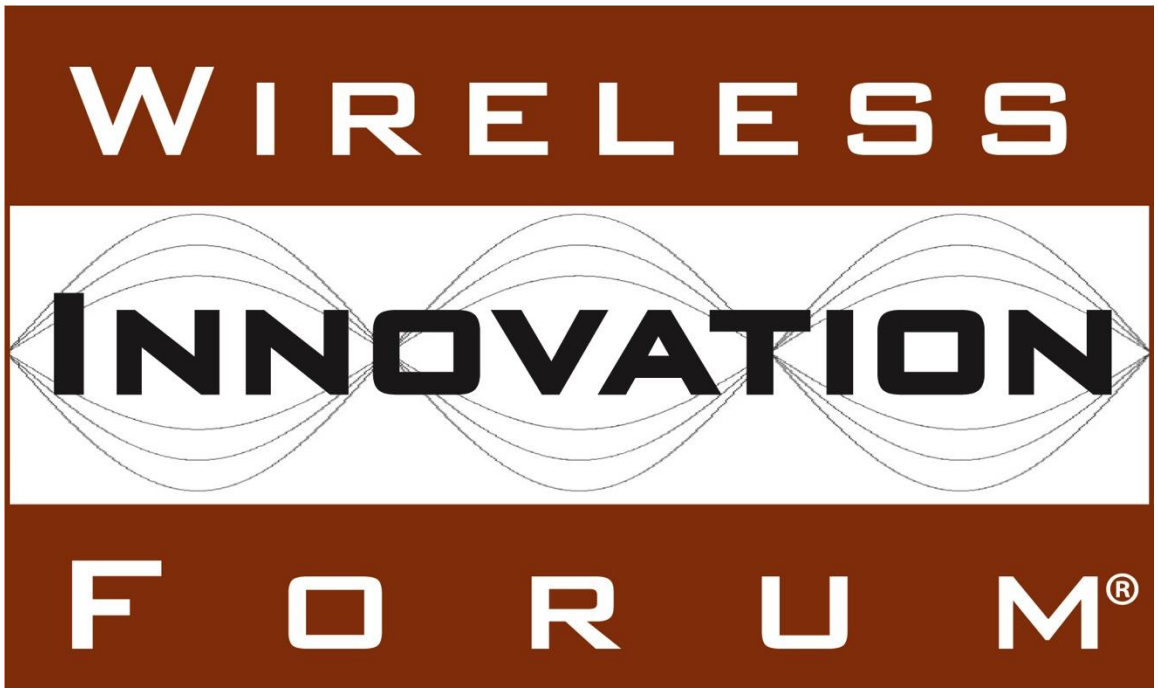


# Interim SAS to SAS Protocol Technical Report-B

## Historical Document WINNF-16-H-0003

Version V1.0.0
3 January 2017

Copyright © 2017 The Software Defined Radio Forum Inc. – All Rights Reserved

# TERMS, CONDITIONS & NOTICES

This document has been prepared by Working Group 3 of the Spectrum Sharing Committee to assist The Software Defined Radio Forum Inc. (or its successors or assigns, hereafter "the Forum"). It may be amended or withdrawn at a later time and it is not binding on any member of the Forum or of the Spectrum Sharing Committee.

Contributors to this document that have submitted copyrighted materials (the Submission) to the Forum for use in this document retain copyright ownership of their original work, while at the same time granting the Forum a non-exclusive, irrevocable, worldwide, perpetual, royalty-free license under the Submitter's copyrights in the Submission to reproduce, distribute, publish, display, perform, and create derivative works of the Submission based on that original work for the purpose of developing this document under the Forum's own copyright.

Permission is granted to the Forum's participants to copy any portion of this document for legitimate purposes of the Forum. Copying for monetary gain or for other non-Forum related purposes is prohibited.

The Forum draws attention to the fact that it is claimed that compliance with this specification may involve the use of a patent ("IPR") concerning "Interim SAS to CBSD Protocol Technical Report-A", Document WINNF-15-P-0023, Version V1.0.0, 2 November 2015, referenced in Section 2.1.The Forum takes no position concerning the evidence, validity or scope of this IPR.

The holder of this IPR has assured the Forum that it is willing to license all IPR it owns and any third party IPR it has the right to sublicense which might be infringed by any implementation of this specification to the Forum and those licensees (members and non-members alike) desiring to implement this specification. Information may be obtained from:

QUALCOMM Incorporate
Attn: Thomas Rouse
5775 Morehouse Drive
San Diego, California, 92121
Email: ip.disclosure@qualcomm.com

Attention is also drawn to the possibility that the Forum shall not be responsible for identifying any or all such IPR.

