Central NG911 SSAP

Schema & Data Entry Guide

Author: John Soliman

Pacific Tech Systems

Version: 1.0

Schema export: 2026-02-14 21:03:41

Professional reference for SSAP data stewardship, quality assurance, and consistent multi-agency editing workflows.

Contents

1 Scope

This guide documents the **central SSAP (Site Structure Address Point)** schema for **SDE.NG911_SiteAddr** and provides field-by-field input guidance for editors across **Sicamous, Golden, Revelstoke, Salmon Arm, and CSRD**.

Geometry: Point features stored in NAD 1983 UTM Zone 11N (WKID 26911).

Key standard: For the *NENA street component fields* (pre/post directional and type), **abbreviations are not accepted**. Enter full words such as "Southwest" (not "SW") and "Road" (not "Rd").

2 Quick Start for Editors

2.1 Minimum recommended fields for a new record

To create a usable SSAP record that will auto-generate identifiers and a full address string, populate these first:

- `Agency` (owning agency)
- `DiscrpAgID` (stable agency identifier used in NGUID)
- `Add_Number` (civic number)
- `St_Name` (street name body)
- `A3` (locality/community; must match AddCode lookup)

If the address has a unit/suite, also populate `Unit`.

2.2 What happens on Save

On insert/update, attribute rules compute key outputs such as `Full_Addr`, `NGUID`, `AddCode`, and `DateUpdate`. In most editing views these should be treated as **read-only**.

2.3 Recommended web form structure (no duplicate fields)

Use the following groups in your web editing form to minimize scrolling and keep logic clear:

1. **Quick Entry (What editors fill first)**: `Agency`, `DiscrpAgID`, `Add_Number`, `St_Name`, `A3`, `Unit`
2. **Civic Number Components (Only if needed)**: `AddNum_Pre`, `AddNum_Suf`
3. **Street Components (No abbreviations)**: `St_PreMod`, `St_PreDir`, `St_PreTyp`, `St_PreSep`, `St_PosTyp`, `St_PosDir`, `St_PosMod`
4. **Jurisdiction & Communities**: `Country`, `A1`, `A2`, `Uninc_Comm`, `Nbrhd_Comm`
5. **Postal / MSAG / ESN**: `Post_Comm`, `Post_Code`, `PostCodeEx`, `MSAGComm`, `ESN`
6. **Legacy Street Fields (Optional)**: `LSt_PreDir`, `LSt_Name`, `LSt_Typ`, `LSt_PosDir`
7. **Unit / Interior Location (Optional)**: `Building`, `Floor`, `Room`, `Seat`, `Addtl_Loc`

8. **Place / Landmark / Placement (Optional)**: `Place_Type`, `Placement`, `LandmkName`, `Milepost`, `Elevation`
9. **Effective Dates (Optional)**: `Effective`, `Expire`
10. **References & Notes (Optional)**: `AddDataURI`, `Featureid`, `Parcelid`, `Roll`, `Addressnotes`, `Alternateaccess`
11. **Auto-Filled (Attribute Rules)**: `Full_Addr`, `NGUID`, `AddCode`, `DateUpdate`, `Longitude`, `Latitude`, `QAStatus`
12. **System Fields (Read-only / Hidden)**: `OBJECTID`, `GlobalID`, `created_user`, `created_date`, `last_edited_user`, `last_edited_date`, `Shape`

3 Multi-agency conventions

3.1 Agency ownership

The `Agency` field should represent **who owns/maintains the record**. It is the recommended basis for definition queries in agency-specific web maps and for reporting/QA dashboards.

3.2 Discrepancy Agency ID

The `DiscrpAgID` field represents the **agency domain** of the agency that owns and maintains the record, and is embedded in `NGUID`.

Important: `DiscrpAgID` is **not** the same as the `Agency` field. `Agency` is the display/label name; `DiscrpAgID` is the domain-based identifier.

`DiscrpAgID` values must be **stable** and **consistent per agency**.

Do not guess values. Use the approved mapping below and standardize these values in web form defaults and data QA checks.

Primary municipalities in this database: CSRD, Revelstoke, Salmon Arm, Golden, and Sicamous.

Other agencies in this table: First Nations and partner jurisdictions.

Agency/Locality	DiscrpAgID
CSRD	`csrd.bc.ca`
Revelstoke	`revelstoke.ca`
Salmon Arm	`salmonarm.ca`
Golden	`golden.ca`
Sicamous	`sicamous.ca`
Adams Lake	`adamslakeband.org`
Skw'lax te Secwepemcúl'ecw	`lslb.ca`
Neskonlith	`neskonlith.net`
RDNO	`rdno.ca`
Splatsin	`splatsin.ca`

4 Attribute rules

4.1 Summary

Rule	Target Field	Order	Triggers
Full Address	Full_Addr	1	Insert, Update
NGUID	NGUID	2	Insert, Update
AddCode	AddCode	5	Insert, Update, Delete
Longitude	Longitude	3	Insert, Update, Delete
Latitude	Latitude	4	Insert, Update, Delete
DateUpdate	DateUpdate	6	Insert, Update, Delete
QAstatus	QAStatus	7	Insert, Update

4.2 Rule notes for editors

- **Full Address:** built from Unit, civic number parts, street component fields, locality (A3), province (A1), and postal fields.
- **NGUID:** requires DiscrpAgID and uses GlobalID. If DiscrpAgID is blank, NGUID becomes NULL.
- **AddCode:** derives from A3 using a fixed lookup (Appendix B). If A3 is not recognized, AddCode becomes NULL.
- **Longitude/Latitude:** rules are fixed and now return decimal-degree coordinates from the source NAD 1983 UTM Zone 11N geometry.
- **QAStatus:** optional rule can default to Pending and reset Pending when key business fields change; see Appendix D.

