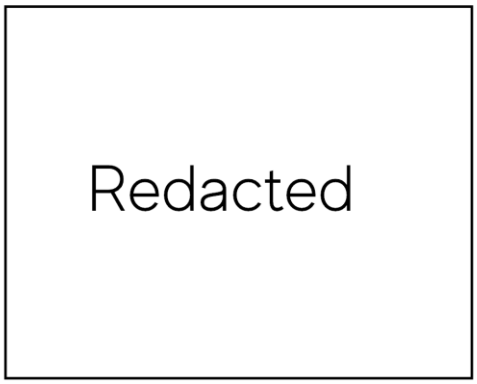
Figure 51: Locality Contact

4.3.12 Security

The following figure shows how at the center of our SDLC process is security. We include security in every step of our development process from the conceptual stage to development, releases, and operations. Security starts from requirements gathering, during the design of the system, in the architecture, development deployment, data security, and while integrating with third parties Integration



At the Platform level, security is achieved by addressing separately the Application from the Infrastructure.



The application Security is responsibility of the Development process and includes data security. Please refer to section EI-13.1Database Access for data security details.
SVRS deploys in Microsoft Azure Government, one of the top Cloud Secure Cloud Service Providers in the world. The following is an excerpt from Microsoft Azure Government Documentation (Azure Government Overview - Azure Government | Microsoft Docs):

“US government agencies or their partners interested in cloud services that meet government security and compliance requirements, can be confident that Microsoft Azure Government provides world-class. security, protection, and compliance services. Azure Government delivers a dedicated cloud enabling government agencies and their partners to transform mission-critical workloads to the cloud. Azure Government services handle data that is subject to certain government regulations and requirements, such as FedRAMP, NIST 800.171 (DIB), ITAR, IRS 1075, DoD L4, and CJIS. In order to provide you with the highest level of security and compliance, Azure Government uses physically isolated datacenters and networks (located in U.S. only).”

This compliance plus our application-level security (data, and network), and BC/DR implementation.

4.3.12.1 SEC-11.1: Maintenance of User Accounts

SVRS encapsulates user management in the security micro service. This component only responsibility is to provide search, filter, add new users, edit existing users, authentication, role-based authorization, Jurisdiction management (State, County hierarchy and horizontal and vertical rules), session management, logs and audit records, login features.

User search is text based on a contextual match. The search has multiple criteria and single criteria search is possible too. The user (usually an administrator) can filter the results using distinct filter criteria. The search results contain not only exact matches, but also similarity results. (SEC-11.1.1). The user management service allows the creation of new users and the modification of existing users accounts (SEC-11.1.2) including enabling or disabling users (SEC-11.1.5) and (SEC-11.1.5.2). An administrator of the system has permissions to add users and assigns roles and permissions to users in the authorization window. Roles and permissions are configurable. Some roles are predefined and editable. The admin has the rights to create new roles and permissions as the business requires (SEC-11.1.3). The admin cannot modify a username but can assign/unassign roles (SEC-11.1.3.2). An authorized administrator with the appropriate permission (“super user”) can modify roles, permissions, and usernames (SEC-11.1.3.3). The solution has the capability to clone a user so it can have the same roles and permissions as the original user to minimize the tasks of adding manually these permissions and promote uniformity. In the user management screen from the list of users, a user with the right permissions can select the user to clone. A new window shows the details of the new user with the roles of the original user. The username and other information of the original user is read only in the screen so the admin has to enter the new user information to complete the process. (SEC-11.1.3.1).

The user architecture is based on a hierarchical multitenant model. In this hierarchy a state user with the appropriate privileges can access a specific County information. The system generates the audits regardless of the user of the system. The system logs modifications performed by the State user in a county system and show them in the reports or the data analytics component visualizers (SEC-11.1.4).

The system keeps usernames unique across active and inactive users, as well as email addresses per tenant. The system validates the usernames and email addresses before submitting the add user request to the backend (SEC-11.1.6). Authorized user to a specific jurisdiction i.e., locality or state. This is part of our multitenant hierarchy and role-based access management architecture. (SEC-11.1.7). SVRS uses ***** service to configure policies that allow users to submit access request for production and non-production environments. Admin can login to a secure portal to review access request then approve or deny access request submissions.

The solution provides the environment and the version number of the deployed application in the footer of the Views. This allows the users to know what is their current environment to prevent them accidental modification of Production data (SEC-11.1.9).

The system when deployed in Azure will integrate with Azure Active Directory and Azure Multi Factor Authentication (MFA) services. This is an out of the box functionality that integrates with SVRS via configuration. (SEC-11.1.9), (SEC-11.1.10).

A user that has been inactive for a configurable amount of time, will be asked to continue with the session or the user gets logout of the system (SEC-11.1.10.1).

4.3.12.2 SEC-11.2: Role-based Access Control

The solution supports roles based on CRUD operations (Create, Update, Read, and Delete (inactivate)) as well as access to different functions in the system. If a user does not have access to a certain functionality, the access or action buttons are either made not visible or are disabled for the user. (SEC-11.2.1). Our understanding of user types and user roes is that a role defines the functional access a user has, and the permission level (i.e., generate, read, update, print, archive) within that function. A user type defines the scope of data the user can access while performing that function. In this regard, the system provides user roles and user functions, as well as data access permission (CRUD) (SEC-11.2.2). The solution allows to group multiple functions into a role to represent a common functionality assigned to a specific user or users. The solution implements the grouping of roles into role groups to facilitate this logic macro assembly of functionalities (SEC-11.2.3). We control roles at the API level (functions) and at the CRUD data level. For example, a user can have roles to access a specific function and this can be limited to be read only. This is done via configuration without code changes. (SEC-11.2.4),

4.3.12.3 SEC-11.3: IP Whitelisting

The system architecture at the infrastructure level uses an access control lists (“white” list) to allow only authorized users entering the system. The access control list contains the user and a pre-defined IP Address. The logs of the system also capture this information for every incoming request. The user is then authenticated and authorized using the provided security component or can be integrated with Azure AD/MFA. The access to the system is thru the web application (portal) using the Firewall that secures the entire system. (SEC-11.3.1), (SEC-11.3.2), (SEC-11.3.3).

4.3.12.4 SEC-11.4: Obfuscation of Personally Identifiable Information (PII)

SVRS by default supports obfuscation (masking) of Personal Identifiable Information such as confidential information. Only authorized users belonging to a specific role group are allowed to view the PII in readable format. Access to the PII data is recorded into the logs as part of our database logging and the other system logs. Any edits to the PII are tracked by the system logs. An administrator of the system should be able to move data to other environment such as development, test and training. The PII can be filtered via the extract scripts. (SEC-11.4.1), (SEC-11.4.2), (SEC-11.4.3), (SEC-11.4.4)

4.3.12.5 SEC-11.2: Who's Online?

SVRS’s security module logs who is logged in the system, from where the user is accessing SVRS, when the session started, and the time the user session is active. This information is persisted in the SVRS database (SEC-11.5.1).

4.3.13 COV Security

4.3.13.1 SEC-15.1: Enterprise Security

SVRS is architected and designed to be eventually complaint with Federal, State, and Local Enterprise Architectures, in this case the COV Enterprise Architecture. Here are the highlights (not exhaustive list) of SVRS design principles and architecture patterns that follows COV ITRM Policies, Standards and Security Procedures:

- SVRS is developed based on recommended best practices for software development, data modeling, business process flows, Service Oriented Architecture, and Cloud based infrastructure. SVRS can be seen as an interoperable collection of services with each one of them following the same architecture pattern, code and data standards.
- We use Gov cloud infrastructure with NIST SP 800 53 R4 compliant security policies.

SVRS complies with the current COV ITRM Policies, Standards and Security Procedures, as applicable. (COV SEC-15.1.1).

4.3.13.2 SEC-15.2: Commonwealth

Security properties of software are dynamic and static. Static security properties are those that can be tested during development time. Dynamic properties are mostly emergent properties that can be measured at run time. SVRS hosted on Azure Cloud is NIST SP 800 53 R4 compliant, as verified through ***** guidelines and verification reports. We fix all the vulnerabilities as recommended by NIST SP 800 53 R4. We perform scheduled third-party security penetration and static test to make sure we are in compliance with Security Policies. Team Canton has worked with COV ITRM Security Policies Standards on other occasions as a Cyber Warfare Officer conducting Pen Testing, Vulnerability Assessments and Governance for the Commonwealth. Team Canton is confident, SVRS will be compliant with COV ITRM Security Policies and Standards. (COV SEC-15.2.1).