THIS DOCUMENT IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY USE OF THIS SPECIFICATION SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN

Copyright © 2017 The Software Defined Radio Forum Inc
All Rights Reserved

RISK, AND NEITHER THE FORUM, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER, DIRECTLY OR INDIRECTLY, ARISING FROM THE USE OF THIS DOCUMENT.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the specification set forth in this document, and to provide supporting documentation.

This document was developed following the Forum's policy on restricted or controlled information (Policy 009) to ensure that that the document can be shared openly with other member organizations around the world. Additional Information on this policy can be found here: http://www.wirelessinnovation.org/page/Policies_and_Procedures

Although this document contains no restricted or controlled information, the specific implementation of concepts contain herein may be controlled under the laws of the country of origin for that implementation. Readers are encouraged, therefore, to consult with a cognizant authority prior to any further development.

Wireless Innovation Forum ™ and SDR Forum ™ are trademarks of the Software Defined Radio Forum Inc.

Copyright © 2017 The Software Defined Radio Forum Inc
All Rights Reserved

# Table of Contents



Copyright © 2017 The Software Defined Radio Forum Inc
All Rights Reserved

# List of Figures



Copyright © 2017 The Software Defined Radio Forum Inc
All Rights Reserved

# Contributors

The following individuals made significant contributions to this document:

Editors: James Ni (Federated Wireless), Greg Billock (Google)

Other Member Representatives:

- Alcatel-Lucent: Milind Buddhikot
- AT&T: Neeti Tandon
- Ericsson: Kumar Balachandran, Mark Racek
- Federated Wireless: Masoud Olfat
- Google: Greg Billock
- Key Bridge Global: Jesse Caulfield
- Motorola Solutions: David Gurney
- NIST: Anirudha Sahoo
- Nokia Networks: Anatoly Andrianov, Al Hirsbrunner, Steve Magee, Prakash Moorut
- Qualcomm: Doug Knisely
- Verizon: Max Solondz, Naseem Khan

Copyright © 2017 The Software Defined Radio Forum Inc
All Rights Reserved

# SAS to SAS Exchange Protocol

## 1 Introduction

This document is a Technical Report describing the protocol by which Spectrum Access Systems (SAS) exchange information as required by the FCC Report and Order 15-47 in order to facilitate access to the band by Citizen's Broadband Radio Service Devices (CBSDs) connected to other SASs, coordinate operations between and among such CBSDs, provide a stable radio frequency environment for Priority Access License (PAL) licensees, and other functions required for orderly spectrum administration and the fulfillment by the SAS of the responsibilities given it under Part 96.

## 2 References

### 2.1 Normative references

[N.1]. "Interim SAS to SAS Protocol Technical Report-A", Wireless Innovation Forum Working Document WINNF-15-P-0051.

[N.2]. "The GeoJSON Format Specification 1.0", 16 June 2008.

[N.3]. "SAS to CBSD Protocol Technical Report-B", Wireless Innovation Forum Working Document WINNF-15-P-0062.

### 2.2 Informative references

[I.1]. "SAS Functional Architecture", Working Document WINNF-15-P-0047, Version V0.3.6, 12 June 2015

[I.2]. "Report and Order and Second Further Notice of Proposed Rulemaking", FCC, FCC 15-47, April 17, 2015.

[I.3]. "CBRS Communications Security Technical Report", Wireless Innovation Forum WINNF-15-P-0065.

[I.4]. "The application/json Media Type for JavaScript Object Notation (JSON)" RFC 4627.

[I.5]. "Interim SAS to CBSD Protocol Technical Report-A", Wireless Innovation Forum Working Document WINNF-15-P-0023.

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

# 3 Abbreviations

| Abbreviation | Description |
|---|---|
| CBSD | *Citizens Broadband Radio Service Device* |
| ESC | Environmental Sensing Capability |
| FCC | Federal Communications Commission |
| FRN | FCC Registration Number |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | Secure HTTP (e.g. with TLS) |
| ID | Identifier |
| JSON | Javascript Object Notation |
| PAL | Priority Access License |
| SAS | Spectrum Access System |
| TLS | Transport Layer Security |
| URL | Universal Resource Locator |
| UTC | Coordinated Universal Time |

# 4 SAS-SAS Exchange Message Encoding and Transport

This section specifies the message encoding and transport methods to be used for all messages specified in section 5 and 6, as well as globally applicable error condition indications and handling.