5 Field reference (what to enter in each field)

This section is organized in the same order as the recommended form groups.

5.1 Quick Entry (What editors fill first)

5.1.1 Agency (Agency)

Type: Text(255) (Nullable)

Who edits: Editor

Purpose: Owning agency label (used for filtering and edit views).

What to enter: Choose/enter: Sicamous, Golden, Revelstoke, Salmon Arm, or CSRD (as applicable).

Formatting: Title case; consistent spelling.

Example: Salmon Arm

Notes: Use for definition queries in agency-specific web maps.

5.1.2 Discrepancy Agency ID (DiscrpAgID)

Type: Text(100) (Nullable)

Who edits: Editor

Purpose: Domain-based identifier for the owning/maintaining agency; embedded in NGUID.

What to enter: Enter the confirmed agency domain (for example, `sicamous.ca`) for the record owner.

Formatting: Lowercase, no spaces recommended.

Example: sicamous.ca

Used in: NGUID

Notes: Not the same as `Agency`. `Agency` is a label, while `DiscrpAgID` is the domain identifier used in NGUID.

5.1.3 Address Number (Add_Number)

Type: Integer (Nullable)

Who edits: Editor

Purpose: Primary civic number.

What to enter: Enter the house/civic number.

Formatting: Numbers only.

Example: 431

Used in: Full_Addr

Notes: Typically required for a deliverable address.

5.1.4 Street Name (St_Name)

Type: Text(254) (Nullable)

Who edits: Editor

Purpose: Core street name body.

What to enter: Enter street name only (no type/directional).

Formatting: Title case; numeric allowed.

Example: Main, Trans-Canada, 1

Used in: Full_Addr

5.1.5 Locality (A3) (A3)

Type: Text(100) (Nullable)

Who edits: Editor

Purpose: Locality/community name.

What to enter: Enter the locality name exactly as used by AddCode lookup.

Formatting: Must match lookup after uppercasing; periods ignored.

Example: Sicamous

Used in: Full_Addr, AddCode

Notes: If not recognized, AddCode will be NULL.

5.1.6 Unit (Unit)

Type: Text(75) (Nullable)

Who edits: Editor

Purpose: Unit/suite identifier (prefixes the civic number).

What to enter: Enter unit/suite identifier only (avoid adding the word 'Unit' unless your standard requires it).

Formatting: Examples: 12, A, 302.

Example: 12 (→ '12-431 ...')

Used in: Full_Addr

Notes: Full Address joins Unit and civic number as Unit-Number when both exist.

5.2 Civic Number Components (Only if needed)

5.2.1 Address Number Prefix (AddNum_Pre)

Type: Text(15) (Nullable)

Who edits: Editor

Purpose: Address number prefix (rare).

What to enter: Enter only if official addressing uses a prefix component.

Formatting: Numeric prefix inserts dash (12-123).

Example: A or 12

Used in: Full_Addr

Notes: Rare in most civic addressing; leave blank unless required.

5.2.2 Address Number Suffix (AddNum_Suf)

Type: Text(15) (Nullable)

Who edits: Editor

Purpose: Address number suffix.

What to enter: Enter letter/identifier after number if official.

Formatting: Numeric suffix inserts dash (123-1).

Example: A or 1

Used in: Full_Addr

Notes: Use suffix for letters; do not put letters in Add_Number.

5.3 Street Components (No abbreviations)

5.3.1 Street Name Pre Modifier (St_PreMod)

Type: Text(15) (Nullable)

Who edits: Editor

Purpose: Modifier before street name.

What to enter: Enter only if part of official name.

Formatting: Full words.

Example: Old, Upper

Used in: Full_Addr

5.3.2 Street Name Pre Directional (St_PreDir)

Type: Text(10) (Nullable)

Who edits: Editor

Purpose: Directional before street name.

What to enter: Enter full directional. No abbreviations: use full words (e.g., Southwest, not SW).

Formatting: North/South/East/West/Northeast/Northwest/Southeast/Southwest.

Example: Southwest

Used in: Full_Addr

Notes: No abbreviations: use full words (e.g., Southwest, not SW).

5.3.3 Street Name Pre Type (St_PreTyp)

Type: Text(50) (Nullable)

Who edits: Editor

Purpose: Type before street name (rare).

What to enter: Enter full type. No abbreviations: use full type words (e.g., Road, not Rd).

Formatting: Highway, Route.

Example: Highway

Used in: Full_Addr

Notes: No abbreviations: use full type words (e.g., Road, not Rd).

5.3.4 Street Name Pre Type Separator (St_PreSep)

Type: Text(20) (Nullable)

Who edits: Editor

Purpose: Separator between Pre Type and Street Name.

What to enter: Leave blank for space; use only if officially required.

Formatting: Blank → space; '-' only if required.

Example: (blank)

Used in: Full_Addr

Notes: Full Address defaults blank to a single space.

5.3.5 Street Name Post Type (St_PosTyp)

Type: Text(50) (Nullable)

Who edits: Editor

Purpose: Type after street name (common).

What to enter: Enter full type. No abbreviations: use full type words (e.g., Road, not Rd).

Formatting: Road, Street, Avenue, Boulevard, Crescent, Drive, Lane, Court, Place.

Example: Road

Used in: Full_Addr

Notes: No abbreviations: use full type words (e.g., Road, not Rd).