As we mentioned in section SEC-15.1, SVRS is architected and designed to be eventually complaint with Federal, State, and Local Enterprise Architectures, in this case the COV Enterprise Architecture. Here are the highlights (not exhaustive list) of SVRS design principles and architecture patterns that follows COV Enterprise Architecture ITRM Policies and Standards:

- We use best practices of DevSecOps to build and manage SVRS SDLC including CI/CD and operations.
- The solution follows the Reactive Architecture, an architecture for building service-oriented applications that are scalable, resilient and built as decoupled microservices that communicate with each other via messaging.
- We use Domain Driven Design (DDD) development. Business (Functional) Architecture (EBA) is defined as Domains. DDD allows us to decompose the domain in sub domains and design them one at a time as decoupled components. Each one of these components define an Object Model and they communicate with each other using messaging.

- Our Information Architecture (EIA) allows us to define how messages and information flow horizontally among components (services) and how they flow vertically to the data layer for persistence or reads. The information security is also defined in this stage. we define how the data is secured in flight and at rest. Secure protocols and encryption algorithm, and masking hashes are determined. We standardize our data model, and use our External Interfaces / Integration Service to manage data format transformations.
- Our Solution Architecture (ESA) deals with components that interact with the custom application such as GIS, Data Lake, External Interfaces, Messaging Services, Jobs schedulers, etc. We determine what is the best tool or component to use to achieve the best solution for the system. This Architecture also provides a high-level view of the solution mentioning the Enterprise Business Architecture, Enterprise Technical Architecture and Enterprise Information Architecture.
- Our Enterprise Technical Architecture (ETA) defines the different architectural views: systems view (layered architecture showing the presentation, API , Business and Information in detail including the software and components used to build each view), information view (shows the Object relation Model - ORM, the transactional database, replica database for reporting and data analytics , data warehousing and the data lake, policies for data governance, data lineage, data standards, data sharing) and infrastructure view (design and all the resources such as containers, orchestrators, API Gateways, Firewalls, the Virtual Networks and the ports and IP addresses defined in the Aure subscription), having the security architecture across these views. It also defines the Business Continuity / Data Recover plan.
- We maintain the software and tools catalog with the current version, end of life dates so we can upgrade them on time before they represent a risk to the solution.
- Privacy, security, confidentiality and statuses are also defined in this ETA.

Our External Integration Microservice (component) allows for a flexible, configurable multi-protocol and multi data format exchange. We can accept any type of data format and transform our internal format into any other external data format such as XML, JSON, HL7, CSV, Fix Length, etc. At the protocol level, we accept SOAP, REST, FTP, Queue protocols as specified in the COVA EA Data Standard. Our transformations module can be defined to generate and parse any enterprise data standards such as: Enterprise Data Management Person Core Matching. We can map these to our internal data model while executing the integration with other systems. (COV SEC-15.2.3).

Our flexible, configurable and easy to maintain solution supports, with minimal effort, changes to the UI layout to meet the inclusion of Commonwealth banner where needed. Our web browser presentation layer is designed around a User Centric Approach (intuitive, easy to use, accessible to all users). We used Domain Driven Design, a methodology where one of the principles of the design is based on the user language helps us avoid technical terms but use more functional user terms. The system shows the user what is the status of their requests or processes. The user is always in control of the process and knows exactly where he/she is. Our design focus in having minimal collateral effects (errors and exceptions) we use “defensive programming’ to handle potential errors and minimize the error results traveling to the user. If that is the case, we word the errors with functional error messages rather than technical descriptions. Our UI design is standard and contains only information to what is relevant to the module in which the user is working on. (COV SEC-15.2.4)

4.3.14 General Administration

4.3.14.1 GA-12.1 Data and Audit Records

GA-12.1.1 Metadata, Audit and Master Records

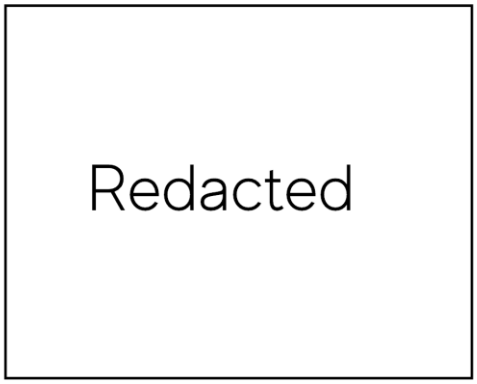
The system supports storage of data within the system. We use a relational database and blob storage to persist transactional data, audit records, documents, images, files, etc. (**GA-12.1.1.1**). The audit records are stored in the database or file storage, secured by security features and policies. The audit records can be purged following retention policies (**GA-12.1.1.2**). Any interaction with transactional data is recorded using triggers in the database or logs in the request and data layer recording contextual information such as the user, date timestamp, operation type and state of the data before the operation. For group of changes and distributed transactions we have a correlation id to group the changes associated to a specific set of transactions. (**GA-12.1.1.3**), (**GA-12.1.1.4**), (**GA-12.1.1.7**), (**GA-12.1.1.9**). In the next release we will have the ability to capture click counts per user for user processing time stats (**GA-12.1.1.6**).

The component where the operations are performed are intrinsically recorded by the audit records (**GA-12.1.1.5**). Audit records of user accounts are stored in journal tables associated to the user related tables. User request reports can be generated from these records in the Reporting/Data Analytics module. (**GA-12.1.1.8**). The audit records cover PII information. Audit records are only visible to administrators of the system. We provide tools to monitor and visualize continuously the logs and audit records. We can also filter the logs to hide PII information. (**GA-12.1.1.10**), (**GA-12.1.1.11**). The solution uses the Cloud (Azure) Network Time to synchronize all the times across the resources of the solution. (**GA-12.1.1.12**). Our ****** Agent Service support management of master data for voters, felons, deceased, adjudicated incapacitated, and non-citizens. Voter data is dynamically linked to the incoming external agency data to provide an up-to-date voter registration data. (**GA-12.1.1.13**).

GA-12.1.2 Documents (Enterprise Content Management)

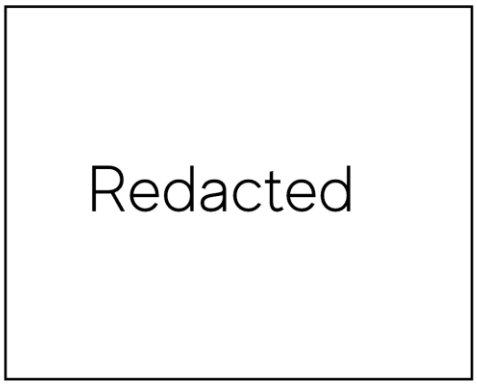
Documents are persisted as files under their original format, but whenever a document needs to be persisted as an image, we use standard image formats. (**GA-12.1.2.2**). We support Optical Character Recognition (OCR) by using industry leading UiPath Automation. This feature allows to extract plain text from the images (**GA-12.1.2.3**). The solution supports entering special characters in the input field of the Voter Registration windows. This allows to capture nonstandard ascii code (**GA-12.1.2.4**). During conversion, we should be able to digitalize data from registration cards; this is a manual or semi-manual process. The solution provides capability to store correspondence records in PDF format.

SVRS supports the attachment of documents in Voter (**GA-12.1.2.5**), Candidate, Locations. (**GA-12.1.2.1**). SVRS allows common image formats and PDFs to be attached (**GA-12.1.2.7**). Timestamps for upload date are added to all attachments (**GA-12.1.2.8**)



Voter image tab

Correspondences are issued (**GA-12.1.2.8**) and tracked on the Notices tab of voter (**GA-12.1.2.6**), but all images/attachments are added on the Images tab.



Voter Notices tab

GA-12.1.3 IT Lifecycle Management

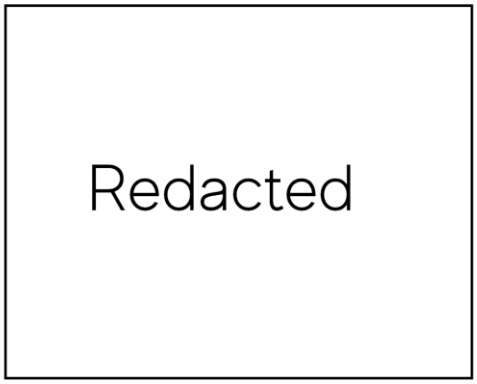
The design of SVRS includes support for retention of data. This is a procedure that must be defined during implementation of the solution. We can easily add functionality in the administration module to provide scheduled of data retention as a configuration parameter. Currently the data that passed their retention schedule is purged using ad hoc scripts. (**GA-12.1.3.1**), (**GA-12.1.3.1.1**), (**GA-12.1.3.1.2**). SVRS architecture implements the Reactive Architecture Principles: The solution is responsive, scalable, resilient, and not monolithic.

Responsive: We assure that the architecture and infrastructure allow for timely responses to request independent of the number of users or load size.

Scalable: SVRS architecture supports horizontal scalability of the micro services based on the container size and capacity. When a microservice is reaching its capacity, our platform creates a new microservice to support the excess of load.it knows how to route the requests to the microservices automatically.