## 4.1 Message Encoding

SAS to SAS exchange messages are to be encoded using JSON (JavaScript Object Notation), a widely accepted network data representation and exchange protocol. JSON encoded messages are in the form of human readable single or multiple of “name”/“value” pair(s), where the “name” is typically a self-descriptive unique string tag identifying the information to be represented and the “value” is the information to be carried. JSON allows a “value” be one of the four primitive types (respectively, String, Number, Boolean and Null) and two structured types (JSON object and array). Brief definitions of all above primitive and structured data types are as follows:

- String: a readable string begins and ends with double quotation marks. All Unicode characters can be placed within the quotation marks. Double quotation mark, backslash and control characters must be proceeded by an escape character (\ i.e. backslash).

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

- Number: a signed integer or a floating number. A floating number can be represented using regular decimal format or scientific exponential format.
- Boolean: a Boolean value this indicates either true or false.
- Null: a null value representation of the value of a string or an object or an array type when the required value is not available.
- Object: A self-contained JSON object enclosed in a pair of curly brackets.
- Array: a value array of any one of the String, Number, Boolean, object and array types, enclosed by square brackets.

A JSON representation of the SAS Administration record, for example, is as follows:

```
{
    “ID” : “admin/sas/elgoogSasAdministrator1”,
    “name” : “Elgoog Inc. SAS Administrator No.1”,
    “contactInformation” : [
        {
            “name” :  “John Dole”,
            “phone” : “(202)123-5678”,
            “email” : “john.dole@elgoog.com”
        },
        {
            “name” :  “Lisa Dole”,
            “phone” : “(202)123-5679”,
            “email” : “lisa.dole@elgoog.com”
        }
    ],
    “fccInformation” : {
        “certificationDate” : [
            “12-30-2016 11:10:11 EST”,
            “05-30-2016 11:00:00 EST”,
            “01-30-2016 11:25:23 EST”
        ]
    }
}
```

## 4.2 Message Transport

For secure and convenient SAS to SAS information exchange, two essential requirements must be met at the SAS to SAS transport layer. First of all, it must guarantee the transport security requirement as addressed in the WG2 security architecture. Secondly, it needs to satisfy a general purpose requirement that allows broad applications and easy adaption. HTTP over Transport Layer Security (TLS) (HTTPS) is an ideal option that satisfies both the two requirements.

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

# 5 SAS-SAS Exchange Message Definitions

This section will contain detailed field-by-field specifications for the SAS to SAS messages for each of the following exchange record types.

## 5.1 SAS Administrator record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: admin/sas/$ADMINISTRATOR<br>• **$ADMINISTRATOR**: SAS-CA certified unique SAS administrator identifier |
| name | string | Human-readable local significant string |
| contactInformation | object: ContactInformation | Contains various contact information |
| fccInformation | object: FCCInformation | Contains the FCC certification information |

## 5.2 ESC Administrator record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: admin/esc/$ADMINISTRATOR<br>• **$ADMINISTRATOR**: SAS-CA certified unique ESC administrator identifier |
| name | string | Human-readable local significant string |
| contactInformation | object: ContactInformation | Contains various contact information |
| fccInformation | object: FCCInformation | Contains the FCC certification information |

FCCInformation object:

| Field | Data Type | Field Definition |
|---|---|---|
| certificationId | string | The FCC-issued certification ID |
| certificationDate | string | Date of certification, in format YYYY-MM-DD |
| certificationExpiration | string | Date of certification expiration, in format YYYY-MM-DD |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| certificationConditions | string | Human-readable string or reference annotating the certification |
| FRN | string | The FRN of the certified entity |
| sasPhase | string | If this is a SAS information object, defines the Phase ("1" or "2") of certification. |
| deviceFccId | string | For a device certification, the FCC ID of the device. |

ContactInformation object:

This object should be of type jCard (See RFC 7095). (Alternate format: schema.org Person type?)

- Contact name: Human-readable string
- Primary phone number: telephone number string
- Secondary phone number: telephone number string
- Email: email address string
- Alternative email: email address string
- Address: Human-readable address string
- Notes: any descriptive notes
- Should be able to represent multiple contact points and/or methods

### 5.3 SAS Implementation record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: sas/$ADMINISTRATOR/$IMPLEMENTATION<br>• **$ADMINISTRATOR**: SAS-CA certified unique SAS administrator identifier<br>• **$IMPLEMENTATION**: SAS-CA certified unique SAS implementation instance identifier |
| name | string | Human-readable local significant string |
| administrator | string | • **Reference**: SAS Administrator ID |
| contactInformation | object: ContactInformation | Contains various contact information |
| publicKey | string | • **Format**: X.509 key |
| fccInformation | object: FCCInformation | Contains the FCC certification information |
| escId | array of string | Array of ESC Implementation IDs<br>• **Reference**: each name entry is in references to an ESC implementation instance |
| url | string | • **Format**: public URL |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| sasToSasUrl | string | • **Format**: Machine-addressable URL |
| sasToCbsdUrl | string | • **Format**: Machine-addressable URL |