5.3.6 Street Name Post Directional (St_PosDir)

Type: Text(10) (Nullable)

Who edits: Editor

Purpose: Directional after street name/type.

What to enter: Enter full directional. No abbreviations: use full words (e.g., Southwest, not SW).

Formatting: North/South/East/West/Northeast/Northwest/Southeast/Southwest.

Example: South

Used in: Full_Addr

Notes: No abbreviations: use full words (e.g., Southwest, not SW).

5.3.7 Street Name Post Modifier (St_PosMod)

Type: Text(25) (Nullable)

Who edits: Editor

Purpose: Modifier after street name/type.

What to enter: Enter only if part of official name.

Formatting: Full words preferred.

Example: Extension, Spur

Used in: Full_Addr

5.4 Jurisdiction & Communities

5.4.1 Country (Country)

Type: Text(2) (Nullable)

Who edits: Editor

Purpose: Country code.

What to enter: Enter 2-letter code.

Formatting: Two letters.

Example: CA

Notes: Default to CA in most forms.

5.4.2 Province (A1) (A1)

Type: Text(2) (Nullable)

Who edits: Editor

Purpose: Province/State code.

What to enter: Enter 2-letter code.

Formatting: Two letters.

Example: BC

Notes: Default to BC in most forms.

5.4.3 Regional District (A2) (A2)

Type: Text(100) (Nullable)

Who edits: Editor

Purpose: Regional district name.

What to enter: Enter official regional district name.

Formatting: Title case.

Example: Columbia Shuswap Regional District

Notes: Often constant; set default where appropriate.

5.4.4 Unincorporated Community (A4) (Uninc_Comm)

Type: Text(100) (Nullable)

Who edits: Editor

Purpose: Unincorporated community (optional).

What to enter: Enter if applicable.

Formatting: Title case.

Example: Malakwa

5.4.5 Neighborhood Community (A5) (Nbrhd_Comm)

Type: Text(100) (Nullable)

Who edits: Editor

Purpose: Neighborhood/community (optional).

What to enter: Enter if applicable.

Formatting: Title case.

Example: Downtown

5.5 Postal / MSAG / ESN

5.5.1 Postal Community Name (Post_Comm)

Type: Text(40) (Nullable)

Who edits: Editor

Purpose: Postal community name (if different from A3).

What to enter: Enter if Canada Post community differs.

Formatting: Title case.

Example: Sicamous

Used in: Full_Addr

5.5.2 Postal Code (Post_Code)

Type: Text(7) (Nullable)

Who edits: Editor

Purpose: Postal code.

What to enter: Enter postal code.

Formatting: Standard format.

Example: V0E 2V0

Used in: Full_Addr

5.5.3 Postal Code Extension (PostCodeEx)

Type: Text(4) (Nullable)

Who edits: Editor

Purpose: Postal extension (optional).

What to enter: Enter only if used.

Formatting: Appends as Post_Code-PostCodeEx.

Example: 123

Used in: Full_Addr

Notes: Optional.

5.5.4 MSAG Community Name (MSAGComm)

Type: Text(30) (Nullable)

Who edits: Editor

Purpose: MSAG community name (legacy).

What to enter: Enter only if required by MSAG.

Formatting: Short label.

Example: SICAMOUS

Notes: Legacy routing.

5.5.5 ESN (ESN)

Type: Text(5) (Nullable)

Who edits: Editor

Purpose: Emergency Service Number (legacy).

What to enter: Enter only if used.

Formatting: Up to 5 chars.

Example: 01234

Notes: Legacy routing.

5.6 Legacy Street Fields (Optional)

5.6.1 Legacy Street Name Pre Directional (LSt_PreDir)

Type: Text(2) (Nullable)

Who edits: Editor

Purpose: Legacy pre-directional (optional).

What to enter: Populate only for legacy exports.

Formatting: Often abbreviated.

Example: N

Notes: Separate from NENA components.

5.6.2 Legacy Street Name (LSt_Name)

Type: Text(75) (Nullable)

Who edits: Editor

Purpose: Legacy street name (optional).

What to enter: Populate only for legacy exports.

Formatting: May include type/directional depending on legacy.

Example: Main St

5.6.3 Legacy Street Name Type (LSt_Typ)

Type: Text(4) (Nullable)

Who edits: Editor

Purpose: Legacy street type (optional).

What to enter: Populate only for legacy exports.

Formatting: Often abbreviated (length 4).

Example: Rd

Notes: Separate from NENA components.

5.6.4 Legacy Street Name Post Directional (LSt_PosDir)

Type: Text(2) (Nullable)

Who edits: Editor

Purpose: Legacy post-directional (optional).

What to enter: Populate only for legacy exports.

Formatting: Often abbreviated.

Example: SW

Notes: Separate from NENA components.

5.7 Unit / Interior Location (Optional)

5.7.1 Building (Building)

Type: Text(75) (Nullable)

Who edits: Editor

Purpose: Building name/identifier (optional).

What to enter: Enter if applicable.

Formatting: Title case.

Example: Lakeside Towers

5.7.2 Floor (Floor)

Type: Text(75) (Nullable)

Who edits: Editor

Purpose: Floor identifier (optional).

What to enter: Enter if applicable.

Formatting: Numeric or text.

Example: 3

5.7.3 Room (Room)

Type: Text(75) (Nullable)

Who edits: Editor

Purpose: Room identifier (optional).

What to enter: Enter if applicable.

Formatting: Numeric or text.

Example: 201

5.7.4 Seat (Seat)

Type: Text(75) (Nullable)

Who edits: Editor

Purpose: Seat/space identifier (optional).

What to enter: Enter if applicable.