Resilient: When a microservices health is deteriorating, the platform fires a new microservice to replace the failing one. This allows our system to be 24X7 up and running.

Message Driven: Each microservice maps a functional domain and only knows how to do that business logic. It does not persist state (stateless), so the API requests from the clients do not interfere with each other. The microservices therefore are decoupled services communicating via messaging to achieve workflows.



The above high-level description of our architecture assures that SVRS is performant, according to the SLAs, regardless of the amount of data entered in the system (**GA-12.1.3.1.3**).

GA-12.1.4 Code Tables

SVRS data model includes code tables. The access to the code tables is via a specialized GUI or using direct SQL. Only a user (administrator) with the right permission can maintain (create, Update, Inactivate/expire) the code tables. Expired codes are filtered out for their use in the application. The code tables are structured having in mind the jurisdiction hierarchy (state, county). Some codes apply to the entire hierarchy while others are subject to be unique at their level. This overwrite capability is supported by SVRS. (**GA-12.1.4.1**), (**GA-12.1.4.2**), (**GA-12.1.4.2.1**).

4.3.14.2 GA 12.2 Usability

GA 12.2.1 Functionality

SVRS client runs on a web browser. A user can have multiple sessions of the application in the same computer on multiple web browsers running concurrently. SVRS stores client preferences when the user initiates a session. The solution persists the user location (IP address, time stamp on the session initiation, user level (roles)). A privileged user (administrator) can lock access to functions of the application to lower-level users via the user management and user roles management views. (**GA-12.2.1.1**), (**GA-12.2.1.2**), (**GA-12.2.1.2.1**).

GA-12.2.2 User Interface

In general, our design is a Domain Driven Design (DDD), where we decompose the elections domains in sub-domains, preferably independent of each other. Specifically, our User Interface design includes the use of user-centered design best practices. We follow an Agile Process where the product owner, a person with extensive knowledge in the election’s arena, is a member of the design team. We also included feedback from actual users of a County in California. We used simple archetypes (personas) to represent election users. The Ideation process includes use of visual aid designs and workflows (mockups). The information architecture is defined in our DDD process to support the Object Model that implements the business logic. Our View Models

represent what the user needs in their workflows and maps the aggregates (object model) in the business layer. This mapping validates the structure and data being used in the UI. (**GA-12.2.2.1**). SVRS implements the Object Action Pattern, in which the user selects first the object (voter) and then the action to perform on it (create, update, etc.) (**GA-12.2.2.2**) (**GA-12.2.2.3**), (**GA-12.2.2.4**), (**GA-12.2.2.5**)

SVRS limits the access of functionality and data to the locality of the user. The application uses the properties of multi-tenancy with locality identifier (jurisdiction id) to segregate data access to the appropriate locality (**GA-12.2.2.6**)

GA-12.2.3 Search and Filter

SVRS provides with powerful searching and filtering capabilities. We offer a permanent search feature in the UI. A user can only search based on assigned privileges. User's roles are designed to allow users to search only in their localities (jurisdictions) or cross jurisdictions. A user can search statewide (unified search) or in multiple localities if they have the appropriate privileges. Searching is around functional criteria. SVRS offers searching with single or combined criteria. System attributes such as primary key, are not included in SVRS because these values have only meaning internally to preserve data integrity or facilitate searches. Allowing users to search by primary can lead to users accessing data that is not allowed to access, such as data from other jurisdictions or simply PII or confidential information. But, adding search by primary key would be very simple, because of the expandability of SVRS. (**GA-12.2.3.1**), (**GA-12.2.3.2**), (**GA-12.2.3.3**) (**GA-12.2.3.3.4**).

GA-12.2.4 Redaction, sub-setting, and substitution

ELECT to define set of data that is considered Personal Identifiable Information (PII), the SVRS solution encrypts the data and is only available on UI and reports for users with permission to view PII data otherwise it is masked (redacted). The level of masking can be setup at field level such as to show **** for SSN4 vs **/**/1987 for date of birth. (**GA-12.2.4.1, GA-12.2.4.2**). In case of protected (confidential) voter, the mailing address is required in addition to residential address, however mailing address is always used in the reports and labels. (**GA-12.2.4.3**). SVRS preserves PII information obscured or masked in all the environments. (**GA-12.2.4.4**).

4.3.14.3 GA-12.3 Process Management

GA-12.3.1 Transactions

No, we do not add comments at the transaction level. SVRS records the transactions and meta data about the transactions for audit purposes. By looking at our historical data you can determine what changed (**GA-12.3.1.1**).

GA-12.3.2 Election Calendar

SVRS has its own calendar support to describe calendar events and occurrences, such as pre & post milestones. This calendar is inherited from our UI framework in the concrete Windows (UI pages) (**GA-12.3.2.1**)

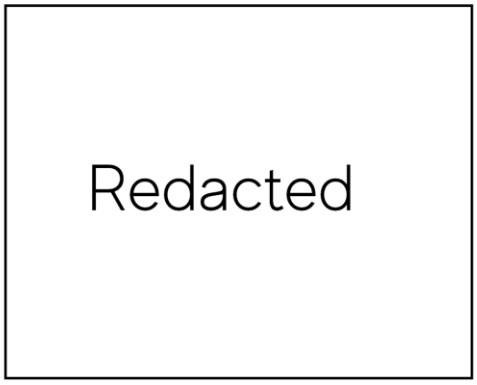
GA-12.3.3 Workflow & Work Allocation

To understand how SVRS effectively manages large number of voter and ballot workloads, please see sections 4.3.1 Voter Registration and section 4.3.5 Absentee Voting for more information (**GA-12.3.3.1**) (**GA-12.3.3.2**)

One user assigning workload to specific users is not something currently in-built; however, pushing work into workable queues for processing is very much part of the SVRS solution. (**GA-12.3.3.3**)

SVRS is built on the premise of “data in, data out”, as a result all primary screens in SVRS start either with an Advanced Search or a Queue card (**GA-12.3.3.4**)

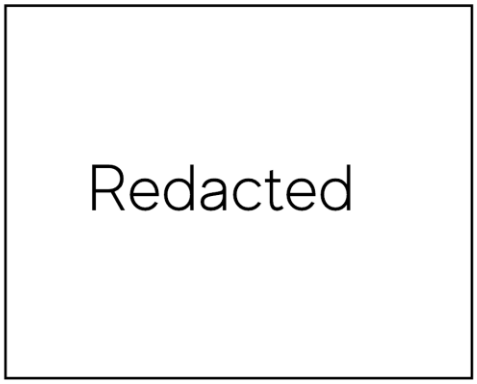
The Election Calendar is a System Configuration, and unique calendars can be created for each election type. This automatically calculates all key election dates during election creation. This acts as a template, and does not restrict modifications in the event a date needs to be modified for a unique circumstance. (**GA-12.3.3.5**)



SVRS System Configuration: Add Election Calendar details

While alerts can be built-in, SVRS comes with a homepage workspace which is intended to take the place of alerts or many reminder e-mails. It strives to answer the question “What should I do today?” or “Where is my jurisdiction at?”

At present, there is one default workspace. There are five homepage workspaces in active development: Odd Year Cleanup, Election Voters, Election Ballots, Election Event (Workers), and Administrative. (**GA-12.3.3.6**)



SVRS Homepage Workspace (Default)

4.3.14.4 GA-12.4 Relationship Management

GA-12.4.1 Authority

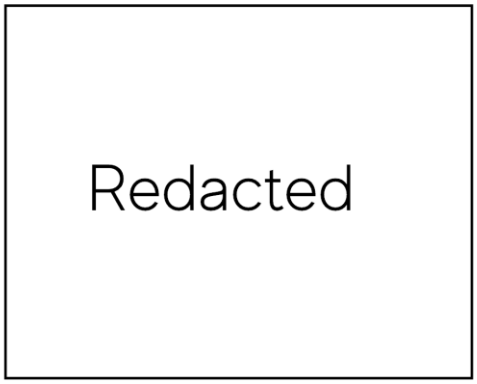
SVRS allows information updates to users with the appropriate privileges based on the user's locality (**GA-12.4.1.1**).

GA-12.4.2 Position holder Contact

For information on how SVRS handles Positions and Position Holders please see 4.3.2 Election Admin, EA-2.2 Manage Offices (**GA-12.4.2.1**) (**GA-12.4.2.2**) (**GA-12.4.2.3**) (**GA-12.4.2.3.1**) (**GA-12.4.2.3.2**) (**GA-12.4.2.3.3**) (**GA-12.4.2.4**) (**GA-12.4.2.4.1**) (**GA-12.4.2.4.2**) (**GA-12.4.2.5.2**). Additionally, on the Voter Record a user can select the "Election Officials" button to view the list of position holders for a particular voter.