### 5.4 ESC Implementation record

| Field | Date Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: esc/$ADMINISTRATOR/$IMPLEMENTATION<br>• **$ADMINISTRATOR**: SAS-CA certified unique SAS administrator identifier<br>• **$IMPLEMENTATION**: SAS-CA certified unique ESC implementation instance identifier |
| name | string | Human-readable local significant string |
| administrator | string | • **Reference**: ESC Administrator ID |
| contactInformation | object: ContactInformation | Contains various contact information |
| publicKey | string | • **Format**: X.509 key |
| fccInformation | object: FCCInformation | Contains the FCC certification information |

### 5.5 Domain Proxy record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: domain/$DOMAIN_PROXY<br>• **$DOMAIN_PROXY**: SAS-CA certified unique Domain Proxy implementation instance identifier |
| name | string | Human-readable local significant string |
| contactInformation | object: ContactInformation | Contains various contact information |
| publicKey | string | • **Format**: X.509 key |
| fccInformation | object: FCCInformation | structured object contains the FCC certification information |

### 5.6 CBSD Device Type record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: cbsd-type/$FCC_ID<br>• **$FCC_ID**: the FCC ID assigned to the device type in the FCC equipment authorization process |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| name | string | Human-readable local significant string, e.g. model number |
| manufacturer | string | Human-readable string. The device manufacturer. |
| contactInformation | object: ContactInformation | Contains various contact information |
| fccInformation | object: FCCInformation | Contains the FCC certification information |
| deviceCharacteristics | object: DeviceCharacteristics | Device parameters for the device type. |

DeviceCharacteristics object:

| Field | Data Type | Field Definition |
|---|---|---|
| airInterface | object: AirInterface (See SAS-CBSD spec) | Air Interface definition of this device |
| antennaGain | number | Gain of the antenna (in dB) |
| antennaBeamwidth | number | 3 dB horizontal beamwidth of the antenna (in degrees) |
| antennaVerticalBeamwidth | number | 3 dB vertical beamwidth of the antenna (in degrees) |

### 5.7 CBSD record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: cbsd/$FCC_ID/$SerialNumber<br>• **$FCC_ID**: the FCC ID assigned to the device type in the FCC equipment authorization process<br>• **$SerialNumber**: the device manufacturer serial number that is unique within the FCC ID namespace scope |
| publicKey | string | • **Format**: X.509 key |
| installationParam | object: InstallationParam (see SAS-CBSD TR-B) | Contains device installation parameters |
| operationParam | array of OperationParam | Contains the outstanding grants |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| | (see SAS-CBSD TR-B) | |

### 5.8 Incumbent record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: incumbent/$SOURCE/$ID<br>• **$SOURCE**: the source of the incumbent information such as a specific FCC database, e.g. IBFS<br>• **$ID**: the identification of the referenced incumbent such as an FSS station call sign |
| type | string | • **Format:** enumeration value describing the incumbent class: “FSS”, “Federal”, or “3650” |
| deploymentParam | Array of DeploymentParam | Contains incumbent deployment parameters |

DeploymentParam object:

| | | |
|---|---|---|
| installationParam | object: InstallationParam (see SAS-CBSD TR-B) | Contains incumbent deployment parameters |
| operationParam | object: OperationParam (see SAS-CBSD TR-B) | Contains incumbent operating parameters. |
| protectionContour | string | • **Reference**: ID of a Zone record |

### 5.9 Zone record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: zone/$CREATOR/$ID<br>• **$CREATOR**: SAS Administrator ID or ESC Administrator ID or static government zone definition source ID<br>• **$ID**: the identification of the referenced zone defined by the $CREATOR |
| name | string | Human-readable local significant string |
| creator | string | • **Format**: Human-readable string, one of the following:<br>  • SAS Administrator record ID<br>  • ESC Administrator record ID |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| | | • Static government zone definition source ID |
| usage | string | • **Format:** Enumeration value describing the usage of the zone:<br>• "census tract"<br>• "service area"<br>• "protection contour"<br>• "XdBm contour", e.g. -80dBm, -95dBm, -110dBm<br>• "antenna pattern"<br>• "exclusion zone" |
| zone | object: GeoJSON ([N.2]) | Self-contained geometry description of the addressed zone. |