Formatting: Free text.

Example: Section A Row 3 Seat 14

5.7.5 Additional Location Information (Addtl_Loc)

Type: Text(225) (Nullable)

Who edits: Editor

Purpose: Operational location notes (optional).

What to enter: Enter response-relevant access notes.

Formatting: Keep concise; avoid personal info.

Example: Rear entrance off alley; gate code 1234

5.8 Place / Landmark / Placement (Optional)

5.8.1 Place Type (Place_Type)

Type: Text(50) (Nullable)

Who edits: Editor

Purpose: Place/site type (optional).

What to enter: Choose from controlled list (recommended).

Formatting: Consistent capitalization.

Example: Residential

5.8.2 Placement Method (Placement)

Type: Text(25) (Nullable)

Who edits: Editor

Purpose: Placement method for geometry (optional).

What to enter: Choose from controlled list (recommended).

Formatting: Entrance/Driveway/Rooftop/Parcel centroid/Interpolated.

Example: Entrance

5.8.3 Complete Landmark Name (LandmkName)

Type: Text(150) (Nullable)

Who edits: Editor

Purpose: Complete landmark/facility name (optional).

What to enter: Enter if applicable.

Formatting: Title case.

Example: Sicamous Community Hall

5.8.4 Milepost (Milepost)

Type: Text(150) (Nullable)

Who edits: Editor

Purpose: Milepost reference (optional).

What to enter: Enter if applicable.

Formatting: Authority formatting.

Example: Hwy 1 MP 23

5.8.5 Elevation (Elevation)

Type: Integer (Nullable)

Who edits: Editor

Purpose: Elevation value (optional).

What to enter: Enter if required.

Formatting: Use consistent units (recommend meters).

Example: 520

Notes: Not auto-filled by provided rules.

5.9 Effective Dates (Optional)

5.9.1 Effective Date (`Effective`)

Type: Date/Time (Nullable)

Who edits: Editor

Purpose: Effective date (optional).

What to enter: Enter if tracked.

Formatting: Date.

Example: 2026-03-01

5.9.2 Expiration Date (`Expire`)

Type: Date/Time (Nullable)

Who edits: Editor

Purpose: Expiry/retirement date (optional).

What to enter: Enter if tracked.

Formatting: Date.

Example: 2027-10-31

5.10 References & Notes (Optional)

5.10.1 Additional Data URI (`AddDataURI`)

Type: Text(254) (Nullable)

Who edits: Editor

Purpose: Link/URI to supporting info (optional).

What to enter: Enter URL/URI if available.

Formatting: Text.

Example: (internal record link)

5.10.2 Feature ID (`Featureid`)

Type: Text(255) (Nullable)

Who edits: Editor

Purpose: External feature identifier (optional).

What to enter: Enter if integrating from another system.

Formatting: Text.

Example: SA-ADDR-001234

5.10.3 Parcel ID (Parcelid)

Type: Text(255) (Nullable)

Who edits: Editor

Purpose: Parcel identifier (optional).

What to enter: Enter if available.

Formatting: Text.

Example: 012-345-678

5.10.4 Roll (Roll)

Type: Text(15) (Nullable)

Who edits: Editor

Purpose: Assessment roll number (optional).

What to enter: Enter if used.

Formatting: Text.

Example: 12-345-678-901

5.10.5 Address Notes (Addressnotes)

Type: Text(255) (Nullable)

Who edits: Editor

Purpose: Non-operational notes (optional).

What to enter: Enter brief notes.

Formatting: Text.

Example: Created from subdivision plan 2026-02-10.

Notes: Avoid sensitive info.

5.10.6 Alternate Access (Alternateaccess)

Type: Text(255) (Nullable)

Who edits: Editor

Purpose: Alternate access notes (optional).

What to enter: Enter if needed.

Formatting: Text.

Example: Access via back lane off Main Street.

5.11 Auto-Filled (Attribute Rules)

5.11.1 Full Address (Full_Addr)

Type: Text(255) (Nullable)

Who edits: System

Purpose: Complete civic address string used for display and exports.

What to enter: Do not type. Auto-built from Unit, civic number parts, street components, locality/province, and postal fields when saved.

Formatting: Leave blank; populated on insert/update.

Example: 12-431 Old Southwest Highway 1 Road, Sicamous, BC V0E 2V0

Used in: Exports, QA review, downstream integrations.

Notes: Calculated by the "Full Address" attribute rule; blank components are omitted.

5.11.2 NENA Globally Unique ID (NGUID)

Type: Text(254) (Nullable)

Who edits: System

Purpose: NENA Globally Unique ID for SSAP features.

What to enter: Do not type. Ensure DiscrpAgID is correct; NGUID is generated on save.

Formatting: urn:emergency:uid:gis:SSAP:¡GlobalID¿:¡DiscrpAgID¿

Example: urn:emergency:uid:gis:SSAP:3F25...3301:sicamous.ca

Used in: NG911 identifiers; must remain stable.

Notes: If DiscrpAgID is blank, NGUID returns NULL.

5.11.3 Additional Code (AddCode)

Type: Text(6) (Nullable)

Who edits: System

Purpose: Numeric code derived from Locality (A3).

What to enter: Do not type. Populate A3 (Locality) with a value in the lookup table; AddCode is filled on save.

Formatting: A3 is uppercased and periods are removed before lookup.

Example: A3='SALMON ARM' → 574

Used in: Locality coding.

Notes: Returns NULL if A3 is empty or not in the lookup (see Appendix).

5.11.4 Date Updated (DateUpdate)

Type: Date/Time (Nullable)

Who edits: System

Purpose: Last edit timestamp for the record.