In regards to importing, the initial data migration would include the setup of historical and current position holders; however, maintaining this is done via Election or manual maintenance using the Add/Edit feature (**GA-12.4.2.5**). SVRS does not currently create a "vCard" file format, however does maintain Position Holder contact information, including social media sites (**GA-12.4.2.5.1**). SVRS allows a search and extract data that can be used to print labels or report member information, depending on the business needs more specific standardized reports or correspondences can be created easily (**GA-12.4.2.6**) (**GA-12.4.2.7**) (**GA-12.4.2.9**).

Positions have the option to be "deleted" in SVRS; however, this is really an archive process which removes the position from the front-end UI and reporting. (**GA-12.4.2.10**) Current position holders become "Former Representatives" when their term End Date is populated—this can happen due to loss of an election, retirement, or other means of vacating a position. Reason Codes can be added to express if it was due to loss of election, retirement, death, withdrawal, recall, etc. (**GA-12.4.2.11**)

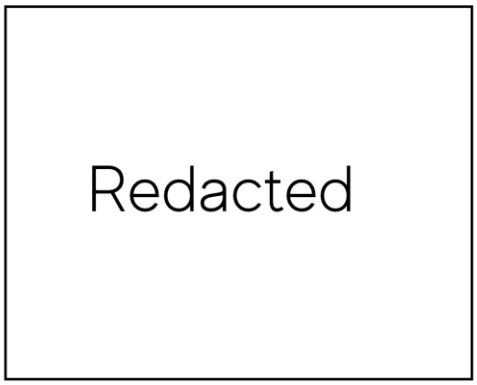


Position Holder details – Current Position holder tab, Former Representatives tab

Candidate allows for image capture; this can be expanded into Position holder—but is not currently a feature of that module; image storage options are expandable into Positions (**GA-12.4.2.8**)

GA-12.4.3 Office Contact Information

In the Jurisdiction details section of System Configuration, users can enter the jurisdiction contact and operating hours. (**GA-12.4.3.1**) (**GA-12.4.3.1.1**) (**GA-12.4.3.7**) (**GA-12.4.3.8**)



Jurisidiction Details (please note, the VoteCal tab would not be visible in VA implementation)

The ability to define contact information for multiple, year-round satellite locations outside of an election event is not currently in SVRS. However, users would be able to use SVRS if the staff and site are provided access via user management and/or Single Sign-On (SSO) integration.

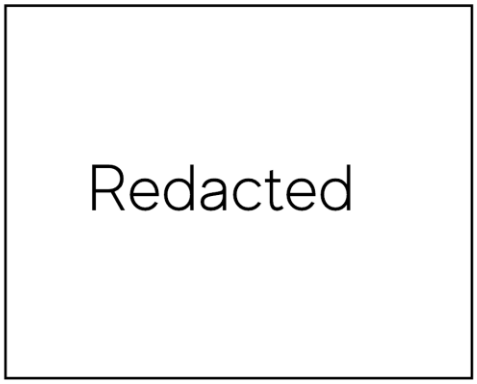
Another variant of a satellite office is a sub-jurisdictional user, who only have access to a particular district within a locality, can be created in SVRS. This allows these users the ability to manage an election and generate poll lists, issue and manage ballots for the assigned district—but keep other parts of the system either blocked or read-only. This is more common in larger jurisdictions for city district special elections (**GA-12.4.3.2**). User information can be extracted from the db (**GA-12.4.3.3**).

Importing would be done during implementation and maintenance would be via SSO or manual user interaction (**GA-12.4.3.4**). Integration with Active Directory (AD) would be a type of import as well.

Election locations can provide services to voters; full or partial services. For more information on Locations in SVRS, see please see 4.3.3 Establish Voting Locations (**GA-12.4.3.5**).

The ability to define contact information for other states is on the SVRS roadmap, this is a System Configuration development item, and is planned to be implemented for the use of contacting other states for any reason that is deemed appropriate for the State of Virginia (**GA-12.4.3.6**)

Labels and envelopes can be printed in SVRS, the specific-business need may need to be configured for State of Virginia (**GA-12.4.3.9**).



System Configuration – User Manager – Add User with Sub-Jurisdictional User Selected

4.3.14.5 GA-12.5 Maintenance, Documentation and Training

GA-12.5.1 Training

SVRS supports the transparent creation of training environments. We create environments in the cloud using scripts. To this automated process we follow with a process of tightening the platform for security using security guidelines/check lists and vulnerability reports. Our Continuous Integration Continuous Deployment (CI/CD) process which includes the creation of Docker images and together with our microservice architecture deployed in Docker Containers can automatically release the application (images and other artifacts) to the newly created training environment by just modifying the configuration of our CI/CD pipelines to point to the proper environment (**GA-12.5.1.1**). The data for the training environment is created using day zero scripts where we configure the training jurisdiction(s), code tables, etc. This is a separate process from the creation of the environment per se. When upgrading training environments, we do not need to refresh the data from the production environment. (**GA-12.5.1.2**).

GA-12.5.2 Documentation

SVRS provides documentation using multiple channels, being the most common printing with options for PDF or actual printing to paper. (**GA-12.5.2.1**) The current state of the system is in the transactional database. This is the single source of information for the distribution channels. The state of the system is replicated in our Data Lake solution in near real time for data analytics and some reporting purposes. (**GA-12.5.2.2**) Our agile Process principles value the product over documentation. But we also understand that we need to comply with the client requirements. We use Microsoft DevOps to manage our requirements in the form of Epics, Features, User Stories and Tasks. Here we also have a place to attach technical and functional designs to implement the features. We use the DevOps Repository to manage our source code. The DevOps pipelines for building and deploying the application are a set of scripts that help automate this process. Beside this we have developed a set of technical and functional documents such as Software Design Document, Application Design Document, Security Test Documents, Technical Infrastructure Design, Step by Step guidelines for the building and deploying environments, Change Control for Releasing (CCR) documents, etc. (**GA-12.5.2.3**)

4.3.14.6 GA-12.6 System

GA-12.6.1 System Design

Our data migration process includes retention of historical data from VERIS and SVRS. This data can be available to users using query tools to view them in their native format. Since we are converting this data to our Data Model, we also provide access to the converted data via our data analytics service or via the application for viewing the current state of the system. (**GA-12.6.1.1**) SVRS Architecture from the design implements configurability as much as possible. We can configure the application functionality, user permissions, code tables data. With this feature we can modify the application behavior without making code changes. Of course, there will be times where a new functionality or a modification to process flows are needed that will require code changes. In this case we follow Agile process to schedule and deliver these modifications incrementally without affecting the application. Only the affected microservice (modules) will be re-deployed in real time. We do not need to re-deploy the entire application just because one module changed. We just deploy the affected components. (**GA-12.6.1.2**)

GA-12.6.2 System Alerts

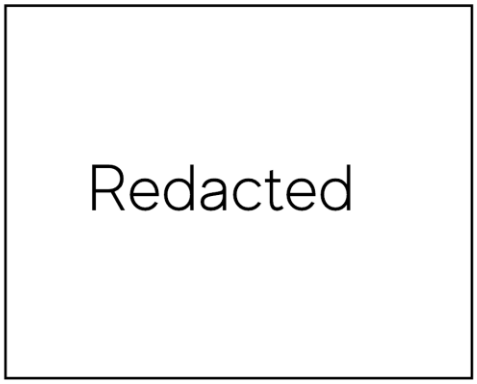
SVRS has standard communication channels. One is the application messages displayed on the user screen letting the user know what the action is to perform, or information about their transactions. The other channel of notifications is for system notifications. Out Infrastructure architecture includes a standard notification via email of any security risks, or performance of the application. Users with the role of administrators or appropriate privileges receive these notifications so they can react to them in a timely manner to avoid issues such as performance degradation, attacks, etc. We also offer a real time monitoring visualization windows help in determining when and what microservice is not performing as expected. (**GA-12.6.2.1**)

4.3.14.7 External Interfaces

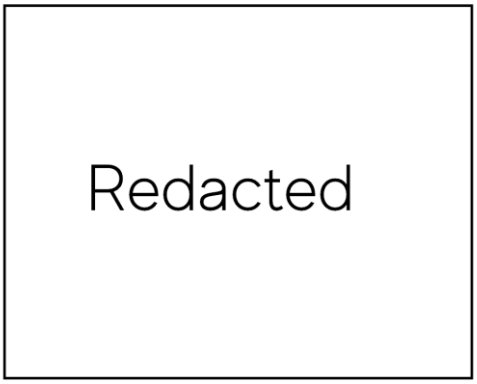
SVRS (Partner Data Integration Service) for External Interfaces implements Enterprise Application Integration (EAI) patterns to interoperate with other systems. SVRS supports data and message protocol mediation. We can send and receive information from different end-point types such as HTTP REST, SOAP, Queues, Database and File Drops (Secure File Transfer). SVRS provides support for data level integration if needed, however application-level integration via EAI messaging patterns and models such as point to point or publish subscribe, synchronous and asynchronous is preferred. Our EAI is a simple interface that implements a series of three steps to consume data from external parties: Input, Process, Output (IPO). These steps are components written as back-end services in .NET core.