### 5.10 CBSD Operator record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: operator/$SAS_ADMINISTRATOR/$ID<br>• **$SAS_ADMINISTRATOR**: SAS Administrator ID<br>• **$ID**: Human-readable string, SAS Administrator Assigned CBSD Operator ID |
| name | string | Human-readable local significant string |
| contactInformation | object: ContactInformation | Contains various contact information |
| acknowledgement | string | Human-readable string communicating operator acceptance of interference from federal radars |

### 5.11 Professional Installer record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: installer/$INSTALLER<br>• **$INSTALLER**: SAS-CA certified unique installer identifier |
| name | string | Human-readable local significant string |
| publicKey | string | • **Format**: X.509 key |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| contactInformation | object: ContactInformation | Contains various contact information |
| piCertificationInformation | object: PICertificationInformation | Contains the certification information |

PICertificationInformation object:

| Field | Data Type | Field Definition |
|---|---|---|
| certificationBody | string | Issuer of the Professional Installer certificate |
| certificationID | string | The certification ID created by the issuer |
| certificationDate | string | Date of certification, in format YYYY-MM-DD |
| certificationExpiration | string | Date of certification expiration, in format YYYY-MM-DD |
| certificationConditions | string | Human-readable string or reference annotating the certification |

### 5.12 Coordination Event record

| Field | Data Type | Field Definition |
|---|---|---|
| ID | string | • **Format**: coordination/$SAS_ADMINISTRATOR/$ID<br>• **$SAS_ADMINISTRATOR**: SAS Administrator ID<br>• **$ID**: event record ID |
| name | string | • **Format**: Human-readable local unique reference to the event |
| creator | string | • **Format**: Human-readable string, one of the following:<br>  • SAS Administrator record ID<br>  • ESC Administrator record ID<br>  • Static government zone definition source ID |
| creationDate | string | • **Format**: structured object describing time and date |
| expirationDate | string | • **Format**: structured object describing time and date |
| description | string | • **Format**: Human-readable description of the coordination event. |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| coordinationType | string | • **Format**: Enumerated value indicating the type of event. [TBD] |
| coordinationDevice | array of string | • **Reference**: ID of the involved device (e.g. a CBSD ID or an incumbent ID) |
| coordinationZone | array of string | • **Reference**: Array of IDs of the involved zones |
| coordinationData | object: type is dependent upon the CoordinationType field | • **Format**: Structured object describing the coordination data<br>  • Per event specific<br>  • (TBD) Extensible anchor for any other metadata needed for automated handling of particular coordination events. |

# 6 SAS-SAS Exchange Protocol

This section specifies in detail the protocol two SASs will use in exchanging messages of the types specified in section 5.

## 6.1 Message Flow Overview

The message exchanges between two SASs are of the typical client-server request and response flows. The two SASs can symmetrically issue requests to their respective peer SASs independently and the peer SASs respond with either success or error responses. As already specified in the preceding TR-A[I.5], the SAS to SAS information exchanges allows flexible data push and pull for

- a specific data type record instance,
- a bulk of data record instances of a specific type created in a specific time range,
- wildcard retrieval or dump for all data type records in a specific time range and
- full activity retrieval and dump since a specific time.

In terms of message flows and procedures, all above cases are same except the differences in the request and response message contents exchanged between the two SASs, as shown in Figure 1 below.