What to enter: Do not type.

Formatting: Auto-set to last_edited_date.

Example: 2026-02-19 14:22

Notes: Calculated by "DateUpdate" rule.

5.11.5 Longitude (Longitude)

Type: Double (Nullable)

Who edits: System

Purpose: Longitude (decimal degrees) for the feature location.

What to enter: Do not type.

Formatting: Decimal degrees (WGS84).

Example: -118.944321

Notes: Auto-calculated by the Longitude rule from NAD 1983 UTM Zone 11N to decimal degrees.

5.11.6 Latitude (Latitude)

Type: Double (Nullable)

Who edits: System

Purpose: Latitude (decimal degrees) for the feature location.

What to enter: Do not type.

Formatting: Decimal degrees (WGS84).

Example: 50.837221

Notes: Auto-calculated by the Latitude rule from NAD 1983 UTM Zone 11N to decimal degrees.

5.11.7 Quality Check Status (QAStatus)

Type: Text(255) (Nullable)

Who edits: QA / System

Purpose: QA/QC status for the record.

What to enter: If you are doing QA, set status using the controlled list; otherwise leave unchanged.

Formatting: Recommended: Pending, Pass, Fail, Needs Review.

Example: Pending

Used in: QA dashboards and checks.

Notes: Optional "QAstatus" rule can default to Pending on insert, preserve explicit edits, and reset to Pending when key fields change.

5.12 System Fields (Read-only / Hidden)

5.12.1 OBJECTID (OBJECTID)

Type: ObjectID (Not nullable)

Who edits: System

Purpose: System row identifier.

What to enter: Do not edit.

5.12.2 GlobalID (GlobalID)

Type: GlobalID (Not nullable)

Who edits: System

Purpose: System GUID (used in NGUID).

What to enter: Do not edit.

5.12.3 created_user (created_user)

Type: Text(255) (Nullable)

Who edits: System

Purpose: Editor tracking: creator username.

What to enter: Do not edit.

5.12.4 created_date (created_date)

Type: Date/Time (Nullable)

Who edits: System

Purpose: Editor tracking: creation date.

What to enter: Do not edit.

5.12.5 last_edited_user (last_edited_user)

Type: Text(255) (Nullable)

Who edits: System

Purpose: Editor tracking: last editor username.

What to enter: Do not edit.

5.12.6 last_edited_date (last_edited_date)

Type: Date/Time (Nullable)

Who edits: System

Purpose: Editor tracking: last edit date.

What to enter: Do not edit.

5.12.7 Shape (Shape)

Type: Geometry (Nullable)

Who edits: System

Purpose: Geometry.

What to enter: Edit by placing/moving the point; not an attribute.

6 Common examples

6.1 Typical single-family address

- `Add_Number` = 431
- `St_Name` = Main
- `St_PosTyp` = Road
- `A3` = Sicamous
- (Optional) `Post_Code` = V0E 2V0

On save, `Full_Addr` will be generated similar to: *431 Main Road, Sicamous, BC V0E 2V0* (depending on which fields are populated).

6.2 Typical multi-unit address

- `Unit` = 12
- `Add_Number` = 431
- `St_Name` = Main
- `St_PosTyp` = Road

On save, `Full_Addr` will begin with *12-431 ...* because Unit and civic number are joined with a dash.

A Field domains

A.1 Domain for Agency

Code	Description
CSRD	CSRD
Golden	Golden
Revelstoke	Revelstoke
Salmon Arm	Salmon Arm
Sicamous	Sicamous
Adams Lake	Adams Lake
Neskonlith	Neskonlith
RDNO	RDNO
Skw'lax te Secwepemcúl'ecw	Skw'lax te Secwepemcúl'ecw
Splatsin	Splatsin

A.2 Domain for DiscrpAgID

Code	Description
csrd.bc.ca	csrd.bc.ca
golden.ca	golden.ca
revelstoke.ca	revelstoke.ca
salmonarm.ca	salmonarm.ca
adamslakeband.org	adamslakeband.org
neskonlith.net	neskonlith.net
rdno.ca	rdno.ca
lslb.ca	lslb.ca
splatsin.ca	splatsin.ca
sicamous.ca	sicamous.ca

A.3 Domain for Locality (A3)

Code	Description
Anglemont	Anglemont
Anstey Arm	Anstey Arm
Beaton	Beaton
Blaeberry	Blaeberry
Blind Bay	Blind Bay
Canyon Hot Springs	Canyon Hot Springs

Code	Description
Castledale	Castledale
Celista	Celista
Craigellachie	Craigellachie
Deep Creek	Deep Creek
Donald	Donald
Eagle Bay	Eagle Bay
East Salmon Arm	East Salmon Arm
Enderby	Enderby
Falkland	Falkland
Field	Field
Galena Bay	Galena Bay
Golden	Golden
Grandview Bench	Grandview Bench
Heywoods	Heywoods
Leanchoil	Leanchoil
Lee Creek	Lee Creek
Magna Bay	Magna Bay
Malakwa	Malakwa
Mica Creek	Mica Creek
North Mara	North Mara
Notch Hill	Notch Hill
Okanagan IR North	Okanagan IR North
Parson	Parson
Revelstoke	Revelstoke
Rogers Pass	Rogers Pass
Salmon Arm	Salmon Arm
Salmon River IR	Salmon River IR
Scotch Creek	Scotch Creek
Seymour Arm	Seymour Arm
Sicamous	Sicamous
Solsqua	Solsqua
Sorrento	Sorrento
South Kinbasket Lake	South Kinbasket Lake
St. Ives	St. Ives
Tappen	Tappen
Three Valley	Three Valley
Trout Lake	Trout Lake
White Lake	White Lake

B AddCode lookup table

This table shows the current `A3` (Locality) to `AddCode` mapping used by the AddCode attribute rule. Ensure that locality values match these keys (case-insensitive; periods ignored).