The inbound end point implements the input interface to ingest sender information. We have connectors to support REST, SOAP, Queues, and File Drop connectors to facilitate this process. Once the inbound endpoint receives the data it performs archival, structural and format validations. Then the Inbound passes the data to the Processor service which implements the Process interface. The Processor implements internal EAI design pattern such as canonical message, data transformations, routing, data enrichment, message splitting, and protocol mediation. Sometimes the Processor is a pass-through service that does not require to do any processing to the data (payload). The processed data is sent to the destination via the Outbound endpoint. The destination can be a Microservice, another external third party or our Data Lake where Business Process services consume the data. For example, when the Department of Corrections sends the Felon voter file, they drop it in our secure inbound directory. Our Inbound end point will archive a copy of the file. The Processor reads the file, validates for structure and content, and stores the records that error out for a user to do further analysis on it. The Processor uses the Outbound connector to the Data Lake to send the new transformed Felon data for persistence in a table. The Felon Intel Matching Agent Service triggers and performs the matching. A post matching process tags the matched records in the database. SVRS can interoperate with legacy and modern applications. SVRS can also send data to external parties using the same EAI IPO steps mentioned above. The inbound and outbound messaging is configurable, and the users can manage the different Interfaces properties such as integration name, payload type, protocol, destination. This is a configurable solution for integration, and we use it for transferring and load testing of files into external systems or devices if they offer integration end points such as OCR, ballot sorters, etc.

The following image shows how out IPO pattern maps to a specific integration. An external Party drops or SFTP a data file (Fixed Length, or csv) into our inbound end point. The data is parsed and enters structural and format validations. After this the data is transformed or mapped into an output data format (JSON) using the appropriate processor. Finally, be sent via a Rest request to a SVRS Microservice for processing (Matching, etc.).

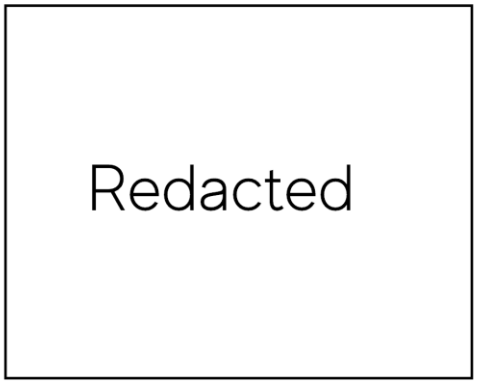


The Partner Data Integration Data Formats and Protocols are shown in the next image:



Protocols:

Data Formats:

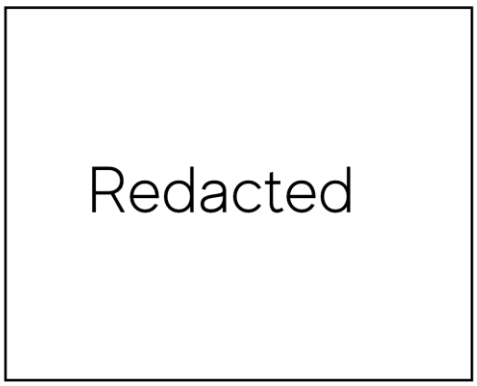


Internally, our application integrates with other components by way of messaging with other microservices or external services such as GIS. We use RabbitMQ, an industry proven Messaging software to implement the inter microservice communication

EI-13.1 Database Access

SVRS data layer architecture includes a Transactional Relational Database: Microsoft SQL Server Database for storing the transactional records, therefore keeping the current state of the system. We can access this database from the application views or using industry standard tools such as Microsoft SQL Server Management Studio (SSMS). We replicate this database in our data repository (Data Lake) to avoid performance issues with the application that uses the Transactional database. We strongly recommend not to query directly the Transactional database, instead fetch or extract data from the replica database or any of the pother artifacts in the data lake. The user can use the tool of his preference (PowerBI, ******, etc.) to access the data from the Data Lake, making this very flexible and tool independent solution. This read only data lake provides the users with reporting and data analytics features such as Voter 360, in which we can get answers to questions such as what happened to this Voter after the las election? In other words, we view the state of the system at any time in point since the implementation of the Solution. (EI-13.1.1).

SVRS uses industry standards for data encryption. We encrypt data a rest and in transit. We encrypt data at rest using Transparent Data Encryption (TDE) with in MS SQL Server database. TDE encrypts the storage of an entire database by using a symmetric key called the database encryption key. The figure below shows the architecture for TDE. Our Security Microservice encrypts the passwords using sha512 as specified in FIPS 180-4 and NIST SHS standards (SHA-2 type). (EI-13.1.2)



Any user with the right privileges can perform ad-hoc queries, preferable in the data lake. The results can be rendered as CSV, Excel, JSAON or any standard format. (EI-13.1.3)

EI-13.2 Data Import

As mentioned above, SVRS ingests file imports in pre-determined formats, and we can configure it to support files from (given their file format and structure):

1. Bureau of Vital Statistics (BVS)
2. Courts: The U.S. District Courts, local Circuit Courts, other states' District/Circuit Courts
3. Federal Courts
4. United States Postal Service (USPS)
5. Virginia State Police (VSP)
6. Department of Motor Vehicles (DMV)

(EI-13.2.1)

SVRS also can integrate to import files from external systems, programs or devices such as:

1. Desktop Application for General Registrars (DAGR)
2. Pollbook
3. Excel and CSV

(EI-13.2.2)

Our External Integration Service puts little logic in the endpoint. This logic is for archiving input files to preserve in their original them for audits, data format and type validations such as date formats, numeric fields, enumerations, etc. It also performs structural validation to make sure the data formats are well formed such as XML, JSON or Fixed Length, CSV (EI-13.2.3).

By default, the External Integration service tries to ingest all the data from the third parties. If the third party allows for chunking the input data, then we can support message correlation process to determine how many chunks the original data contains and wait until all chunks are received and the re-create the original data with all the chunks. (EI-13.2.4).

EI-13.3 Data Export

SVRS also can export files to external systems, programs or devices such as:

1. Election Night Reporting System
2. Pollbook
3. Excel and CSV

All this data formats have to be configured or implemented during implementation time (EI-13.3.1).
SVRS supports fixed length file formats and can be configured during implementation to support formats from:

1. Fairfax Service
2. Desktop Application for General Registrars (DAGR)
3. Custom Membership Administration Portal (MAP) (EI-13.3.2)

Using our Job Scheduler, we can configure a job to trigger a job to execute or call an API to perform file extracts. Then the Outbound Interface of the External Interfaces services sends the file to a pre-determined configurable location (EI-13.3.3).

The Job Scheduler is configurable and can pass values to the process that performs the data file extract. We will expand this to add a setting to configure the amount of data that can be extracted from the system in a single data extraction process (EI-13.3.3.1).

Given the great expandability and configurability of SVRS, our External Interfaces Microservice can be set to support Election Results Common Data Format Specification Version 1.0 as defined by NIST Special Publication 1500-100 (EI-13.3.1).

EI-13.4 API Integration

As mentioned in the introduction to this main section (4.3.14) our External Interfaces Microservice provides Inbound API interface supporting multiple protocols such as REST API, SOAP, Queues and SFTP. (EI-13.4.1).

Data integration with ERIC via an external API can be configured in our External Interfaces Microservice, as any other external third-party system (EI-13.4.2).

We can integrate via API to other systems as well. Some of these systems are for example:

1. Citizen Portal
2. Department of Motor Vehicles (DMV)
3. Smarty Streets

This is a process that can be done during implementation or prior to it once we know the data format and the external party API protocols (EI-13.4.3).

The same way we can integrate for data ingestion via API, the External Interface Microservice can be configured to integrate via external third-party APIs to export data. Some of these third-party systems could be:

1. Enhanced Voting/My Ballot
2. Democracy Works/Ballot Scout
3. Committee Electronic Tracking (COMET)
4. Secure Access Portal (SAP)
5. Client Services Interface
6. Voting Information Project (VIP)
7. Voter Photo Id (VOPhoID)
8. Soch, Inc. (Integra)

(EI-13.4.4)

EI-13.5 Data Type

Our External Interfaces Microservices supports ingestion in UTF-8 for character-based information and BLOB for binary data (images, etc.) (EI-13.5.1).

EI-13.5.2 Department of Motor Vehicles (DMV) Felon and Mentally Incapacitated Questions

During implementation we will develop an external interface process to answer questions from the DMV for Felons and Mentally Incapacitated regardless if previously answered (EI-13.5.2).