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

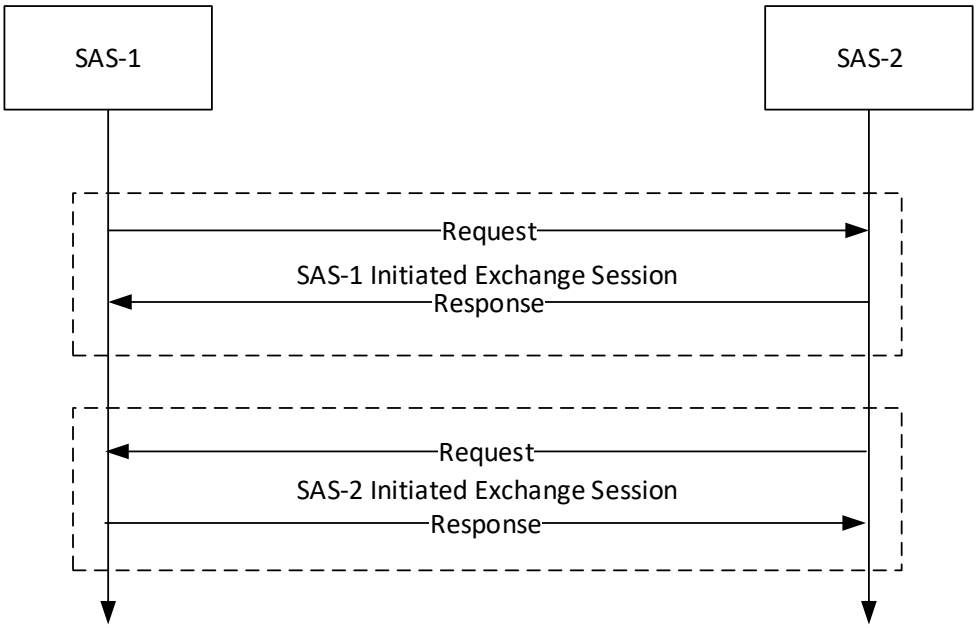


**Figure 1 SAS to SAS Exchange Flow**

## 6.2 Message Methods, Categories and Types

### *6.2.1 Message Methods*

Two methods, push and pull, are allowed for SAS to SAS information exchange.

- "Push": used by one SAS to push information to be shared to peer SASs.
- "Pull": used by one SAS to pull wanted information from peer SASs.

"Push" and "Pull" methods are directly mapped to the 'POST/PUT" and "GET" methods respectively defined in the HTTP protocol. No new definition is needed in the SAS to SAS information exchange messages.

### *6.2.2 URL endpoints*

URL endpoints for data exchange should be based on a base URL which is supplied by a SAS implementation, and when path defines the resource to be exchanged (requested or supplied) following this table. (Note this is an adjustment to the table included in TR-A.)

The construction uses the following format convention:

$BASE_URL/$RECORD_TYPE/$ID for single record exchange, where $RECORD_TYPE is the type of record to be exchanged and $ID is the url-escaped ID key for the record to be exchanged.

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

$BASE_URL/$RECORD_TYPE:searchByTime?startTime=$START&endTime=$END for time-range requests, where the $RECORD_TYPE is the type of records to be exchanged and the $START and $END parameters are url-escaped ISO 8601 time codes defining time limits for the records exchanged.

| Information Element Type | URL construction |
|---|---|
| SAS Administrators | Individual Records<br>Pull: `GET $BASE_URL/sas_admin/$ID`<br>Push: `POST $BASE_URL/sas_admin/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/sas_admin:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/sas_admin:searchByTime?start=$T1&end=$T2` |
| ESC Administrators | Individual Records<br>Pull: `GET $BASE_URL/esc_admin/$ID`<br>Push: `POST $BASE_URL/esc_admin/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/esc_admin:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/esc_admin:searchByTime?start=$T1&end=$T2` |
| SAS Implementations | Individual Records<br>Pull: `GET $BASE_URL/sas/$ID`<br>Push: `POST $BASE_URL/sas/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/sas:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/sas:searchByTime?start=T1&end=$T2` |
| ESC Implementations | Individual Records<br>Pull: `GET $BASE_URL/esc/$ID`<br>Push: `POST $BASE_URL/esc/$ID`<br><br>Time-range records |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | |
|---|---|
| | Pull: `GET $BASE_URL/esc:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/esc:searchByTime?start=T1&end=$T2` |
| CBSD device types | Individual Records<br>Pull: `GET $BASE_URL/cbsd_type/$ID`<br>Push: `POST $BASE_URL/cbsd_type/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/cbsd_type:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/cbsd_type:searchByTime?start=T1&end=$T2` |
| CBSDs | Individual Records<br>Pull: `GET $BASE_URL/cbsd/$ID`<br>Push: `POST $BASE_URL/cbsd/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/cbsd:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/cbsd:searchByTime?start=T1&end=$T2` |
| Incumbents | Individual Records<br>Pull: `GET $BASE_URL/incumbent/$ID`<br>Push: `POST $BASE_URL/incumbent/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/incumbent:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/incumbent:searchByTime?start=T1&end=$T2` |
| Zones | Individual Records<br>Pull: `GET $BASE_URL/zone/$ID`<br>Push: `POST $BASE_URL/zone/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/zone:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/zone:searchByTime?start=T1&end=$T2` |
| CBSD operators | Individual Records<br>Pull: `GET $BASE_URL/operator/$ID` |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | |
|---|---|
| | Push: `POST $BASE_URL/operator/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/operator:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/operator:searchByTime?start=T1&end=$T2` |
| Domain Proxies | Individual Records<br>Pull: `GET $BASE_URL/domain/$ID`<br>Push: `POST $BASE_URL/domain/$ID`<br><br>Time-range records<br>Pull: `GET $BASE_URL/domain:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/domain:searchByTime?start=T1&end=$T2` |
| Wildcard Retrieval for time-range records | Time-range records<br>Pull: `GET $BASE_URL/any:searchByTime?start=$T1&end=$T2`<br>Push: `POST $BASE_URL/any:searchByTime?start=T1&end=$T2` |
| Full activity dump | Pull: `GET $BASE_URL/dump`<br>Push: `POST $BASE_URL/dump` |

### *6.2.3 Message Categories*

All messages used for SAS and SAS information exchanges are of the following categories:

- "Request": request message sent from one SAS to a peer SAS to request a designated action, such as push or pull etc.