A3 Locality Key	AddCode
ANSTEY ARM	1066
BEATON	60
BLAEBERRY	1054
BLIND BAY	581
CANYON HOT SPRINGS	1053
CASTLEDALE	18
CELISTA	594
CRAIGELLACHIE	182
DEEP CREEK	577
DONALD	180
EAGLE BAY	593
FALKLAND	586
FIELD	795
GALENA BAY	61
GOLDEN	179
HEYWOODS	587
LEANCHOLI	1056
LEE CREEK	597
MAGNA BAY	880
MALAKWA	185
MICA CREEK	183
NORTH MARA	584
NOTCH HILL	592
OKANAGAN IR NORTH	1119
PARSON	19
REVELSTOKE	181
ROGERS PASS	178
SALMON ARM	574
SALMON RIVER IR	1118
SCOTCH CREEK	598
SEYMOUR ARM	1065
SICAMOUS	583
SOLSQUA	186
SORRENTO	580
SOUTH KINBASKET LAKE	872
SPALLUMCHEEN	1075

A3 Locality Key	AddCode
ST IVES	595
TAPPEN	576
THREE VALLEY	555
TROUT LAKE	59
WHITE LAKE	582

C Attribute rule scripts (Arcade)

The following scripts are included for technical reference and troubleshooting.

C.1 Full Address

```
// --- Helpers ---
function isNumericText(val) {
  if (val == null || Trim(val) == "") return false;
  var n = Number(val);
  return !IsNan(n);
}

// --- Unit ---
var unitPart = Trim(DefaultValue($feature.Unit, ""));

// --- Address number: prefix + number + suffix (with dash if numeric
   prefix/suffix) ---
var pre = Trim(DefaultValue($feature.AddNum_Pre, ""));
var num = Trim(DefaultValue($feature.Add_Number, ""));
var suf = Trim(DefaultValue($feature.AddNum_Suf, ""));

var prefixSep = "";
if (pre != "" && num != "" && isNumericText(pre)) prefixSep = "-";

var suffixSep = "";
if (num != "" && suf != "" && isNumericText(suf)) suffixSep = "-";

var addrNumPart = Trim(pre + prefixSep + num + suffixSep + suf);

// --- Unit-Address with dash if both exist ---
var unitAddr = "";
if (unitPart != "" && addrNumPart != "") unitAddr = unitPart + "-" +
   addrNumPart;
else unitAddr = unitPart + addrNumPart;

// --- Street type prefix + separator + street name body ---
var preTyp = Trim(DefaultValue($feature.St_PreTyp, ""));
var stName = Trim(DefaultValue($feature.St_Name, ""));
```

```
var sepRaw = DefaultValue($feature.St_PreSep, "");
var sep = IIF(Trim(sepRaw) == "", " ", sepRaw);

var preTypName = "";
if (preTyp != "" && stName != "") {
  preTypName = Trim(preTyp + sep + stName);
} else {
  preTypName = Trim(preTyp + stName);
}

// --- Street part in required sequence ---
var streetArr = [];
var v = "";

v = Trim(DefaultValue($feature.St_PreMod, "")); if (v != "") Push(
   streetArr, v);
v = Trim(DefaultValue($feature.St_PreDir, "")); if (v != "") Push(
   streetArr, v);
if (preTypName != "") Push(streetArr, preTypName);
v = Trim(DefaultValue($feature.St_PosTyp, "")); if (v != "") Push(
   streetArr, v);
v = Trim(DefaultValue($feature.St_PosDir, "")); if (v != "") Push(
   streetArr, v);
v = Trim(DefaultValue($feature.St_PosMod, "")); if (v != "") Push(
   streetArr, v);

var streetPart = Concatenate(streetArr, " ");

// --- Base: Unit-AddrNum, Street ---
var baseArr = [];
if (unitAddr != "") Push(baseArr, unitAddr);
if (streetPart != "") Push(baseArr, streetPart);

var base = Concatenate(baseArr, " ");

// --- Postal (Post_Code + optional -PostCodeEx) ---
var post = Trim(DefaultValue($feature.Post_Code, ""));
var postEx = Trim(DefaultValue($feature.PostCodeEx, ""));
var postalPart = IIF(post != "" && postEx != "", post + "-" + postEx,
   post);

// --- Tail: muni, prov, postal (muni/prov with comma) ---
var muni = Trim(DefaultValue($feature.A3, ""));
var prov = Trim(DefaultValue($feature.A1, ""));

var muniProv = "";
if (muni != "" && prov != "") muniProv = muni + ", " + prov;
```

```
else muniProv = muni + prov;

var tailArr = [];
if (muniProv != "") Push(tailArr, muniProv);
if (postalPart != "") Push(tailArr, postalPart);

var tail = Concatenate(tailArr, " ");

// --- Final: base, tail ---
var full = IIF(base != "" && tail != "", base + ", " + tail, base + tail
   );
return IIF(Trim(full) == "", Null, full);
```

C.2 NGUID

```
var ag = Trim(DefaultValue($feature.DiscrpAgID, ""));
if (ag == "") return Null;

// GlobalID sometimes comes with { } - strip them
var gid = Text($feature.GlobalID);
gid = Replace(Replace(gid, "{", ""), "}", "");

return "urn:emergency:uid:gis:SSAP:" + gid + ":" + ag;
```