4.3.15 Maintain Geographic Data

We understand the adoption of GIS is not uniform throughout the state, so SVRS offers two solutions: SVRS GIS and SVRS Geo. SVRS GIS is available for localities that are GIS ready, otherwise locality can opt to use SVRS Geo. SVRS Geo is a friendly option for smaller jurisdictions where the cost of organizational change to adopt GIS outweigh the benefits.

4.3.15.1 Plan Management

SVRS GIS Service provides Production and Staging environment to manage current (officially published) and proposed plan types separately. Staff creates a proposed plan using the copy of current plan. When the proposed plan is ready to be published, the staff ******s the plan from Staging to Production. During the migration, the solution creates an archive of the current plan and merges the proposed with the current plan and published the results. The historical plans are associated with a user defined names and plan effective dates; Staff uses these attributes to load these plans into Map view as needed in the future. Staff may refresh the current plan with a most recent historical plan if issues were found. Authorized Staff removes any unused geographical plans. (MGD-14.1.1, MGD-14.1.2)

The solution provides a Role Based Access Control, the user role types are as follows (MGD-14.1.2.1):

- Viewer - View items such as maps.
- Data Editor - Viewer privileges plus the ability to edit features shared by other users.
- Publisher - User privileges plus the ability to publish features and map tiles as hosted web layers.
- Administrator - Publisher privileges plus privileges to manage the organization and other users.

GIS Service offers Staff the ability to bulk modify address, precinct, and district plans with the use of ****** Data (import), ****** Editor, and ****** ******s. (MGD-14.1.3)

As precinct, precinct-splits, and district plans are changed, the GIS service performs spatial analysis to determine the addresses that are impacted by the change. The results of the spatial analysis are loaded into an impact analysis table. The impact analysis table is merged with the voter residential address data to determine the impacted voters. (MGD-14.1.4)

The changes to the current plans are logged into the Journal Table and retained per Virginia policies. (MGD-14.1.5).

4.3.15.2 Maintain Address

Authorized Staff can import using ****** Data ******, as shown in Figure 52: ****** Data below, a wide variety of GIS formats such as but not limited to: Shapefile (SHP), Keyhole Markup Language (KML), File Geodatabase (GDB) or GeoPackage, Layers (LYR), OpenStreetMap (OSM), and ArcGIS. (MGD-14.2.2).

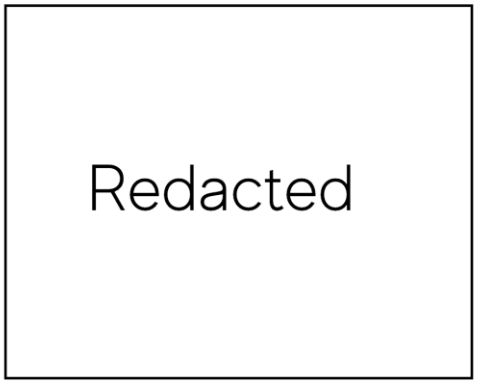


*Figure 52: ****** Data*

As shown in Figure 53: ****** Data Map Columns below, ****** Data provides Staff to map the incoming data fields to the SVRS standardized address format (MGD-14.2.9). SVRS solution validates the incoming plan meets the data validations rules such as data type, field length, mandatory, and so on (MGD-14.2.10).

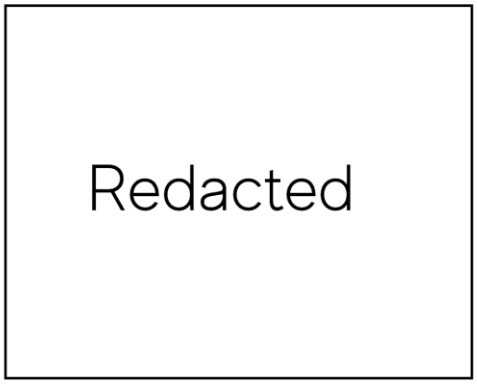


*Figure 53: ****** Data Map Columns*

GIS Service provides on the map dynamic search (similar to google address search) to locate address within county (MGD-14.2.1). As shown in Figure 54: GIS Update Address below, user can select and manually adjust the geocoordinates of an address using the map layers as the reference. (MGD-14.2.3, MGD-14.2.4, MGD-14.2.5, MGD-14.2.6).

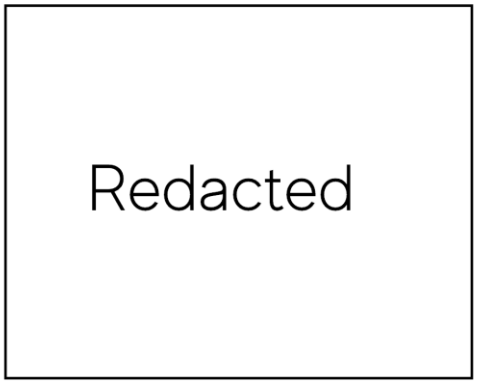


Figure 54: GIS Update Address

GIS Service uses ArcGIS World Geocoding Service to receive metadata about the address including date of the geocode, the geocode service used, the data set used to geocode the address, and the level of accuracy. Team Canton can preserve the metadata in auxiliary table and associate it the address (MGD-14.2.8).

GIS Service's Layers List ******, as shown in Figure 55: GIS Layer List below, allows Staff to toggle the viewing of the shapefile layers and render the draw order of the maps. The Layers List ****** also allows to overlap the state and locality level shapefiles by adjusting their visibility. (MGD-14.2.7, MGD-14.2.7.1)

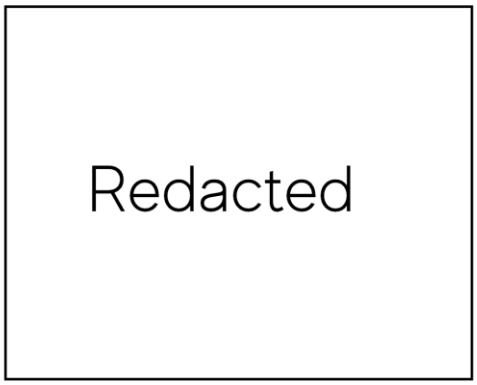


Figure 55: GIS Layer List

As shown in Figure 56: GIS Address Meta Data below, SVRS solution supports tracking metadata zoning information to define ineligible addresses (e.g., government addresses or commercial addresses), other elements such as Mailing Address Required (e.g., PO Box) or to define that a Unit No. is needed for an address. Temporary Housing is not currently included but it can be very easily added. (MGD-14.2.12).

Currently, SVRS solution captures the when and why mail was last returned on the voter record however Team Canton can capture this additionally on the residential address as well (MGD-14.2.12.1).

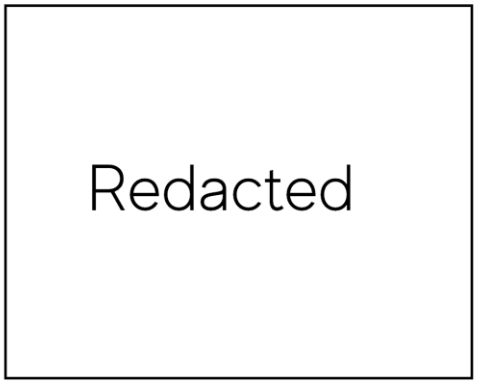


Figure 56: GIS Address Meta Data

As shown in figures below locality using SVRS Geo Service can complete address maintaining activities without GIS. Staff can manually associate street alias to address (MGD-14.2.11).

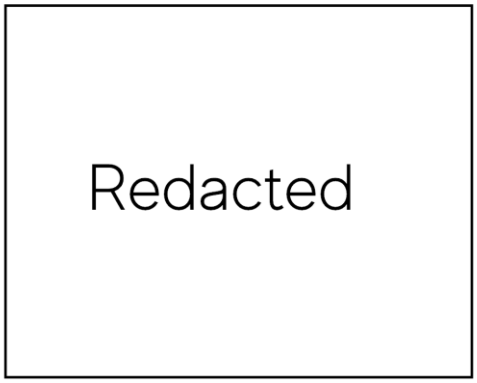


Figure 57: Geo Address Search

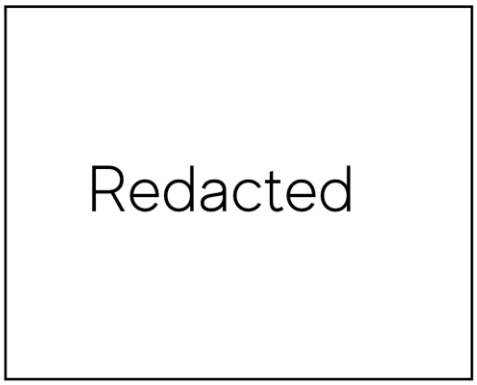


Figure 58: Create Geo Address

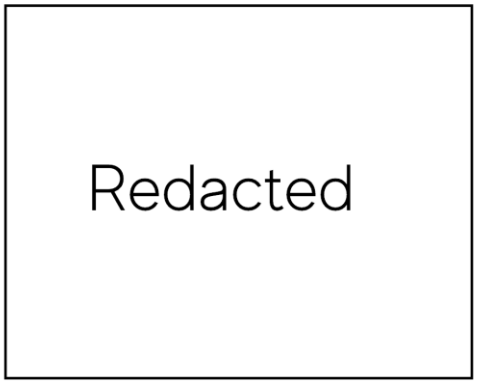


Figure 59: Geo Address Detail

4.3.15.3 Maintain District Boundaries

SVRS GIS Service using the ****** Data ******, ELECT and Locality can import district boundaries from external agencies. ****** Data ****** allows Staff to map the incoming file to SVRS standard columns. SVRS validates the incoming file meets the data validation specification for required, field data types, and so on. (MDG-14.3.2).