- "Response": response message sent from one SAS to a peer SAS in response to a prior received request. A response can be a "success" response indicating the successful processing of the corresponding request and containing the required response results. It can also be an "error" response indicating the unsuccessful handling of the corresponding request either because of errors in the request or required actions cannot be fulfilled. Appropriate error codes shall be contained in "error" responses to indicate the best known cause of the error condition. The format of the error condition data should follow the Error information element defined in the SAS-CBSD Protocol [N.3].

- "Notification": non-acknowledgement-required notification message from one SAS to a peer SAS. This is reserved for future use.

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

### *6.2.4 Message Types*

Message types are used to identify the action and results types carried in the request and response messages. In correspondence to the SAS to SAS exchange mechanisms as specified in TR-A[I.5], the following message types are required:

- "Individual": indicates request or response associated with a specific individual data record of a specific data record type.
- "Time-range": indicates request or response associated with a specific type data records created in a specific time range.
- "Wildcard": indicates request or response associated with all types of data records created in a specific time range.
- "Full-dump": indicates request or response associated with all types of data records created after a given time.

### *6.2.5 Time/Date formats*

Date and time references exchanged in the protocol (startTime and endTime fields) should follow ISO 8601 formats as described in RFC 3339 (example format: YYYY-MM-DDThh:mm:ss.ssssZ) and exchange times in UTC.

## 6.3 Message Contents Aggregation

In order to achieve efficient SAS to SAS exchange, it is allowed to aggregate multiple required data elements into single request and similarly the corresponding response will contain the aggregated data elements found.

## 6.4 Message Definition

### *6.4.1 Definition*

#### 6.4.1.1 Request message

| Field | Definition | |
|---|---|---|
| messageType | Message type, choose from:<br>• "Individual"<br>• "Time-range"<br>• "Wildcard"<br>• "Full-dump" | |
| records | Array of structured object specifying the SAS to SAS exchanged data records required, each structured object contains the following fields | |
| | recordType | SAS to SAS exchanged data record type, each choose from:<br>• "SAS Administrator" |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| | | • "ESC Administrator"<br>• "SAS Implementation"<br>• "ESC Implementation"<br>• "Domain Proxy"<br>• "CBSD Device Type" "CBSD"<br>• "Incumbent"<br>• "Zone"<br>• "CBSD Operator"<br>• "Professional Installer"<br>• "Coordination Event"<br>• "any" – for wildcard type |
| | recordId | SAS to SAS exchanged data record instance ID, each choose from:<br>• Unique record instance ID as specified in TR-A[I.5].<br>• "any" – for wildcard type |
| | startTime | String representation of standard timestamp indicating the start time of the concerned time range, not applicable in "individual" type, indicates the very beginning since server is started if absent. |
| | endTime | String representation of standard timestamp indicating the end time of the concerned time range, not applicable in "individual" and "full-dump" types, indicates "now" if absent. |

6.4.1.2 Response message

| Field | Definition | |
|---|---|---|
| messageType | Message type, choose from:<br>• "Individual"<br>• "Time-range"<br>• "Wildcard"<br>• "Full-dump" | |
| records | Array of structured object specifying the SAS to SAS exchanged data records responded, each structured object contains the following fields | |
| | recordType | SAS to SAS exchanged data record type, each choose from:<br>• "SAS Administrator"<br>• "ESC Administrator"<br>• "SAS Implementation"<br>• "ESC Implementation"<br>• "Domain Proxy"<br>• "CBSD Device Type" "CBSD" |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| | | • "Incumbent"<br>• "Zone"<br>• "CBSD Operator"<br>• "Professional Installer"<br>• "Coordination Event"<br>• "any" – for wildcard type |
| | recordId | SAS to SAS exchanged data record instance ID, each choose from:<br>• Unique record instance ID as specified in TR-A[I.5].<br>• "any" – for wildcard type |
| | recordData | Array of structured object contains required record data of the specific type, no "error" response |
| | error | Response status, defined by the Error information element in SAS-CBSD TR-B [N.3] and associated codes<br>• errorCode, errorData, errorMessage |
| | startTime | String representation of standard timestamp indicating the start time of the concerned time range, not applicable in "individual" type, indicates the very beginning since server is started if absent. |
| | endTime | String representation of standard timestamp indicating the end time of the concerned time range, not applicable in "individual" and "full-dump" types, indicates "now" if absent. |