C.3 AddCode

```
var key = Upper(Replace(Trim($feature.A3), ".", ""));
if (IsEmpty(key)) {
  return null;
}

var m = {
  "ANSTEY ARM": 1066,
  "BEATON": 60,
  "BLAEBERRY": 1054,
  "BLIND BAY": 581,
  "CANYON HOT SPRINGS": 1053,
  "CASTLEDALE": 18,
  "CELISTA": 594,
  "CRAIGELLACHIE": 182,
  "DEEP CREEK": 577,
  "DONALD": 180,
  "EAGLE BAY": 593,
  "FALKLAND": 586,
  "FIELD": 795,
  "GALENA BAY": 61,
  "GOLDEN": 179,
```

```
  "HEYWOODS": 587,
  "LEANCHOLI": 1056,
  "LEE CREEK": 597,
  "MAGNA BAY": 880,
  "MALAKWA": 185,
  "MICA CREEK": 183,
  "NORTH MARA": 584,
  "NOTCH HILL": 592,
  "OKANAGAN IR NORTH": 1119,
  "PARSON": 19,
  "REVELSTOKE": 181,
  "ROGERS PASS": 178,
  "SALMON ARM": 574,
  "SALMON RIVER IR": 1118,
  "SCOTCH CREEK": 598,
  "SEYMOUR ARM": 1065,
  "SICAMOUS": 583,
  "SOLSQUA": 186,
  "SORRENTO": 580,
  "SOUTH KINBASKET LAKE": 872,
  "SPALLUMCHEEN": 1075,
  "ST IVES": 595,
  "TAPPEN": 576,
  "THREE VALLEY": 555,
  "TROUT LAKE": 59,
  "WHITE LAKE": 582
};

return IIF(HasKey(m, key), Number(m[key]), null);
```

C.4 Longitude

```
  // Longitude (WGS84/GRS80 decimal degrees) from NAD83 / UTM Zone 11N
     (26911)
  // Assumes Northern Hemisphere (BC is Northern)

  var g = Geometry($feature);
  if (IsEmpty(g)) return Null;
  if (TypeOf(g) != "Point") g = Centroid(g);

  var xUTM = g.x;
  var yUTM = g.y;
  if (IsEmpty(xUTM) || IsEmpty(yUTM)) return Null;

  // If coordinates already look like lat/long, keep them
  if (Abs(xUTM) <= 180 && Abs(yUTM) <= 90) return Round(xUTM, 6);
```

```
// ---- GRS80 (NAD83) constants ----
var a  = 6378137.0;
var f  = 1.0 / 298.257222101;
var e2 = 2*f - f*f;
var ep2 = e2 / (1 - e2);
var k0 = 0.9996;

// UTM zone 11N
var zone = 11;
var lon0 = (zone - 1) * 6 - 180 + 3;   // -117 degrees central
   meridian

var x = xUTM - 500000.0;
var y = yUTM;

var M  = y / k0;
var mu = M / (a * (1 - e2/4 - 3*Pow(e2,2)/64 - 5*Pow(e2,3)/256));

var e1 = (1 - Sqrt(1 - e2)) / (1 + Sqrt(1 - e2));

var J1 = (3*e1/2 - 27*Pow(e1,3)/32);
var J2 = (21*Pow(e1,2)/16 - 55*Pow(e1,4)/32);
var J3 = (151*Pow(e1,3)/96);
var J4 = (1097*Pow(e1,4)/512);

var fp = mu + J1*Sin(2*mu) + J2*Sin(4*mu) + J3*Sin(6*mu) + J4*Sin(8*mu
   );

var sinfp = Sin(fp);
var cosfp = Cos(fp);
var tanfp = Tan(fp);

var C1 = ep2 * Pow(cosfp, 2);
var T1 = Pow(tanfp, 2);

var N1 = a / Sqrt(1 - e2 * Pow(sinfp, 2));
var D  = x / (N1 * k0);

var lonRad =
  (D
   - (1 + 2*T1 + C1) * Pow(D,3) / 6
   + (5 - 2*C1 + 28*T1 - 3*Pow(C1,2) + 8*ep2 + 24*Pow(T1,2)) * Pow(D
      ,5) / 120) / cosfp;

return Round(lon0 + (lonRad * 180 / PI), 6);
```