As shown in Figure 60: Click Start to Create District, Staff can also create district boundaries for the standard district types. (MGD-14.3.2).

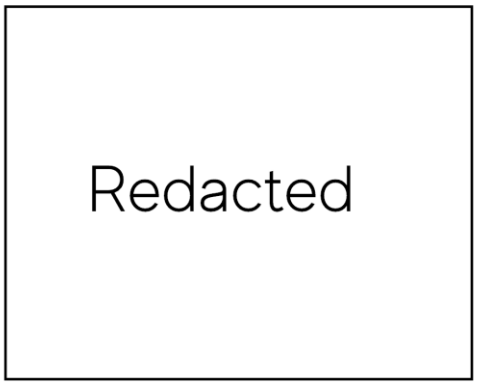


Figure 60: Click Start to Create District

As shown in Figure 61: ******, Staff can adjust the district boundaries using the ****** Editor. (MGD-14.3.1).

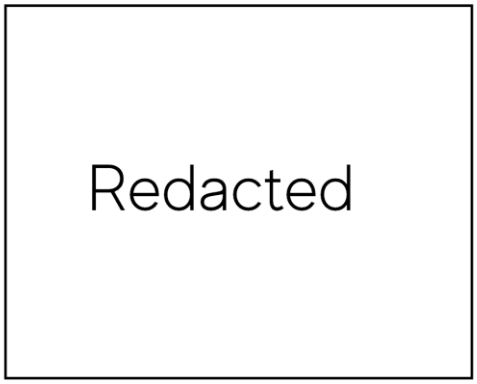


*Figure 61: ******

Staff can attach documentation (e.g., image and pdf) to the district boundaries, as shown in Figure 62: Attach Image to ShapeFile below. (MGD-14.3.3).

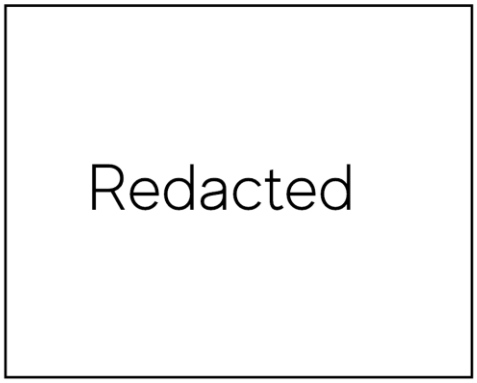


Figure 62: Attach Image to ShapeFile

As shown in Figure 63: Look up District Associations below, Staff can look up precinct and precinct-splits associated with the district boundary.

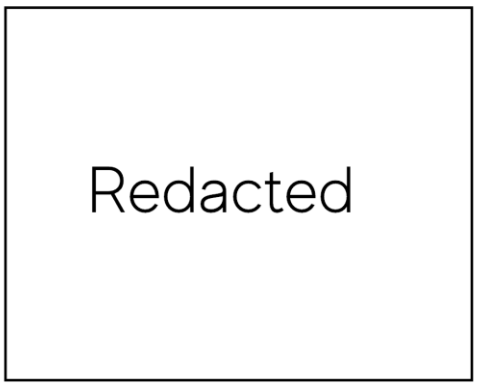


Figure 63: Look up District Associations

SVRS GIS Service using the ****** ****** supports configuration of ****** for districts such as districts of same type cannot overlap boundaries (MGD-14.3.4)

Districts created within GIS service is automatically synced with Geo service if locality uses GIS. Otherwise, Locality can manually manage the district boundaries just with Geo service. As shown in Figure 64: Geo District Search and Figure 65: Geo District Details below, Staff can search, manage, view history, and capture contact information. (MGD-14.3.2).

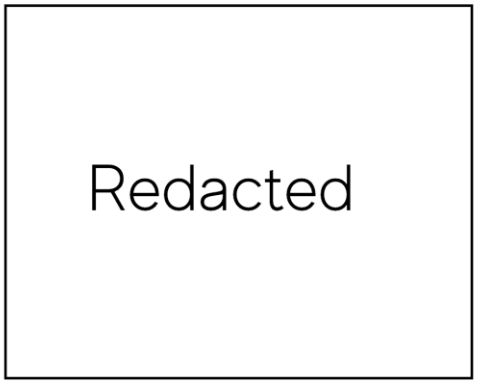


Figure 64: Geo District Search

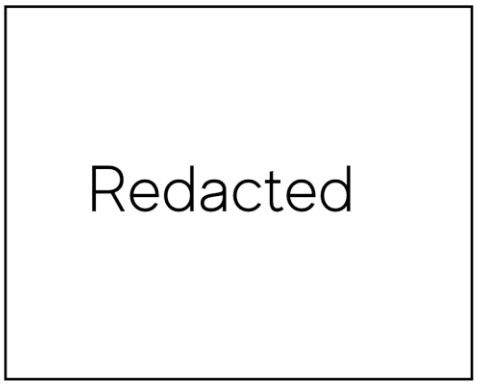


Figure 65: Geo District Details

4.3.15.4 Maintain Precinct Boundaries

SVRS GIS Service using the ****** Data ******, Locality can import precinct boundaries. ****** Data ****** allows staff to map the incoming file to SVRS standard columns. SVRS validates the incoming file meets the data validation specification for required, field data types, and so on. (MDG-14.4.2).

As shown in Figure 66: GIS Create Precinct, Staff can also create county, city, and town precinct boundaries. (MDG-14.4.2, MDG-14.4.5).

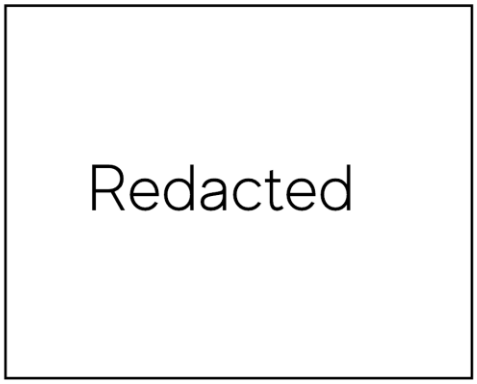


Figure 66: GIS Create Precinct

As shown in Figure 61: ******, Staff can adjust the precinct boundaries using the ****** Editor. (MDG-14.4.2).

Staff can attach documentation (e.g., image and pdf) to the district boundaries, as shown in Figure 62: Attach Image to ShapeFile below. (MGD-14.4.3).

As shown in Figure 67: Precinct Associations below, staff can look up district and precinct-splits associated with the precinct boundary. (MDG-14.4.6).

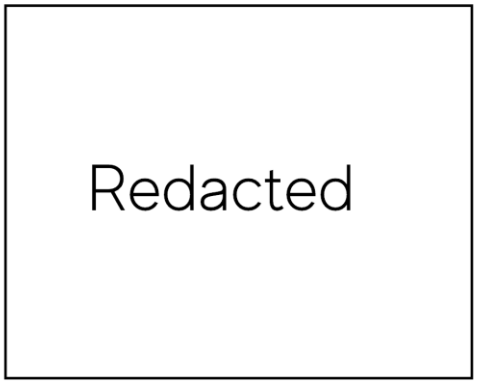


Figure 67: Precinct Associations

SVRS GIS Service using the ****** ****** supports configuration of ****** and demographic rules for precincts as shown below (MDG-14.4.4) -

- Precincts cannot overlap boundaries.
- Precincts has no area that is not covered by a precinct-split.
- Each precinct is wholly contained within a single congressional district, Senate district, and House of Delegates district. (MDG-14.4.8)
- Each precinct is wholly contained within an election district used for the election of one or more members of the governing body or school board for the county or city. (MDG-14.4.8)
- A precinct contains no less than ‘X’ and no more than ‘Y’ registered voters; X and Y are defined at locality. (MGD-14.4.9.1)

SVRS solutions logs the error (violation) of the ****** and demographic rules, the staff can review and fix the issues with boundaries. Locality can ignore the errors to accommodate for exceptions such as minimum population of registered voters for a precinct if the creation of a larger district would result in a precinct-split between the governing body and school board of the county or city, House of Delegates, state Senate, and United States House of Representatives.