#### 6.4.1.2.1 Response error codes

Response error codes are a subset of those 1xx category of general errors found in Section 8.13 of [N.3]. The following Error codes can be present in SAS-SAS responses.

| errorCode | Name | Description |
|---|---|---|
| 0 | SUCCESS | Request is successfully processed by the receiving SAS |
| 100 | VERSION | Version information contained in the request is not supported by the receiving SAS |
| 101 | BLACKLISTED | Requesting SAS is blacklisted by the peer receiving SAS |

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

| | | |
|---|---|---|
| 102 | MISSING_PARAM | Required parameters missing |
| 103 | INVALID_VALUE | One or more parameters have invalid value |
| 105 | RECORD_NOT_FOUND | Required record not found |

6.4.1.3 Notification message

TBD

### *6.4.2 JSON Representation Examples*

6.4.2.1 Request message (for a push exchange)

Multiple request records can be aggregated in on request message.

```
{
    “messageType” : “individual”,
    “records” : [
                {
                    “recordType” : “SAS-Administrator”
                    “recordId” : “admin/sas/$ADMINISTRATOR-ID”
                    “startTime” : “2015-07-16T19:30:27Z”,
                    “endTime” : “2015-07-17T19:30:27Z”
                },
                {
                    “recordType” : “SAS-Administrator”
                    “recordId” : “admin/sas/$ADMINISTRATOR-ID”
                    “startTime” : “2015-07-21T23:30:27Z”,
                    “endTime” : “2015-07-25T23:30:27Z”
                },
                {
                    // more records to be aggregated in the request if any
                    ……
                }
            ]
}
```

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

#### 6.4.2.2 Response message

Response is always on per request record basis, if multiple request records were aggregated into a single request message, a corresponding number of response records will be contained in the response message. And furthermore, “success” and “error” response records are allowed to be in the same response message.

- All success response

```
{
    “messageType” : “individual”,
    “records” : [
                {
                    “recordType” : “SAS-Administrator”
                    “recordId” : “admin/sas/$ADMINISTRATOR-ID”
                    “startTime” : “2015-07-16T19:30:27Z”,
                    “endTime” : “2015-07-17T19:30:27Z”
                    “recordData” : {
                        // record data JSON object
                     },
                     “error” : “OK”,
                },
                {
                    “recordType” : “SAS-Administrator”
                    “recordId” : “admin/sas/$ADMINISTRATOR-ID”
                    “startTime” : “2015-07-21T23:30:27Z”,
                    “endTime” : “2015-07-25T23:30:27Z”
                    “recordData” : {
                        // record data JSON object
                     },
                     “error” : “OK”,
                },
                {
                     // additional records found per corresponding request
                     ……
                 }
               ]
}
```

- Mixed Success and Error response

```
{
    “messageType” : “individual”,
    “records” : [
                {
                    “recordType” : “SAS-Administrator”
                    “recordId” : “admin/sas/$ADMINISTRATOR-ID”
                    “startTime” : “2015-07-16T19:30:27Z”,
```

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

```
                    "endTime" : "2015-07-17T19:30:27Z"
                    "recordData" : null,
                     "error" : {"errorCode": 103, "errorMessage": "required data not found"
                },
                {
                    "recordType" : "SAS-Administrator"
                    "recordId" : "admin/sas/$ADMINISTRATOR-ID"
                    "startTime" : "2015-07-21T23:30:27Z",
                    "endTime" : "2015-07-25T23:30:27Z"
                    "recordData" : {
                        // record data JSON object
                     },
                     "error" : "OK"
                },
                {
                     // additional response records
                     ……
                 }
               ]
}
```

- All Error response

```
{
     "messageType" : "individual",
     "records" : [
                 {
                    "recordType" : "SAS-Administrator"
                    "recordId" : "admin/sas/$ADMINISTRATOR-ID"
                    "startTime" : "2015-07-16T19:30:27Z",
                    "endTime" : "2015-07-17T19:30:27Z"
                    "recordData" : null,
                     "error" : {"errorCode": 104, "errorMessage": "required data not found"
                },
                {
                    "recordType" : "SAS-Administrator"
                    "recordId" : "admin/sas/$ADMINISTRATOR-ID"
                    "startTime" : "2015-07-21T23:30:27Z",
                    "endTime" : "2015-07-25T23:30:27Z"
                    "recordData" : null,
                     "error" : {"errorCode": 102, "errorMessage": "xxx parameter is missing"
                },
                {
                     // additional response records
                     ……
                 }
```

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved

]
}

Copyright © 2017 The Software Defined Radio Forum Inc.
All Rights Reserved