C.5 Latitude

```
// Latitude (WGS84/GRS80 decimal degrees) from NAD83 / UTM Zone 11N
   (26911)
// Assumes Northern Hemisphere (BC is Northern)

var g = Geometry($feature);
if (IsEmpty(g)) return Null;
if (TypeOf(g) != "Point") g = Centroid(g);

var xUTM = g.x;
var yUTM = g.y;
if (IsEmpty(xUTM) || IsEmpty(yUTM)) return Null;

// If coordinates already look like lat/long, keep them
if (Abs(xUTM) <= 180 && Abs(yUTM) <= 90) return Round(yUTM, 6);

// ---- GRS80 (NAD83) constants ----
var a  = 6378137.0;
var f  = 1.0 / 298.257222101;
var e2 = 2*f - f*f;
var ep2 = e2 / (1 - e2);
var k0 = 0.9996;

// UTM zone 11N
var x = xUTM - 500000.0;  // remove false easting
var y = yUTM;             // north hemisphere: no false northing

var M  = y / k0;
var mu = M / (a * (1 - e2/4 - 3*Pow(e2,2)/64 - 5*Pow(e2,3)/256));

var e1 = (1 - Sqrt(1 - e2)) / (1 + Sqrt(1 - e2));

var J1 = (3*e1/2 - 27*Pow(e1,3)/32);
var J2 = (21*Pow(e1,2)/16 - 55*Pow(e1,4)/32);
var J3 = (151*Pow(e1,3)/96);
var J4 = (1097*Pow(e1,4)/512);

var fp = mu + J1*Sin(2*mu) + J2*Sin(4*mu) + J3*Sin(6*mu) + J4*Sin(8*mu
   );

var sinfp = Sin(fp);
var cosfp = Cos(fp);
var tanfp = Tan(fp);

var C1 = ep2 * Pow(cosfp, 2);
var T1 = Pow(tanfp, 2);
```

```
  var N1 = a / Sqrt(1 - e2 * Pow(sinfp, 2));
  var R1 = a * (1 - e2) / Pow(1 - e2 * Pow(sinfp, 2), 1.5);
  var D  = x / (N1 * k0);

  var Q1 = (N1 * tanfp) / R1;
  var Q2 = Pow(D, 2) / 2;
  var Q3 = (5 + 3*T1 + 10*C1 - 4*Pow(C1,2) - 9*ep2) * Pow(D,4) / 24;
  var Q4 = (61 + 90*T1 + 298*C1 + 45*Pow(T1,2) - 252*ep2 - 3*Pow(C1,2))
     * Pow(D,6) / 720;

  var latRad = fp - Q1 * (Q2 - Q3 + Q4);
  return Round(latRad * 180 / PI, 6);
```

C.6 DateUpdate

```
$feature.last_edited_date
```

D QAStatus rule script (Arcade)

```
// Preserve explicit QAStatus edits (including GP tool writes)
if ($originalFeature != null && $feature.QAStatus != $originalFeature.
   QAStatus) {
    return $feature.QAStatus;
}

// On insert, keep incoming QAStatus if provided (e.g., from QA->DEFAULT
    post);
// otherwise default to Pending.
if ($originalFeature == null) {
    var incomingStatus = Trim(DefaultValue($feature.QAStatus, ""));
    return IIF(incomingStatus == "", "Pending", $feature.QAStatus);
}

function sameValue(a, b) {
    if (a == null && b == null) return true;
    if (a == null || b == null) return false;
    return Text(a) == Text(b);
}

// Watch only QA-relevant business fields (exclude DateUpdate to avoid
   rule loops)
var changed =
    !sameValue($feature.DiscrpAgID, $originalFeature.DiscrpAgID) ||
    !sameValue($feature.NGUID, $originalFeature.NGUID) ||
    !sameValue($feature.Country, $originalFeature.Country) ||
    !sameValue($feature.A1, $originalFeature.A1) ||
```

```
    !sameValue($feature.A2, $originalFeature.A2) ||
    !sameValue($feature.A3, $originalFeature.A3) ||
    !sameValue($feature.Full_Addr, $originalFeature.Full_Addr);

// If watched fields changed and status wasn't already Pending, reset to
    Pending
var oldStatus = Lower(Trim(DefaultValue($originalFeature.QAStatus, "")))
   ;
var isPending = Find("pending", oldStatus) == 0;

if (changed && !isPending) {
    return "Pending";
}

// Otherwise keep current status
return $feature.QAStatus;
```

E Web form default-value expressions (Arcade)

Use **form calculated expressions** to pre-fill constant fields on INSERT (new features). Create separate web maps per agency and set defaults for `Agency`, `DiscrpAgID`, `Country`, `A1`, `A2`, and (optionally) `A3`.

E.1 Template (INSERT-only default)

```
if ($editcontext.editType == "INSERT") {
  return DefaultValue($feature.FIELDNAME, "DEFAULT_VALUE");
}
return $feature.FIELDNAME;
```

E.2 Example: Sicamous

```
// Agency
if ($editcontext.editType == "INSERT") return DefaultValue($feature.
   Agency, "Sicamous");
return $feature.Agency;

// DiscrpAgID (confirm official value per agency)
if ($editcontext.editType == "INSERT") return DefaultValue($feature.
   DiscrpAgID, "sicamous.ca");
return $feature.DiscrpAgID;

// Country / Province / Regional District / Locality (edit to your
   standards)
if ($editcontext.editType == "INSERT") return DefaultValue($feature.
   Country, "CA");
return $feature.Country;
```

Repeat per agency, substituting confirmed DiscrpAgID values and locality defaults as appropriate.

F References

F.1 Standards

1. **NENA Standard for NG9-1-1 GIS Data Model** (NENA-STA-006.2-2022).
   Defines required and recommended GIS data layers for location databases supporting NG9-1-1, including the Site/Structure Address Point (SSAP) layer on which this schema is based.
   https://cdn.ymaws.com/www.nena.org/resource/resmgr/standards/nena-sta-006.2-2022_ng9-1-1_.pdf

2. **NENA Next Generation 9-1-1 Canadian Civic Location Data Exchange Format** (CLDXF-CA, NENA-STA-029.1-2025).
   Defines the Canadian profile of the PIDF-LO for exchanging civic location data in NG9-1-1 systems deployed in Canada. Specifies normative civic address element definitions, parsing rules, administrative levels by province/territory, and the PIDF-LO construct for interoperable location exchange.
   https://cdn.ymaws.com/www.nena.org/resource/resmgr/standards/NENA-STA-029.1-2025_CLDXF-CA.pdf

F.2 Web resources

- **BC NG9-1-1 GIS Data Hub**
  Provincial portal for NG9-1-1 GIS data standards, templates, and validation tools relevant to British Columbia addressing authorities.
  https://ng911gisdata.gov.bc.ca/pages/gisdata