Currently, there is not explicit restriction on changes to precinct boundaries however for the State of Virginia Team Canton can add option for locality administrator to set blackout periods for changes to precinct boundaries. (MDG-14.4.7)

Precincts created within GIS service is automatically synced with Geo service if locality uses GIS. Otherwise, locality can manually manage the precinct boundaries just with Geo service. As shown in Figure 68: Geo Precinct Search and Figure 69: Geo Precinct Details below, staff can search, manage, view district, precinct-split association, view history, and capture contact information.

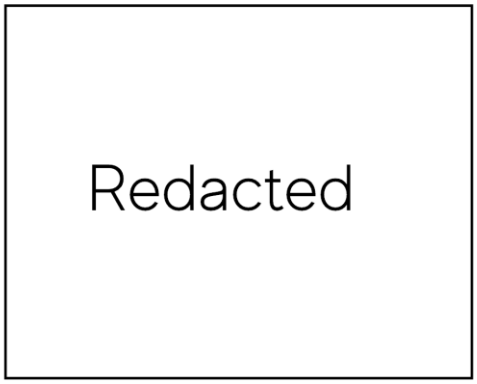


Figure 68: Geo Precinct Search

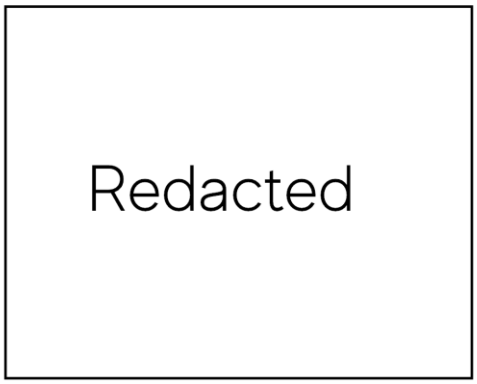


Figure 69: Geo Precinct Details

SVRS solution provides auto-precinct service that standardizes the incoming residential address, determines the geocoordinates, and assigns precinct and precinct-splits based on geocoordinates (MGD-14.4.1). If auto-precinct service, failed to standardize or assign precinct-splits with high-level of confidence, the voter record is marked as 'Pending' with reason of pending precinct. Staff works the queue and manually assigns precinct and precinct-split (MGD-14.4.1.1).

4.3.15.5 Maintain Precinct-Splits

SVRS GIS Service using the ****** Data ******, Locality can import precinct-split boundaries. ****** Data ****** allows Staff to map the incoming file to SVRS standard columns. SVRS validates the incoming file meets the data validation specification for required, field data types, and so on. (MDG-14.5.1.1).

As shown in Figure 70: GIS Create Precinct Portion, Staff can also create precinct-split boundaries by overlaying qualifying districts on existing precinct shapes. (MDG-14.5.1).

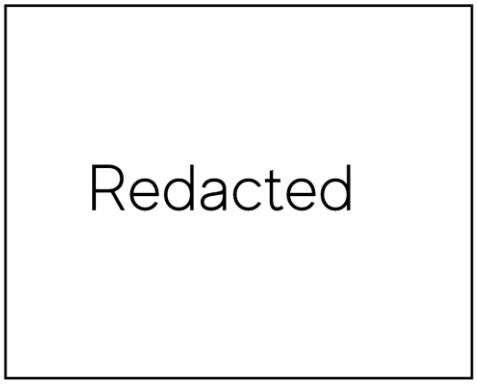


Figure 70: GIS Create Precinct Portion

Precinct-splits created within GIS service is automatically synced with Geo service if locality uses GIS. Otherwise, Locality can manually manage the precinct-split boundaries just with Geo service. As shown in Figure 71: Geo Precinct Portion Search and Figure 72: Geo Portion Details below, Staff can search, associate precincts and districts, and view history.

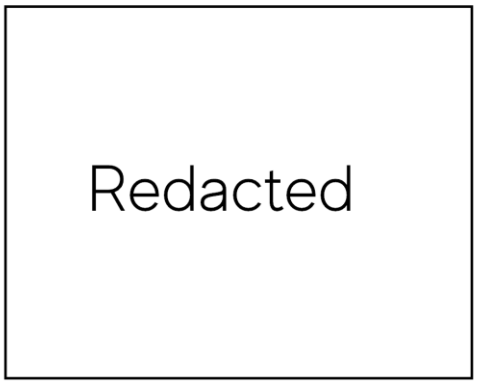


Figure 71: Geo Precinct Portion Search

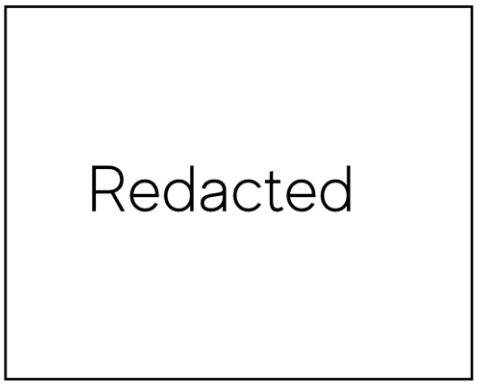


Figure 72: Geo Portion Details

4.3.15.6 Index Addresses to Precinct-splits

SVRS GIS Service conducts spatial analysis of precinct, precinct-splits, and addresses features to associate the address to precinct and precinct-splits. As shown in Figure 73: GIS Address Associations below, each address displays the precinct and precincts-splits associated to it. This process is automatic in GIS Service. (MGD-14.6.)

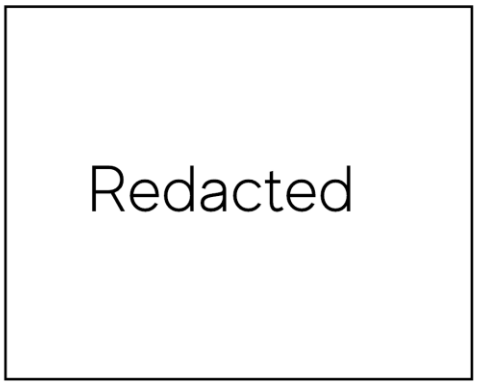


Figure 73: GIS Address Associations

As shown in Figure 74: Geo Address Associations below, SVRS Geo Services allows staff to manually associate single addresses to precinct and precinct-splits (MGD-14.6.1.1). For the State of Virginia, Team Canton adds the option to select a group of addresses from address search and assign them to precinct and precinct-splits (MGD-14.6.1.2).

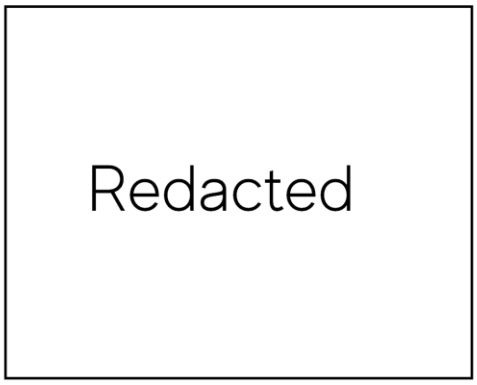


Figure 74: Geo Address Associations

4.3.15.7 Reporting

In addition to the Data Analytics ad hoc reporting engine, standardized reports not limited to the following can be created for the State of Virginia within the Report module –

- Precinct, Locality and District Reporting
- Voter Change in Precinct Exception Report
- Voter Change in Precinct Report
- Locality Legislative Breakdown Report
- Locality Redistricting Report

4.3.15.8 Correspondence

SVRS GIS Service conducts an impact analysis of addresses thus voters what are affected by changes to precinct-splits, precincts, and districts. The results of the impact analysis are logged for review by Staff. Staff can optionally queue the affected voter records to be sent Voter Notification Card (VNC). Notice Service uses the combination of geocoordinates, house suffix, house prefix, Unit/Apt # and so on attributes to complete the householding process thus only one notice is sent per household. (MGD-14.8.1, MGD-14.8.2)

As shown in Figure 75: Send VNC for District Changes here below is an example of how SVRS Geo Service provides the prompt for the Staff to optionally select to send VNC for the affected voters.

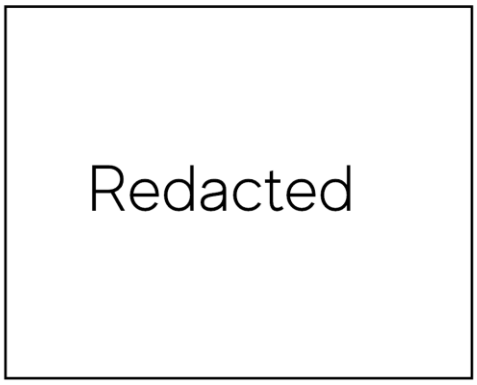


Figure 75: Send VNC for District Changes